Désambiguïsation lexicale de l'arabe pour et par la traduction automatique

Marwa Hadj Salah

Communauté UNIVERSITÉ Grenoble Alpes

THÈSE

Pour obtenir le grade de

DOCTEUR DE LA COMMUNAUTE UNIVERSITE GRENOBLE ALPES ET L'UNIVERSITE DE SFAX

préparée dans le cadre d'une cotutelle *entre la Communauté Université Grenoble Alpes* et l'Université de Sfax

Spécialité : **Informatique**

Arrêté ministériel : 25 mai 2016

Présentée par

Marwa Hadj Salah

Thèse dirigée par **Hervé Blanchon** et **Mounir Zrigui**
codirigée par **Didier Schwab**

préparée au sein du **Laboratoire d'Informatique de Grenoble (LIG)** et du **Laboratoire de Technologies de l'Information et de la Communication & Génie Electrique (LaTICE)**

dans **l'École Doctorale Mathématiques, Sciences et Technologies de l'Information, Informatique** et l'**École Doctorale Informatique de la Faculté des sciences économiques et de gestion de Sfax**

Désambiguïsation Lexicale de l'Arabe pour et par la Traduction Automatique

Thèse soutenue publiquement le **18 décembre 2018,**
devant le jury composé de :

M. Mohamed JEMNI
Professeur, Université de Tunis, Rapporteur
M. Patrick PAROUBEK
Ingénieur de recherche, LIMSI - CNRS, Rapporteur
M. Kamel SMAILI
Professeur, Université de Lorraine, Président
M. Hervé BLANCHON
Maître de conférences (HDR), Université Grenoble Alpes, Directeur de thèse
M. Mounir ZRIGUI
Professeur, Université de Monastir, Directeur de thèse
M. Didier SCHWAB
Maître de conférences, Université Grenoble Alpes, Co-directeur de thèse

Remerciements

Je tiens tout d'abord à exprimer mon immense gratitude à mes encadrants de thèse M. Hervé Blanchon (Maître de Conférences au sein de l'Université Grenoble Alpes-France), M. Mounir Zrigui (Professeur à la Faculté des Sciences de Monastir-Tunisie) et M.Didier Schwab (Maître de Conférences au sein de l'Université Grenoble Alpes-France) pour m'avoir guidé et pour le soutien qu'ils m'ont apporté tout au long de la réalisation de ce travail de recherche. Ils ont su me faire profiter de leurs profondes connaissances et leurs nombreuses compétences dans le domaine du traitement automatique de la langue. Qu'il me soit permis de leur exprimer mon plus profond respect et mes vifs remerciements.

Je tiens aussi à exprimer mes remerciements à M.Mohamed Jemni Professeur à l'Université de Tunis ainsi que M.Patrick Paroubek Ingénieur de Recherche (HDR) au LIMSI (CNRS), qui m'ont fait l'honneur de bien vouloir accepter d'évaluer ce travail et d'en être les rapporteurs. Je remercie également M.Kamel Smaili Professeur à l'Université de Lorraine pour avoir accepter d'évaluer cette thèse et pour l'intérêt qu'il a porté à mon travail.

Je profite de cette occasion pour exprimer mon respect et mes sincères remerciements à toutes les personnes qui travaillent au sein de l'équipe GETALP du Laboratoire d'Informatique de Grenoble. Je pense particulièrement à M.Laurent Besacier, et à toute personne qui a participé de près ou de loin dans l'accomplissement de ce modeste travail.

Mes pensées vont bien entendu à toute ma famille pour leurs soutiens et pour la confiance qu'ils m'ont toujours accordée. Merci infiniment !

Un grand merci à mon mari, pour ses nombreuses aides et son continuel encouragement durant la réalisation de cette thèse de doctorat.

Résumé

Nous abordons dans cette thèse une étude sur la tâche de la désambiguïsation lexicale qui est une tâche centrale pour le traitement automatique des langues, et qui peut améliorer plusieurs applications telles que la traduction automatique ou l'extraction d'informations. Les recherches en désambiguïsation lexicale concernent principalement l'anglais, car la majorité des autres langues manque d'une référence lexicale standard pour l'annotation des corpus, et manque aussi de corpus annotés en sens pour l'évaluation, et plus important pour la construction des systèmes de désambiguïsation lexicale. En anglais, la base de données lexicale *Princeton WordNet* est une norme *de-facto* de longue date utilisée dans la plupart des corpus annotés et dans la plupart des campagnes d'évaluation. Notre contribution porte sur plusieurs axes : dans un premier temps, nous présentons une méthode pour la création automatique de corpus annotés en sens pour n'importe quelle langue, en tirant parti de la grande quantité de corpus anglais annotés en sens *Princeton WordNet*, et en utilisant un système de traduction automatique. Cette méthode est appliquée sur la langue arabe et est évaluée sur le seul corpus arabe, qui à notre connaissance, soit annoté manuellement en sens *Princeton WordNet* : l'OntoNotes 5.0 arabe que nous avons enrichi semi-automatiquement. Son évaluation est réalisée grâce à la mise en œuvre de deux systèmes supervisés (SVM, LSTM) qui sont entraînés sur les corpus produits avec notre méthode. Grâce ce travail, nous proposons ainsi une base de référence solide pour l'évaluation des futurs systèmes de désambiguïsation lexicale de l'arabe, en plus des corpus arabes annotés en sens que nous fournissons en tant que ressource librement disponible. Dans un second temps, nous proposons une évaluation *in vivo* de notre système de désambiguïsation de l'arabe en mesurant sa contribution à la performance de la tâche de traduction automatique.

Mots clés : Désambiguïsation lexicale, Traduction automatique, Portage des annotations, Enrichissement de corpus.

Abstract

This thesis concerns a study of Word Sense Disambiguation (WSD), which is a central task in natural language processing and that can improve applications such as machine translation or information extraction. Researches in word sense disambiguation predominantly concern the English language, because the majority of other languages lacks a standard lexical reference for the annotation of corpora, and also lacks sense annotated corpora for the evaluation, and more importantly for the construction of word sense disambiguation systems. In English, the lexical database *Princeton WordNet* is a long-standing *de-facto* standard used in most sense annotated corpora and in most WSD evaluation campaigns. Our contribution to this thesis focuses on several areas : first of all, we present a method for the automatic creation of sense annotated corpora for any language, by taking advantage of the large amount of *Princeton WordNet* sense annotated English corpora, and by using a machine translation system. This method is applied on Arabic and is evaluated, on the only, to our knowledge, Arabic manually sense annotated corpus with *Princeton WordNet* : the Arabic OntoNotes 5.0, which we have semi-automatically enriched. Its evaluation is performed thanks to an implementation of two supervised word sense disambiguation systems that are trained on the corpora produced using our method. We hence propose a solid baseline for the evaluation of future Arabic word sense disambiguation systems, in addition to sense annotated Arabic corpora that we provide as a freely available resource. Secondly, we propose an *in vivo* evaluation of our Arabic word sense disambiguation system by measuring its contribution to the performance of the machine translation task.

Key Words : Word Sense Disambiguation, Machine translation, Annotation transfert, Corpus enrichment.

Table des matières

Liste des tableaux

Table des figures

Acronymes

TAL Traitement automatique des langues

ASM Arabe standard moderne

AD Arabe dialectal

AR Arabe

EN Anglais

AWN WordNet arabe

PWN Princeton WordNet anglais

DL Désambiguïsation lexicale

TA Traduction automatique

TAS Traduction automatique statistique

NMT Traduction automatique neuronale

ML Modèle de langue

MT Modèle de traduction

RNN Réseaux de neurones récurrents

LSTM Mémoire à long et court terme

SVM Machine à vecteurs supports

Introduction générale

La clarification de texte est une tâche centrale pour le traitement automatique des langues, qui peut permettre d'améliorer de nombreuses applications comme l'extraction d'informations multilingues, le résumé automatique ou encore la traduction automatique.
Il s'agit de lever manuellement ou automatiquement un certains nombre d'ambiguïtés telle que l'ambiguïté lexicale qui représente l'une des grandes difficultés du traitement automatique du langage naturel, étant donné que certains mots peuvent avoir plusieurs significations.

La désambiguïsation lexicale est l'un des problèmes importants à résoudre directement ou indirectement pour traiter automatiquement le langage naturel. Cette thèse se situe dans le cadre de la désambiguïsation lexicale et vise principalement l'arabe qui est considérée comme une langue peu dotée pour cette tâche.
La désambiguïsation lexicale visera plus particulièrement deux applications importantes qui permettront, d'une part, l'évaluation *in vivo* (appelée aussi extrinsic) [Jones, 1994] du module de désambiguïsation lexicale et l'amélioration de la traduction automatique. Évaluer la façon dont un module de désambiguïsation lexicale exécute une tâche de désambiguïsation est considéré comme une évaluation *in vitro* (appelée aussi intrinsic) [Jones, 1994]. Il s'agit de l'évaluation des résultats de la tâche elle-même. L'évaluation de la façon dont la désambiguïsation lexicale contribue à une autre tâche, ici la traduction automatique, est considérée comme une évaluation *in vivo*.

Nous proposons deux objectifs principaux pour la thèse :

1. Désambiguïsation lexicale de l'arabe : dans un premier temps nous présentons une méthode pour la création automatique de corpus annotés en sens pour n'importe quelle langue, en tirant parti de la grande quantité de corpus anglais annotés en sens WordNet, et en utilisant un système de traduction automatique. Cette

méthode est utilisée pour la langue arabe. Nous exploiterons ces corpus annotés pour créer un système de désambiguïsation lexicale pour l'arabe.

2. Amélioration de la traduction automatique avec la désambiguïsation lexicale : dans un deuxième temps nous utilisons la désambiguïsation lexicale pour la traduction automatique. Plus précisément, nous annotons les corpus parallèles utilisés lors de l'entraînement des systèmes de traduction automatique à l'aide du système de désambiguïsation lexicale préalablement développé. Les prédictions produites devraient améliorer la traduction automatique de l'arabe. Il s'agira d'étudier dans quelle mesure ces annotations permettent d'améliorer la traduction automatique.

Deux types de ressources sont nécessaires à la désambiguïsation lexicale : des bases lexicales et des corpus annotés en sens. Ce sont particulièrement les secondes qui sont absentes pour la plupart des langues et en particulier pour l'arabe. En effet, les difficultés posées par leur création créent un manque important qui empêche non seulement la création des systèmes de désambiguïsation mais aussi leur évaluation. Il en résulte, par conséquence, l'impossibilité de comparer les performances des différents systèmes. Par exemple, dans le cas de l'arabe, qui nous intéresse plus particulièrement ici, il est rare que deux systèmes de désambiguïsation lexicale soient comparés sur le même corpus d'évaluation : la comparaison qualitative des approches est donc difficile à évaluer.
Tandis que l'anglais est la langue qui possède la plus grande quantité de telles ressources annotées, la plupart des autres langues n'en possèdent pas ou trop peu pour construire des systèmes robustes. Nous présentons dans ce manuscrit une méthode pour exploiter des données en anglais annotées (corpus UFSAC) afin de créer des systèmes de désambiguïsation lexicale pour n'importe quelle autre langue pour laquelle nous disposons d'un système de traduction automatique depuis l'anglais.

Contributions : Dans ce travail, nous présentons les problèmes posés à la désambiguïsation lexicale d'une langue moins dotée que l'anglais comme l'est l'arabe. Nous montrons que nous avons essayé de pallier ce manque par une méthode de création automatique de corpus annotés en sens à partir de corpus annotés en sens provenant d'une autre langue. Notre méthode consiste à traduire des corpus anglais en arabe et à porter les annotations de l'anglais vers l'arabe. Dans ce manuscrit, nous présentons les contributions qui seront disponibles pour la communauté :

1. 12 corpus arabes nouvellement créés annotés par des «synsets» issus du Princeton WordNet.

2. un corpus de l'arabe manuellement créé (la partie arabe d'OntoNotes Release 5.0) que nous avons complétée avec son annotation avec des «synsets» issus du Princeton WordNet.
3. Deux systèmes de traduction automatique (statistique et neuronale) et de portage des annotations d'une langue riche en corpus annotés (l'anglais) vers une langue moins bien dotée (l'arabe).
4. Deux systèmes de désambiguïsation lexicale arabe : un premier système basé sur des séparateurs à vaste marge et un second système basé sur les réseaux de neurones récurents (LSTM bidirectionnel), qui montrent que l'on peut utiliser les 12 corpus créés pour entraîner un système. Ces systèmes sont évalués sur le corpus OntoNotes Release 5.0 mis à jour.
5. Un système de traduction automatique neuronal capable de prendre en compte des données d'entraînement factorisées, et qui peut produire en sortie un corpus annoté en sens avec Princeton WordNet.

Structure du manuscrit

Ce manuscrit est organisé en deux grandes parties : Nous présentons dans un premier temps, l'état de l'art de la langue arabe en TAL, de la désambiguïsation lexicale en général et celle de l'arabe en particulier. Nous montrons que le manque des ressources en arabe pose des problèmes à la fois pour améliorer, pour évaluer les systèmes désambiguïsation lexicale arabes et pour les comparer entre eux.

Nous développons dans un second temps, les contributions réalisées durant cette thèse, à savoir la mise en œuvre de la méthode proposée pour la construction d'un système de désambiguïsation lexicale pour l'arabe, l'élaboration de ressources tant pour la création d'un système de désambiguïsation lexicale supervisé que pour son évaluation, ensuite, l'exploitation des ressources produites pour construire le système envisagé, et enfin l'étude de l'apport de la tâche de désambiguïsation lexicale arabe pour la traduction automatique.

Ainsi, nous présentons dans le premier chapitre, la langue arabe en nous focalisant sur les principales caractéristiques de cette langue. Également, nous présentons quelques ressources et outils de TAL arabe existants.

Dans le deuxième chapitre, nous présentons la tâche de désambiguïsation lexicale, son processus de mise en œuvre, les ressources génériques utiles, les approches existantes, ainsi que les deux méthodes de son évaluation. Ensuite, nous présentons plus

particulièrement la désambiguïsation lexicale de l'arabe en illustrant l'état de la langue arabe pour cette tâche et notre objectif, ainsi que les travaux connexes.

Dans le chapitre 3, nous présentons la première partie de nos contributions. Il s'agit de décrire le cadre expérimental de notre travail.

Nous nous intéressons dans le chapitre 4 à enrichir un corpus de référence existant, et à produire des corpus arabes annotés en sens en appliquant une méthode que nous proposons pour la construction d'un système de désambiguïsation lexicale supervisé arabe qui se base sur la traduction automatique.

Le dernier chapitre a pour objectif d'exploiter les ressources produites afin de créer des systèmes de désambiguïsation lexicale arabes, et ensuite d'étudier l'impact de la tâche de désambiguïsation lexicale sur la traduction automatique.

Première partie

Contexte du travail et état de l'art

1

Arabe et TAL

1.1 Introduction

De nombreux travaux de recherche s'intéressent au traitement automatique de la langue arabe. Cette langue présente deux caractéristiques importantes qui complexifient les traitements automatiques : l'agglutination et la non-voyellisation des mots. Les performances des applications et tâches classiques du domaine comme la reconnaissance automatique de la parole, le résumé automatique, la traduction automatique ou la désambiguïsation lexicale en pâtissent.

Nous présentons dans ce chapitre, les principales caractéristiques de la langue arabe, en montrant bien la richesse morphologique de cette langue, ainsi que quelques ressources et outils utiles pour son traitement automatique. Nous commençons, d'abord, par décrire la langue arabe d'une manière succincte, tout en précisant la différence entre l'arabe moderne standard et l'arabe dialectal. Nous évoquons, ensuite, certains concepts de base de la morphologie de l'arabe. Enfin, nous présentons ses différentes catégories grammaticales, avant d'exposer, à la fin de ce chapitre, quelques ressources linguistiques et outils existants.

1.2 Caractéristiques de la langue arabe

1.2.1 La langue arabe

Dans la langue arabe, nous trouvons l'arabe standard moderne (ASM) qui est la forme la plus utilisée et comprise de l'arabe dans le monde et qui apparaît dans des situations formelles et officielles (enseignement, discours religieux, émissions de radio et télévision, documents administratifs etc.) ainsi que l'arabe dialectal (AD) qui est la langue de la vie quotidienne et qui varie suivant la région et le pays (arabe maghrébin, arabe égyptien, arabe iraqien, arabe du Golf etc.). Toutefois, comme décrit dans la figure 1.1, ces deux variétés de la langue arabe sont totalement distinctes.

Dans ce travail, nous nous intéressons à l'arabe standard moderne qui est une langue internationale et intercontinentale, classée parmi les langues sémitiques, et parlée par plus de 263 millions de personnes. Comme il est illustré dans la figure 1.2 elle est la langue officielle dans 25 pays, et se place ainsi en troisième position après l'anglais et le français selon *Wikipedia* [1] .

1. https://fr.wikipedia.org/wiki/Arabe

Français ⇢ **Rami n'a pas acheté une table**

Arabe standard moderne ⇢ لم يشتر رامي طاولة

Arabe tunisien ⇢ رامي ماشراش طاولة

Arabe égyptien ⇢ رامي ماشتراش طربيزة

Arabe marocain ⇢ رامي ماشراش ميدة

FIGURE 1.1: Variation lexicale de l'arabe

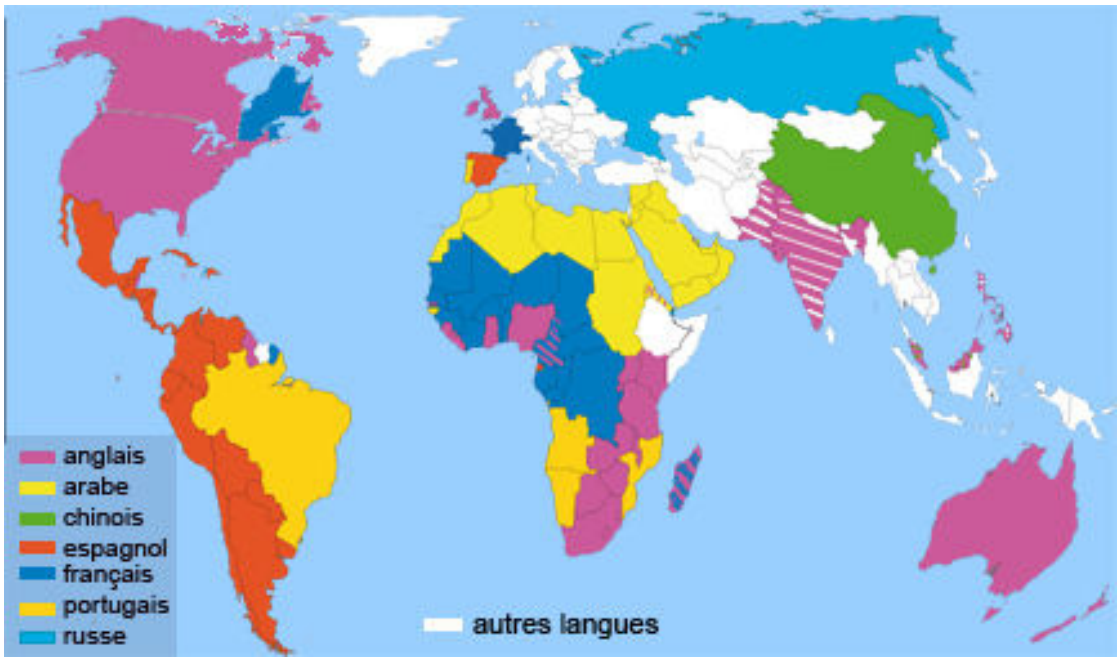

FIGURE 1.2: Situation de la langue arabe dans le monde RFI [2009]

C'est une langue qui s'écrit de droite à gauche, utilisant une écriture cursive, comportant 28 lettres dans son alphabet. Ces lettres sont des consonnes et des voyelles longues (ا «Alif», و «Waw» et ي «Ya»). Cependant, nous trouvons le «Hamza» (ء) qui peut prendre des formes différentes (آ، أ، إ، ا), des voyelles courtes 1.3 mises au dessus ou au dessous des lettres permettant leur articulation, et qui comportent : la «fatha» (le son A), la «kasra» (le son I), et la «damma» (le son OU). Le «soukoun» correspond à

l'absence de voyelle courte.

Soukoun [..]	بْ [b]	تْ [t]	ثْ [s]	جْ [j]	حْ [h]	نْ [n]	لْ [l]
Fatha [a]	بَ [ba]	تَ [ta]	ثَ [sa]	جَ [ja]	حَ [ha]	نَ [na]	لَ [la]
Damma [ou]	بُ [bou]	تُ [tou]	ثُ [sou]	جُ [jou]	حُ [hou]	نُ [nou]	لُ [lou]
Kasra [i]	بِ [bi]	تِ [ti]	ثِ [si]	جِ [ji]	حِ [hi]	نِ [ni]	لِ [li]

FIGURE 1.3: Voyelles courtes en arabe [Verschaere, 2016]

De plus, il y a le «Tanwine» qui est un deuxième type de voyelles permettant de vocaliser les consonnes, il marque la répétition de deux voyelles identiques à la fin d'un mot (nom), indiquant s'il s'agit d'un nom défini ou indéfini. Le «chadda» est une voyelle double qui indique le doublement de la consonne ou semi-consonne concernée. Ce symbole se met uniquement au dessus de la lettre, et s'écrit avec la «fatha», la «kasra», la «damma» ainsi que le «Tanouine», comme il peut être placé au dessus de la lettre sans la présence d'une voyelle courte.

Contrairement aux langues comme l'anglais ou le français, en arabe il n' y a pas de différence entre les lettres minuscules ou majuscules. Cependant, comme il est montré dans la figure 1.4, la forme d'écriture de certaines lettres arabes peut varier suivant leur emplacement dans le mot (au début, au milieu ou à la fin) et selon les lettres qui les entourent.

L'arabe standard moderne utilise des signes de ponctuation similaires à ceux utilisés dans les langues européennes (.,;, :,!, "", (), etc), mais il diffère légèrement en ce qui concerne quelques signes de ponctuation en raison de l'écriture de droite à gauche. Ainsi, le point d'interrogation inversé (؟) et la virgule inversée (،) sont inversés. Le Tatweel (ـ) est un symbole arabe particulièrement unique utilisé pour allonger les caractères sans changer le sens du mot (ex. نـال et نال)

L'arabe est considéré comme une langue peu dotée pour certaines applications et relativement difficile à traiter dans le domaine du traitement automatique du langage naturel. C'est une langue très riche en termes comportant plusieurs sens (termes poly-

nom	lettre	fin	milieu	début	phonétique	nom	lettre	fin	milieu	début	phonétique
alif	ا	ـا	ـا	ا	a	Dad	ض	ـض	ـضـ	ضـ	d
ba	ب	ـب	ـبـ	بـ	b	Ta	ط	ـط	ـطـ	طـ	t
ta	ت	ـت	ـتـ	تـ	t	Za	ظ	ـظ	ـظـ	ظـ	z
tha	ث	ـث	ـثـ	ثـ	th (= s)	'ayn	ع	ـع	ـعـ	عـ	'a
jim	ج	ـج	ـجـ	جـ	j	ghayn	غ	ـغ	ـغـ	غـ	gh (=r)
ha	ح	ـح	ـحـ	حـ	h	fa	ف	ـف	ـفـ	فـ	f
kha	خ	ـخ	ـخـ	خـ	Kha (= ra)	qaf	ق	ـق	ـقـ	قـ	q
dal	د	ـد	ـد	دـ	d	kaf	ك	ـك	ـكـ	كـ	k
dhal	ذ	ـذ	ـذ	ذـ	ð (= th)	lam	ل	ـل	ـلـ	لـ	l
ra	ر	ـر	ـر	رـ	r	mim	م	ـم	ـمـ	مـ	m
zay	ز	ـز	ـز	زـ	z	nun	ن	ـن	ـنـ	نـ	n
sin	س	ـس	ـسـ	سـ	s	ha	ه	ـه	ـهـ	هـ	ha
šin (shin)	ش	ـش	ـشـ	شـ	sh (= ch)	waw	و	ـو	ـو	و	w
Sad	ص	ـص	ـصـ	صـ	s	ya	ي	ـي	ـيـ	يـ	i (ou y)

FIGURE 1.4: Différentes formes des lettres arabes selon leur position dans un mot Verschaere [2016]

sémiques), ce qui peut poser de problèmes de compréhension voire même engendrer de grossières confusions notamment lorsque nous traitons des corpus arabes non voyellés. En effet, les voyelles jouent un rôle très important non seulement pour lever l'ambiguïté d'un mot arabe mais aussi pour identifier sa catégorie grammaticale indépendamment de sa position dans une phrase donnée. Elles permettent de distinguer les mots ayant la même représentation, comme il est montré dans le tableau 1.1. Debili et al. [2002] ont confirmé que les textes voyellés en arabe contiennent 43% de mots ambigus, tandis que les textes non voyellés peuvent en contenir 72%. Un mot arabe peut prendre un tout autre sens après changement d'une simple voyelle.

De plus, la richesse morphologique de la langue arabe peut causer des problèmes d'ambiguïté lors de la segmentation (tokenisation) lexicale d'un mot, ce qui influence la détermination de sons sens exact. Prenons l'exemple du mot «ورد» [wrd] qui a deux tokenisations possibles : une première qui le considère comme un seul mot «ورد» [warada] et une seconde qui considère que la lettre «و» [w] est un préfixe du mot «وَرَدَّ» [radda] .

Mot	Interprétation 1	Interprétation 2	Interprétation 3
شعر	شِعْرٌ (poésie)	شَعْرٌ (cheveux)	شَعُرَ (a senti)
كتب	كَتَبَ (a écrit)	كُتُبٌ (livres)	كُتِبَ (a été écrit)
بين	بَيَّنَ (a déclaré)	بَيِّنٌ (évident)	بَيْنَ (parmi)
ورد	وَرَدَ (est mentionné)	وَرْدٌ (roses)	وُرِدَ (a été mentionné)

TABLE 1.1: Exemples d'ambiguïté des mots arabes due à l'absence de voyellisation

1.2.2 Concepts de base de la morphologie de l'arabe

1.2.2.1 La racine

Chaque mot arabe est basé sur trois lettres (des consonnes) ou parfois quatre et très rarement deux (quelques mots spéciaux). Ces lettres constituent la racine (جذر) ou le cœur du mot, à laquelle nous pouvons ajouter d'autres lettres pour former des mots différents (lexèmes, plusieurs dérivations) appartenant au même champ sémantique. Nous trouvons environ 6 000 racines en arabe [2]. En effet, la racine est la partie la plus importante dans la définition de la signification principale d'un mot. Prenons l'exemple de la racine trilitère [ك + ت + ب] [k+t+b] qui désigne (signifie) l'idée générale (la notion) de l'écriture, à partir de laquelle plusieurs mots peuvent être dérivés comme :

— Écrivain, celui qui écrit كاتب [katib]
— Livre كتاب [kitab]
— Vous écrivez تكتبون [taktubuna]
— Nous écrivons نكتب [naktubu]
— Bibliothèque مكتبة [maktaba]
— Écriture كتابة [kitaba]
— Ce qui est écrit مكتوب [maktub]
— Bureau مكتب [maktab] ...

1.2.2.2 Le schème

Un mot arabe est issu d'une racine et intégré dans un schème (وزن) qui représente la forme du mot. Le schème arabe est de racine [ف + ع + ل], et permet de

2. https://fr.wiktionary.org/wiki/Cat%C3%A9gorie:Radicaux_en_arabe

déterminer la structure des mots. En associant une racine à un ou plusieurs schèmes, nous obtenons du nouveau vocabulaire. En arabe, nous trouvons quelques centaines de schèmes [3]. Ainsi, nous citons quelques exemples de schèmes appliqués sur la racine [ك + ت + ب] [k+t+b] :

— Le mot كتب [kataba] correpsond au schème فعل [faEala]
— Le mot كاتب [katib] correspond au schème فاعل [faEiL]
— Le mot مكتوب [maktub] correspond au schème مفعول [mafEul]
— Le mot كتاب [kitab] correspond au schème فعال [fiEal] ...

1.2.2.3 Les affixes

Les affixes (préfixes, suffixes) sont des lettres qui peuvent être ajoutées au début et/ou à la fin de la base d'un mot, utilisés afin d'identifier des traits grammaticaux. Ils marquent l'aspect, le mode du verbe, la personne, le genre ainsi que le nombre du sujet, etc.

1.2.2.3.1 Les préfixes

Les préfixes sont des morphèmes verbaux placés au début d'un mot afin de modifier son sens ou de former un autre mot. Nous présentons dans le tableau 1.2 des exemples de mots dérivés de la racine [ك + ت + ب] [k+t+b] avec des préfixes en arabe :

Préfixe	Signification	Mot
أ	J'écris	أَكْتُبُ
ت	Tu écris	تَكْتُبُ
ي	Il écrit	يَكْتُبُ
ن	Nous écrivons	نَكْتُبُ

TABLE 1.2: Exemples de mots avec préfixes en arabe

1.2.2.3.2 Les suffixes

Les suffixes sont des morphèmes (verbaux ou nominaux) qui se placent à la fin de

3. https://fr.wiktionary.org/wiki/Cat%C3%A9gorie:Radicaux_en_arabe

la base (verbale ou nominale) d'un mot. Nous présentons dans le tableau 1.3 quelques exemples de mots comportant des suffixes en arabe.

Suffixes	Signification	Mot
تُ	J'ai écrit	كَتَبْتُ
تَ	Tu as écrit	كَتَبْتَ
نا	Nous avons écrit	كَتَبْنَا
وا	Ils ont écrit	كَتَبُوا
ه	Son livre (Masculin)	كِتَابُهُ
ها	Son livre (Féminin)	كِتَابُهَا

TABLE 1.3: Exemples de mots avec suffixes en arabe

1.2.2.4 Les clitiques

Les clitiques (proclitiques et enclitiques) en arabe sont des unités linguistiques qui se prononcent et s'écrivent comme des affixes mais qui sont grammaticalement indépendants [Alotaiby et al., 2010]. En effet, un mot graphique arabe peut être décomposé en cinq éléments : proclitique(s), préfixe, base, suffixe(s) et enclitique(s).

1.2.2.4.1 Les proclitiques

En arabe, les proclitiques sont collés au mot qui le précède afin de former un nouveau mot. En effet, nous trouvons des proclitiques simples (des conjonctions, des prépositions, des coordonnants etc.) comportant une seule lettre et d'autres composés (une combinaison des premiers) comportant plus qu'une lettre. Nous présentons, ci-après, quelques exemples de proclitiques en arabe :

— L'article défini : ال
— L'outil d'interrogation : أ
— Les coordonnants : و، ف
— Les prépositions : ل، ب
— La lettre qui indique le futur : س

Le tableau 1.4 montre quelques exemples des proclitiques en arabe :

Proclitique(s)	Signification	Mot
ال	Le bureau	المكتب
أ	Est ce que tu as écrit	أكتبت
ف	Donc il a écrit	فكتب
و	Et il a écrit	و كتب
ل	Pour écrire	ليكتب
ب	Avec son livre	بكتابها
س	Il écrira	سيكتب

TABLE 1.4: Exemples de mots avec proclitiques en arabe

1.2.2.4.2 Les enclitiques

Les enclitiques en arabe représentent les lettres qui peuvent être attachées à la fin d'un mot (verbe, nom ou préposition). Tout comme les proclitiques, les enclitiques peuvent être combinés entre eux pour produire un autre mot composé. Le tableau 1.5 montre quelques exemples des enclitiques arabes existants :

Enclitique(s)	Signification	Mot
ي	Ils m'ont appris	علّموني
ك	Ils t'ont fait sortir	أخرجوك
كم	Ils vous ont écrit	كتبوكم

TABLE 1.5: Exemples de mots avec enclitiques en arabe

Nous présentons dans la figure 1.5 un exemple d'un mot agglutiné en arabe constitué d'une pré-base (préfixe et proclitique), d'une base, ainsi que d'une post-base (suffixe et enclitique).

1.2.2.5 Le radical

Dans la langue arabe il y a une différence entre le radical (جذع) et la racine. En effet, le radical ou le stem est la plus petite unité lexicale dérivée à partir d'une racine selon une forme donnée. Celui-ci peut être obtenu à partir d'un mot (forme agglutinée)

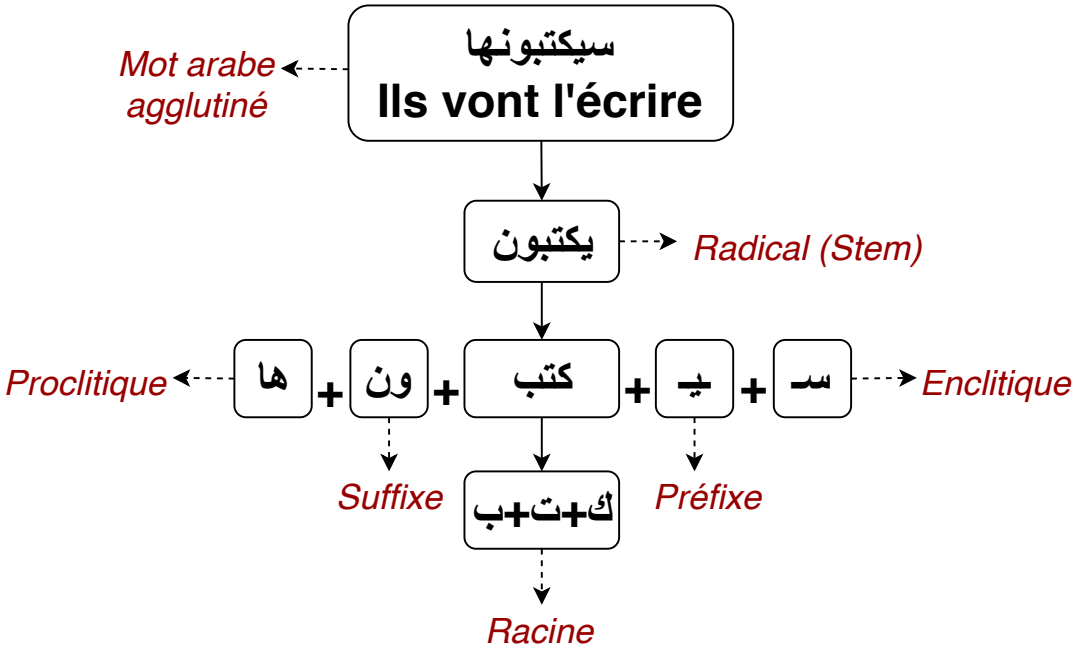

FIGURE 1.5: Structure d'un mot arabe agglutiné

après suppression des affixes (préfixes et ou affixes) qui l'entourent. Sachant que, en arabe, le radical des verbes trilitères se forme de trois lettres suivant le schème فَعَلَ. Plus précisément, prenons l'exemple précédent de la racine [ك + ت + ب] [k+t+b], à partir de laquelle nous pouvons générer différents radicaux tels que : le verbe كتب [kataba] (il a écrit), le verbe يكتب [yaktubu] (il écrit), le nom كاتب [katib] (écrivain), le nom كتاب [kitab] (livre) etc.

1.2.2.6 Le lemme

Le lemme est la forme d'entrée du dictionnaire. Il s'agit de la forme canonique (de base) d'un mot, obtenue après élimination des affixes (préfixes, infixe et suffixes) et application de règles autorisées par la classe grammaticale du mot :

- Troisième personne de singulier de l'accompli actif pour les verbes.
- Nominatif singulier pour les noms variables (masculin pour les noms qualificatifs) et forme classique voyellée pour les noms invariables (masculin singulier pour les pronoms).
- Forme classique voyellée pour les particules.

Par ailleurs, un radical est lui-même une forme fléchie et il est dérivé à partir d'un

lemme. En effet, une forme fléchie en arabe peut être composée de la manière suivante :

— Lemme + Conjonction
— Lemme + Préposition
— Lemme + Article défini
— Lemme + Article possessif

Nous citons ici quelques exemples de lemmes produits à partir de radicaux de la racine [ك+ ت+ ب] [k+t+b] :

— Le mot يكتبون [yaktubuna] (ils écrivent) a été dérivé du lemme كَتَبَ [kataba] (il a écrit)
— Le mot كُتُبٌ [kutub] (livres) a été dérivé du lemme كتاب [kitab] (livre)
— Le mot مكاتب [makateb] (bureaux) a été dérivé du lemme مكتب [maktab] (bureau)
— le mot مكتوبة [maktuba] (écrite) a été dérive du lemme مكتوب [maktub] (écrit).

1.2.3 Les parties du discours en arabe

Dans le lexique arabe, les mots sont divisés en trois types de catégories grammaticales :

— Nom : un élément qui indique un endroit, une personne ou une chose et qui peut être de genre masculin ou féminin et de nombre singulier, duel ou pluriel. Il peut être défini ou indéfini. En arabe, l'adjectif est une sous-classe du nom qui se place après celui qu'il qualifie, il correspond au genre et au nombre du nom qu'il suit et subit les mêmes règles de formation.
— Verbe : un mot auquel se relie directement ou indirectement les différents termes d'une phrase est considéré comme un verbe en arabe. Il réfère à une action ou un état d'être.
— Particule : une entité utilisée avec le nom et le verbe pour identifier un certain rôle dans la phrase, qui permet une lecture cohérente de celle-ci.

1.3 Ressources et outils de l'arabe

Contrairement à d'autres langues, les ressources linguistiques électroniques ainsi que les outils TAL libres de droit en arabe sont rares et peu développés, notamment les corpus arabes annotés en sens.

1.3.1 Bases lexicales

1.3.1.1 WordNet

Le *WordNet arabe* (AWN) [Black and ElKateb, 2004, Christiane et al., 2005, Elkateb et al., 2006] est une base de données lexicale disponible en ligne, qui suit la conception et le contenu de *Princeton WordNet* (PWN) [Miller, 1995] et qui contient 23 481 mots arabes (dont 21 833 mots traduits) et 11 269 synsets arabes (dont 10 127 synsets traduits par divers contributeurs). Les ensembles de synonymes sont reliés entre eux par des relations sémantiques (hyperonymie, antonymie, etc)

En effet, le *WordNet arabe* est aligné avec le *Princeton WordNet* ainsi qu'avec tous les WordNets qui sont alignés avec le PWN, soit directement, soit indirectement par le biais d'un index interlingual (ILI) et l'ontologie SUMO (Suggested Upper Merged Ontology) [Black et al., 2006].

1.3.1.2 BabelNet

BabelNet est un grand réseau sémantique multilingue en ligne, qui fournit des gloses (des entrées lexicalisées) dans plusieurs langues, parmi lesquelles la langue arabe. Il est construit en intégrant automatiquement *WordNet*, *Wikipedia*, *Wiktionary*, *Wikidata* ainsi qu'*OmegaWiki*. La partie arabe de la version 4.0 de *BabelNet*, comporte 2 942 886 *Babel synsets* et 4 267 782 sens de mots parmi lesquels 102 704 sens traduits de *WordNet* et 144 952 sens traduits de *Wikipedia*.

1.3.1.3 PropBank arabe

Le *Propbank* est un corpus annoté sémantiquement qui existait tout d'abord en anglais et chinois. Depuis 2009, l'état de l'art du TAL arabe a été enrichi par le Propbank arabe (APB) [Palmer et al., 2008, Zaghouani et al., 2010]. Ce corpus comporte des phrases annotées avec leurs structures prédicat-argument ainsi que leurs étiquettes de rôles sémantiques. Le *Propbank* arabe se compose de deux parties : des fichiers frames contenant les différents prédicats avec leurs arguments, et un corpus annoté syntaxiquement à l'aide de *Penn Treebank arabe* [Maamouri et al., 2004] pour annoter en rôles sémantiques les prédicats.

1.3.1.4 Arramooz AlWaseet

Arramooz AlWaseet [4] est un dictionnaire arabe libre de droit construit manuellement pour l'analyse morphologique. Il contient plus de 10 000 verbes, 40 000 noms, et une dizaines de particules et d'outils syntaxiques.

1.3.1.5 Wikipedia en arabe

Wikipedia est une encyclopédie en ligne libre, universelle, multilingue. Cependant l'édition de *Wikipedia* en arabe est encore pauvre et limitée (comportant 589 917 articles) par rapport à certaines langues comme l'anglais (5 721 476 articles), le français (2 042 399 articles), etc.

1.3.1.6 DIINAR.1

DIINAR.1 (DIctionnaire INformatisé de l'ARabe, version 1) [Dichy et al., 2002] est une ressource lexicale payante connue et utilisée dans le traitement automatique de l'arabe. Elle a été conçue et réalisée en commun entre l'IRSIT de Tunis, l'ENSSIB de France et l'université Lumière-Lyon 2. Elle est diffusée par ELRA-ELDA et elle contient au total 121 522 entrées voyellées, parmi lesquelles il y a 29 534 noms, 19 457 verbes ainsi que 70 702 dérivés nominaux.

1.3.2 Outils

1.3.2.1 MADAMIRA

L'analyseur morphologique MADAMIRA [Pasha et al., 2014] : est un système d'analyse morphologique et de désambiguïsation de l'arabe librement disponible qui exploite certains des meilleurs aspects des deux systèmes existants et les plus utilisés pour le traitement automatique de la langue arabe que sont : MADA [Habash and Rambow, 2005, Habash et al., 2009, 2013] et AMIRA [Diab, 2009]. En effet, MADAMIRA permet la tokenisation, la lemmatisation, la racinisation, l'étiquetage morpho-syntaxique, la désambiguïsation morphologique, la diacritisation, la reconnaissance des entités nommées, etc.

Comme décrit dans le tableau 1.6, MADAMIRA propose les deux schémas de tokenisation suivants :

4. http://arramooz.sourceforge.net/index.php?content=projects_en

— **ATB :** consiste à segmenter tous les clitiques (voir la section 1.2.2.4) sauf les articles définis, et à normaliser les caractères ALIF et YA en utilisant le caractère '+' comme marqueur de clitiques.
— **D3 :** consiste à tokeniser les proclitiques (voir la section 1.2.2.4.1) QUES, CONJ, les clitiques PART, ainsi que tous les articles et enclitiques (voir la section 1.2.2.4.2). En outre, il normalise les caractères Alif et Ya après la dévoyelisation des caractères arabes.

Segmentation	Séquence de mots
Brut	أنه سيتم الانتهاء من [...]
D3	ان +ه س+ يتم ال+ انتهاء من [...]
ATB	ان +ه س+ يتم الانتهاء من [...]

TABLE 1.6: Exemple de segmentation D3 et ATB d'une séquence de mots en arabe avec MADAMIRA

1.3.2.2 Alkhalil Morpho Sys

C'est un analyseur morphosyntaxique de mots arabes disponible en ligne. Le système peut traiter des textes non voyellés ainsi que des textes partiellement ou totalement voyellés. Pour chaque mot analysé il fournit, le stem (voir la section 1.2.2.5) voyellé, sa catégorie grammaticale, ses racines (voir la section 1.2.2.1) possibles associées aux motifs correspondants, ainsi que les proclitiques et enclitiques .

1.3.2.3 AraComLex

AraComLex, ou *Arabic Computer Lexicon*, est un transducteur morphologique à état fini pour l'arabe standard moderne, libre de droit, et qui contient plus de 30 000 lemmes arabes.

1.3.2.4 L'étiqueteur morphologique de Khoja

Dans leur étiqueteur morphologique arabe disponible en ligne, Khoja [2001, 2003] combine des techniques statistiques et des règles linguistiques permettant d'obtenir un taux de précision élevé. Les étiquettes utilisées sont issues du jeu d'étiquettes du *British National Corpus* (BNC), et adaptées afin d'intégrer certaines notions de la grammaire

arabe traditionnelle. Ils ont utilisé 131 étiquettes différentes qui sont classés en cinq catégories : nom, verbe, particule, résiduel et ponctuation. L'adjectif, le pronom, le nom propre, le nom commun ainsi que le numéral appartiennent à la catégorie nom. Pour entraîner le système, ils ont exploité un corpus de 50 000 mots issu du journal saoudien Al-Jazira, et ont obtenu 90% en précision.

1.4 Conclusion

Dans ce chapitre, nous avons présenté certaines caractéristiques et spécificités de la langue arabe, tout en citant des exemples qui montrent la distinction et la richesse de cette langue. C'est une langue agglutinante, ayant une taille de vocabulaire très importante. Nous avons montré que, la non voyellisation des mots arabes augmente l'ambiguïté de ces derniers. Ensuite, nous avons définit quelques éléments essentiels pour la morphologie de l'arabe tels que la racine, le schème, les affixes etc. En outre, nous avons présenté les trois classes grammaticales de l'arabe, que sont, le nom, le verbe ainsi que la particule. Enfin, nous avons passé en revue quelques ressources et outils utilisés pour le traitement automatique de l'arabe. Nous constatons qu'il existe très peu de ressources lexicales et notamment sémantiques libres de droit en arabe, ce qui fait que l'arabe peut être classé parmi les langues peu dotées pour certaines tâches de traitement automatique des langues ; tâches qui nécessitent une grande quantité de données lexicales ou de corpus annotés en sens, telle que la désambiguïsation lexicale.

2

Désambiguïsation lexicale et langues peu dotées

2.1 Introduction

Pour l'anglais, la disponibilité de ressources (sources de connaissance) ainsi que de grande quantité de données annotées manuellement en sens permet de produire des systèmes de désambiguïsation lexicale performants. De même, l'existence de corpus anglais de référence dédiés à cette tâche rend possible la comparaison des systèmes. Au contraire, cela demeure difficile pour la langue arabe. Ceci est lié, d'une part, au manque crucial de ressources lexicales et de corpus annotés en sens arabes pour pouvoir entraîner des systèmes de désambiguïsation lexicale, et d'autre part, à la non disponibilité d'un corpus de référence auquel se comparer.

Dans ce chapitre, nous présentons d'abord, la tâche de désambiguïsation lexicale, son processus de mise en œuvre, les ressources génériques utiles, les approches existantes pour la mise en œuvre, ainsi que les deux méthodes possibles pour évaluer un système de désambiguïsation lexicale. Ensuite, nous décrivons plus particulièrement la désambiguïsation lexicale de l'arabe et son état par rapport aux autres langues. Aussi, nous proposons un rappel bibliographique sur les méthodes appliquées à la tâche en question. Nous clôturons ce chapitre par une brève étude sur l'apport de la désambiguïsation lexicale en traduction automatique.

2.2 Désambiguïsation lexicale

La tâche de désambiguïsation lexicale (*Word Sense Disambiguation*) consiste à trouver pour chaque mot d'un texte le sens le plus approprié parmi un inventaire de sens pré-défini. Par exemple, dans la phrase « *Je vois la montagne à travers ma fenêtre.* », l'algorithme devrait choisir le sens de ‹‹fenêtre›› qui correspond à la **menuiserie** plutôt que celui qui correspond à l'**interface graphique**.

Dans cette tâche, seul l'anglais peut être considéré comme une langue réellement dotée. En effet, deux types de ressources qui demandent un travail humain particulièrement important peuvent être utilisées pour créer un système de désambiguïsation lexicale : les ressources basées sur des savoirs (dictionnaires, bases lexicales…) et les corpus annotés en sens. Les langues peu dotées disposent rarement de ces ressources.

Dans cette section, nous présentons la tâche de désambiguïsation lexicale et les ressources qui lui sont nécessaires : des bases lexicales créées manuellement ou automatiquement ainsi que des corpus annotés essentiellement créés manuellement. Nous montrons que la rareté ou la libre disponibilité de ces derniers pour l'arabe complique non seulement la création de systèmes de désambiguïsation lexicale pour cette langue mais empêche surtout la comparaison des systèmes entre eux.

2.2.1 Processus de mise en œuvre

Trois étapes sont nécessaires pour mettre en place une désambiguïsation lexicale automatique [Schwab, 2017].

1. *Constitution d'une ressource générique* : plusieurs ressources non dédiées à la désambiguïsation lexicale sont possibles : dictionnaires, encyclopédies, corpus non annotés, corpus annotés, bases lexicales... Cette étape optionnelle est souvent réalisée par des équipes spécialisées.

2. *Constitution d'une ressource dédiée à la désambiguïsation lexicale* : utilisation d'une ou plusieurs ressources brutes pour donner une représentation informatique à chacun des sens d'un mot, il s'agit ainsi de constituer une ressource dédiée à la tâche. Ces sens sont définis par l'expertise humaine ou induits à partir des contextes d'utilisation dans les textes.

3. *Utilisation de la ressource dédiée pour désambiguïser des textes* : il s'agit de l'algorithme de désambiguïsation proprement dit. Plusieurs facteurs peuvent entrer en compte. Certains sont communs à chaque algorithme comme la taille du contexte considéré pour le mot à désambiguïser (par exemple quelques mots avant ou après celui-ci, la phrase qui le contient, voire le texte) tandis que d'autres dépendent du type d'algorithme : par exemple la limite à considérer pour la profondeur de la recherche dans un graphe ou encore les paramètres à considérer pour des algorithmes stochastiques.

Ainsi, selon cette méthodologie, Schwab et al. [2013] utilisent WordNet comme ressource générique ; des représentations en sac de mots issues des définitions des sens et de leurs liens comme ressource dédiée ; un algorithme à colonies de fourmis et une mesure de proximité entre les sacs de mots comme algorithme de désambiguïsation lexicale.

De même, Navigli and Ponzetto [2012] utilisent de nombreux corpus, WordNet et Wikipedia pour constituer une ressource générique, *BabelNet*, que nous présenterons plus en détail à la section 2.2.2.1.2. À partir de cette ressource, ils construisent un grand réseau lexical dédié : un graphe dont ils exploitent la structure pour désambiguïser des textes.

2.2.2 Ressources génériques utiles

En désambiguïsation lexicale, deux types de ressources génériques sont importantes : des corpus manuellement annotés par des sens et des sources de connaissances. Les campagnes d'évaluation sur l'anglais ont globalement montré une corrélation directe entre la quantité/qualité de corpus annoté et la qualité du système final (*SemEval 2007 task 07* [Navigli et al., 2007]).

Dans le processus d'informatisation d'une langue, avant de pouvoir construire un corpus annoté manuellement par des sens, il faut disposer d'un inventaire de sens. Aucune autre langue que l'anglais ne bénéficie d'autant de corpus de textes manuellement annotés par des sens ainsi que de connaissances lexicales. La figure 2.1 illustre l'état des ressources disponibles pour la désambiguïsation lexicale (librement accessibles) pour un certain nombre de langues. Il est donné pour que le lecteur puisse se faire une idée de la situation actuelle. Un recensement plus précis est difficile à obtenir et il faut donc interpréter les positions des langues les unes par rapport aux autres plutôt que de manière absolue, sauf pour l'anglais que nous avons placé le plus en haut à droite. Si nous pouvons considérer que la quantité de données annotées est un paramètre quantifiable (par exemple en nombre moyen d'occurrences par terme du lexique), la richesse des sources de connaissances disponibles est, elle, plus floue. C'est en particulier le cas entre deux langues différentes puisque la taille de leur vocabulaire est différente. Il faut noter également que certaines langues peuvent bénéficier de données provenant d'autres langues par des alignements (comme c'est le cas dans *BabelNet*, par exemple).

2.2.2.1 Bases lexicales

2.2.2.1.1 WordNet

Avant les années 1990, la désambiguïsation lexicale de l'anglais n'était pratiquement réalisée qu'à partir de dictionnaires électroniques. Le *Princeton WordNet* [Miller, 1995], initié au milieu des années 1980, a permis la mise à disposition d'une ressource

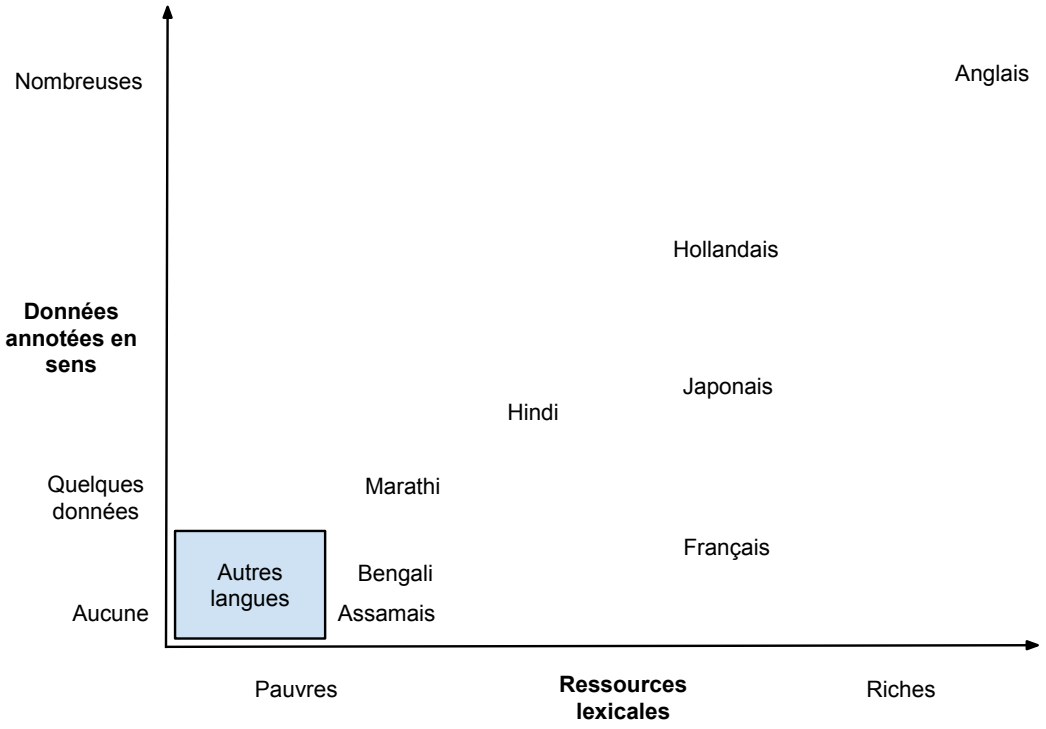

FIGURE 2.1: Données disponibles pour la désambiguïsation lexicale en fonction de la langue [Schwab, 2017]

utilisable librement. C'est une base de données lexicale de l'anglais libre de droit qui s'organise autour de la notion d'ensemble de synonymes nommés (*synsets*) et elle est devenue rapidement très populaire et a rapidement conduit à la disparition de l'usage des dictionnaires électroniques en désambiguïsation lexicale. Le *Princeton WordNet* est une base de données lexicale organisée autour de la notion d'ensemble de synonymes nommés (*synsets*) décrits par une partie du discours (nom, verbe, adjectif, adverbe), une définition et leurs liens (hyperonyme, hyponyme, antonyme, ...). Chaque sens d'une entrée lexicale correspond à un *synset*. La version courante du *Princeton WordNet*, la 3.0, comprend 155 287 items lexicaux pour un total de 117 659 *synsets*. Des versions pour d'autres langues existent mais, faute de moyens humains équivalents, leur qualité et couverture sont inférieures à celle de l'anglais. Bien souvent, les mots de ces langues sont décrits grâce à des *synsets* du *Princeton WordNet* anglais. C'est le cas du *WordNet arabe* [Elkateb and Fellbaum, 2006, Abouenour et al., 2013] qui contient 11 269 *synsets* arabes (dont 10 127 *synsets* traduits par divers contributeurs) ainsi que 23 481 mots arabes (dont 21 833 mots traduits). La *Global WordNet Association* établit la liste des wordnets existants [1].

2.2.2.1.2 BabelNet

Même si nos travaux ne l'exploitent pas directement [2], il est difficile de nos jours de ne pas évoquer BabelNet [Navigli and Ponzetto, 2012]. Il s'agit une ressource lexicale à grande échelle construite par alignement automatique des *synsets*, issus de *Princeton WordNet* et de pages Wikipedia correspondantes. BabelNet introduit la notion de *Babel Synset*, qui contient tout le contenu du *synset* correspondant dans le *Princeton WordNet*, ainsi qu'un ensemble de pages Wikipedia similaires. Cette correspondance entre *synsets Princeton WordNet* et pages Wikipédia se fait par un algorithme de désambiguïsation automatique. Les pages Wikipédia reliées par des hyperliens internes à Wikipédia ainsi que les articles associés dans les autres langues disponibles dans Wikipedia sont liés aux pages correspondantes. Pour toutes les pages dans les autres langues, si il n'y a pas de définition disponible ou extraite de la page, la définition anglaise du *Princeton WordNet* ou un extrait venant de *SemCor* est traduit par *Google Translate* pour servir de définition.

1. http://globalwordnet.org/wordnets-in-the-world/
2. BabelNet intégrant le *Princeton WordNet* et l'*WordNet arabe* grâce à l'*Open Multilingual Wordnet [Bond and Foster, 2013]*, on peut considérer qu'exactement la même expérience aurait pu être menée avec une sous-partie de BabelNet.

BabelNet, dans sa dernière version publiée en août 2016, la 3.7[3], comprend 271 langues, 13 801 844 *Babel synsets*, 745 859 932 sens, 380 239 084 relations lexico-sémantiques et 40 709 194 définitions textuelles. Pour l'anglais il y a 11 769 205 entrées, 6 667 855 *Babel synsets*, 17 265 977 sens de mots et un degré de polysémie de 2,59. Pour le français, il y a 5 301 989 entrées, 4 141 338 *Babel synsets*, 7 145 031 sens de mots et un degré de polysémie de 1,73.

2.2.2.2 Corpus annotés

Selon Benoit Habert, « *un corpus est une collection de données langagières qui sont sélectionnées et organisées selon des critères linguistiques et extra-linguistiques explicites pour servir d'échantillon d'emplois déterminés d'une langue* » [Habert et al., 1998]. Généralement, un corpus contient jusqu'à une douzaine de millions de mots et peut être lemmatisé et annoté avec des informations concernant les parties du discours. Parmi ces corpus, nous trouvons le *British National Corpus* [Burnard, 1998] (100 millions de mots) et le *American National Corpus* [Ide and Macleod, 2001] (20 millions de mots). Les textes proviennent de diverses sources comme des journaux, des livres, des encyclopédies ou du Web.

2.2.2.2.1 Exemples de corpus annotés

En désambiguïsation lexicale, plusieurs corpus annotés en sens sont utilisés. Nous pouvons citer, par exemple :

1. La *Defense Science Organisation* [Ng and Lee, 1996] a produit un corpus non disponible librement. 192 800 mots ont été annoté avec des *synsets* du *Princeton WordNet*. L'annotation se concentre sur 121 noms (113 000 occurences) et 70 verbes (79 800 occurences) qui ont été choisis parmi les plus fréquents et les plus ambigus de l'anglais. Selon les auteurs, la couverture correspond à environ 20% des occurrences de noms et de verbes en anglais.
2. Le *SemCor* [Miller, 1995] est un sous-ensemble du Corpus de Brown [Francis and Kučera, 1964]. Sur les 700 000 mots de ce dernier, environ 230 000 sont annotés avec des *synsets* du *Princeton WordNet*. L'annotation porte au total sur 352 textes. Pour 186 d'entre eux, 192 639 mots (soit l'ensemble des noms, verbes,

3. L'ensemble de ces statistiques vient de la page `http://babelnet.org/stats` consultée le 23/09/2017

adjectifs et adverbes) sont annotés. Sur les 166 autres, seulement 41 497 verbes sont annotés.

3. Les corpus issus des campagnes d'évaluation. Depuis 1998, il y a eu plusieurs campagnes (semeval-senseval) destinées à évaluer la désambiguïsation lexicale. La plupart ont concerné l'anglais mais également le japonais, l'espagnol, le chinois ou le français. La taille de ces corpus est de l'ordre d'une centaine de fois plus petite que celle des deux précédents corpus, soit quelques milliers de mots.
4. UFSAC (*Unification of Sense Annotated Corpora and Tools*) est une ressource récemment rendue disponible [Vial et al., 2017]. Elle regroupe l'ensemble des corpus annotés en anglais disponibles avec des sens du *Princeton WordNet* 3.0, uniformisés lorsque les droits le permettent et le code source pour construire l'ensemble des corpus à partir des données originales. Dans les recherches décrites dans ce manuscrit, nous exploitons l'ensemble des 12 corpus d'UFSAC (voir tableau 4.4).

Ressource	Phrases	Mots		Parties du discours annotées			
		Total	Annotés	Noms	Verbes	Adjectifs	Adverbes
SemCor	37 176	778 587	229 533	87 581	89 051	33 752	19 149
DSO	101 004	2 705 190	176 197	105 245	70 952	0	0
WNGT	117 659	1 634 691	496 776	287 798	77 234	107 135	24 609
MASC	31 760	585 354	113 546	49 474	39 356	12 894	11 822
OMSTI	820 084	35 800 061	920 357	476 692	253 555	190 110	0
OntoNotes	124 851	2 475 926	233 616	79 765	153 851	0	0
SemEval 2007 task 07	245	5 637	2 261	1 108	591	356	206
SemEval 2007 task 17	126	3 438	455	159	296	0	0
SemEval 2 013 task 12	306	8 142	1 644	1 644	0	0	0
SemEval 2015 task 13	138	2 637	1 053	554	251	166	82
Senseval 2	238	5 589	2 301	1 061	541	422	277
Senseval 3 task 1	300	5 507	1 957	886	723	336	12
Total	1 233 649	44 010 759	2 179 696	1 091 967	686 401	345 171	56 157

TABLE 2.1: Informations relatives aux corpus UFSAC-eng Vial et al. [2017]

2.2.2.2.2 Difficultés liées à la construction d'un corpus annoté

Il existe peu de données manuellement annotées. La *Global WordNet Association* dresse la liste des 26 corpus annotés avec un wordnet[4]. Ces corpus concernent 17

4. http://globalwordnet.org/wordnet-annotated-corpora/ consultée le 23 septembre 2017. Il existe d'autres corpus annotés avec des *synsets* de wordnets comme ces corpus de domaines annotés avec des *synsets* de l'*Hindi Princeton WordNet* http://www.cfilt.iitb.ac.in/wsd/annotated_corpus/

langues. Seules trois d'entre elles (l'anglais, le hollandais et le bulgare) atteignent les 100 000 annotations. À notre connaissance, il existe assez peu de données annotées en sens pour le français (environ 3 600 mots annotés avec le dictionnaire Larousse pour la campagne Romanceval 1998 et 1 656 mots annotés avec des sens de BabelNet pour la tâche 12 de la campagne SemEval 2013) et pour l'arabe (32 000 du AQMAR 1.0 Arabic Wikipedia Supersense Corpus développé par Schneider et al. [2012] qui est un corpus multi-domaines qui contient 65 000 mots issus de 28 articles de Wikipédia arabe, annotés à la main par des supersens propres à ce corpus. OntoNotes Release 5.0 propose également des annotations sur environ 13 000 mots parmi 300 000 issus de corpus journalistiques). Cette langue qui nous intéresse plus particulièrement est donc peu dotée en ce domaine.

La construction d'un corpus manuellement annoté en sens est réputée être une tâche très difficile par comparaison à d'autres tâches d'annotation. En effet, s'il n'y avait que 45 annotations possibles pour le *Penn Treebank* [Marcus et al., 1993], un corpus annoté en parties du discours, il y en a autant que de *synsets* (117 000) pour une annotation en sens issus du *Princeton WordNet*. Ainsi, pour l'annotation du corpus de la *Defense Science Organisation*, alors que les conditions étaient plus favorables que celles du *SemCor* (uniquement 191 mots différents pour seulement 1 800 annotations possibles), le taux d'annotation était seulement de 150 à 250 mots par heure (1 homme-année pour les 192 800 occurrences de mots) tandis que les annotateurs du *Penn Treebank* réalisaient 6 000 annotations par heure. Dans de telles conditions, nous comprenons mieux pourquoi assez peu de corpus annotés existent. Des recherches ont visé à faciliter cette annotation. Par exemple, pour le hollandais, Vossen et al. [2011] utilisent un algorithme de désambiguïsation automatique dont les annotations les moins sûres sont vérifiées/-modifiées manuellement par les annotateurs et Mihalcea and Chklovski [2003] utilisent des tâches de production participative (*crowdsourcing*) pour augmenter le nombre d'annotateurs.

Le même principe est utilisé par Taghipour and Ng [2015], qui annotent une grande quantité de textes provenant de corpus parallèles anglais-chinois grâce à un système de désambiguïsation lexicale utilisant les mots de la traduction alignés comme source principale (projection interlingue d'annotations). La qualité du corpus ainsi généré est ensuite démontré via l'amélioration d'un système de desambiguïsation lexicale supervisé entraîné sur ce même corpus.

Malheureusement, ces techniques restent possibles uniquement lorsque la désambi-

guïsation est de bonne qualité ce qui n'est pas le cas pour les langues pour lesquelles il n'existe pas de corpus annotés.

2.2.3 Approches pour la mise en œuvre

Il existe différentes méthodes de désambiguïsation lexicale, parmi lesquelles on peut citer les méthodes basées sur les similarités entre les sens ou encore les méthodes supervisées. La table 2.2 les présente toujours en fonction des axes ressources lexicales et corpus annotés en sens. Nous avons choisi de travailler avec les méthodes de désambiguïsation lexicale supervisées. Le principe issu de l'apprentissage automatique, consiste à entraîner un classifieur pour chaque mot cible, afin de prédire le sens le plus vraisemblable suivant son contexte. En effet, ces approches ont de bonne performance et elles fournissent de bons résultats dans les évaluations des systèmes de désambiguisation lexicale. En revanche, l'utilisation d'une méthode supervisée a besoin de grands corpus annotés en sens pour être entrainés et ceux-ci n'existent que dans peu de langues comme l'anglais, le japonais etc. L'arabe et le français sont considérées comme des langues peu dotées pour la désambiguïsation lexicale.

2.2.3.1 Algorithmes supervisés

Les algorithmes supervisés utilisent des techniques d'apprentissage automatique. Ils apprennent un classificateur sur les corpus annotés en sens en utilisant des classificateurs classiques : séparateurs à vaste marge (NUS-PT [Chan et al., 2007]), classificateurs naïfs bayésiens (NUS-ML, [Cai et al., 2007]), combinaison de séparateurs à vaste marge, entropie maximale (LCC-WSD, [Novischi et al., 2007]). Nous ne pouvons pas vraiment affirmer que tel ou tel classificateur est meilleur qu'un autre (avant l'émergence des réseaux de neurones) et ce qui différencie les performances des systèmes est principalement directement lié à la taille des données annotées. Bien que LCC-WSD et NUS-ML utilisent uniquement SemCor, NUS-PT utilise à la fois SemCor et le DSO.

2.2.3.2 Algorithmes non supervisés (induction de sens) et méthodes faiblement supervisées

De notre point de vue, les algorithmes non supervisés constituent leur ressource générique uniquement à partir de corpus non annotés en sens. Cette étape, appelée également induction des sens des mots utilise des techniques d'apprentissage par machine

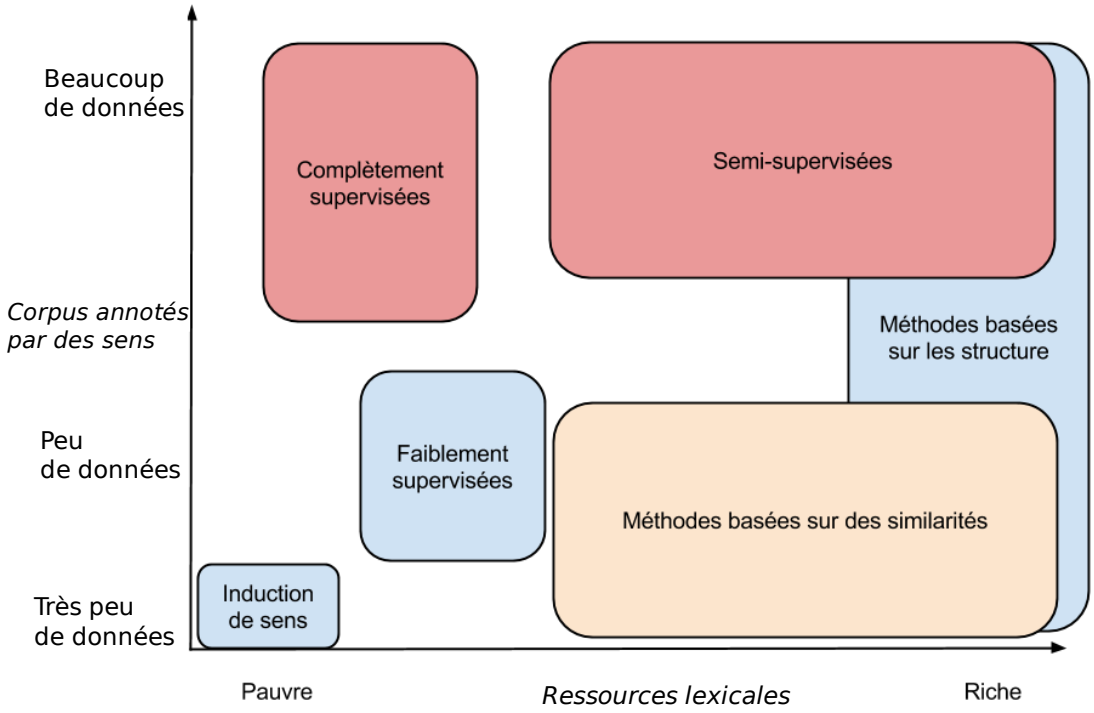

FIGURE 2.2: Les différentes méthodes de désambiguïsation lexicale [Schwab, 2017]

sur les corpus non annotés en sens sans avoir de connaissance *a priori* sur la tâche. Ces algorithmes induisent les sens des mots en considérant les co-occurrences suivant l'hypothèse distributionnelle [Harris, 1954] pour laquelle deux mots sont considérés comme sémantiquement proches (similaires) s'ils sont utilisés dans les mêmes contextes. Les principales techniques de regroupement comprennent : l'identification de structures particulières dans les graphes de co-occurrence [Véronis, 2003] et le regroupement de vecteurs construits à partir du contexte (voisins ou voisins des vecteurs voisins).

Une fois l'étape de regroupement réalisée, il s'agit d'une désambiguïsation lexicale classique en fonction de la représentation calculée obtenue.

2.2.3.3 Algorithmes basés sur les similarités

Les algorithmes basés sur des similarités sont composés à la fois d'un algorithme local (rarement plusieurs) et d'un algorithme global. Les algorithmes locaux correspondent aux mesures de proximité sémantique et permettent d'estimer la proximité entre deux sens de mots du texte. On s'attend, par exemple, à ce que ‹*professeur*› et

‹*enseignant*› soient évalués comme plus proches que ‹*ciel*› et ‹*trottoir*›. Une littérature entière est consacrée à ces mesures et on peut citer parmi elles, les mesures vectorielles comme dans LSA [Deerwester et al., 1990] ou Word2Vec [Mikolov et al.] comme celle de Lesk [Lesk, 1986] qui mesure le nombre de mots entre les définitions des sens correspondants.

L'approche globale, elle, propage ces mesures locales aux niveaux supérieurs (syntagmes, phrases, paragraphes, voire le texte, selon l'algorithme choisi) afin de désambiguïser l'ensemble du texte. Parmi les algorithmes globaux on trouve, entres autres, des algorithmes génétiques [Gelbukh et al., 2003], de recuit simulé [Cowie et al., 1992], à colonies de fourmis [Schwab et al., 2011] [Schwab et al., 2012] ou encore depuis peu, des algorithmes à colonies d'abeilles [Abualhaija and Zimmermann, 2016] ou de coucous [Vial et al., 2016].

2.2.3.4 Algorithmes basées sur les structures

Ces algorithmes reposent sur la topologie, la structure de grands graphes lexicaux. L'exemple typique de cette catégorie sont les travaux de Roberto Navigli exploitant BabelNet que nous avons présenté dans la section 2.2.2.1.2. La ressource dédiée à la désambiguïsation est fabriquée à partir des liens issus de cette grande base lexicale, elle-même constituée de très nombreuses ressources (autres bases lexicales, corpus annotés en sens). Pour des raisons de calculs, la désambiguïsation se fait souvent dans le contexte de la phrase et l'idée de base est de construire un nouveau graphe à partir de ses mots. Tandis que les systèmes [Navigli and Ponzetto, 2012] sont assez bons, avec une quantité importante de ressources, ces méthodes semblent limitées pour les langues qui ont moins de ressources et en particulier les langues pauvres en ressources annotées et pour lesquelles la désambiguïsation ne peut se faire que par l'intermédiaire des autres langues. À notre connaissance, il existe assez peu de travaux utilisant BabelNet pour d'autres langues que l'anglais et aucun sur l'arabe.

2.2.4 Évaluation de la désambiguïsation lexicale

Pour évaluer un système de désambiguïsation lexicale, deux approches sont possibles :

— ***In vivo*** : les systèmes sont évalués en fonction de leur contribution à la performance d'une application particulière (traduction automatique, recherche d'infor-

mation, *etc*.). Ce type d'évaluation est assez lourd à mettre en place et n'a encore jamais été utilisé, à notre connaissance, dans les campagnes d'évaluations.

— ***in vitro*** : les systèmes sont évalués en utilisant des corpus annotés de référence. En particulier ceux des campagnes d'évaluation Semeval-SensEval qui existent depuis une vingtaine d'années.

Avec cette dernière approche, les mesures classiques sont la précision P, le rappel R et le score F1 qui correspond à la moyenne harmonique de P et R. La précision se définit comme :

$$P = \frac{annotés \quad correctement}{total \quad annotés} \tag{2.1}$$

le rappel :

$$R = \frac{annotés \quad correctement}{total \quad à \quad annoter} \tag{2.2}$$

le score F1 :

$$F1 = \frac{2 * P * R}{P + R} \tag{2.3}$$

2.3 Désambiguïsation lexicale de l'arabe

2.3.1 État de la langue arabe pour la désambiguïsation lexicale

Comme mentionné précédemment, l'approche de désambiguïsation lexicale exige l'existence des grandes ressources lexicales. Cependant, comme illustré dans la figure 2.3, il y a un manque crucial de corpus manuellement annotés en sens libre de droit pour la plupart des langues notamment pour l'arabe. Ce type de corpus n'existaient jusqu'à présent que pour trois langues (anglais, japonais, bulgare).

Dans ce travail, nous visons à améliorer l'état de la langue arabe au niveau de la disponibilité des corpus annotés en sens. Ainsi, nous allons fabriquer des ressources lexicales en arabe, en utilisant une méthode de traduction automatique et de transfert direct des annotations qui nous permet d'obtenir des corpus annotés en sens dans une langue disposant d'un système de traduction d'une langue source riche en corpus annotés comme l'anglais vers une langue cible peu dotée (ici l'arabe). Pour ce faire, nous avons besoin des corpus parallèles bilingues afin de construire un système de traduction automatique, un système de désambiguïsation lexicale supervisé, ainsi qu'un corpus sémantiquement annoté de référence pour évaluer la désambiguïsation lexicale.

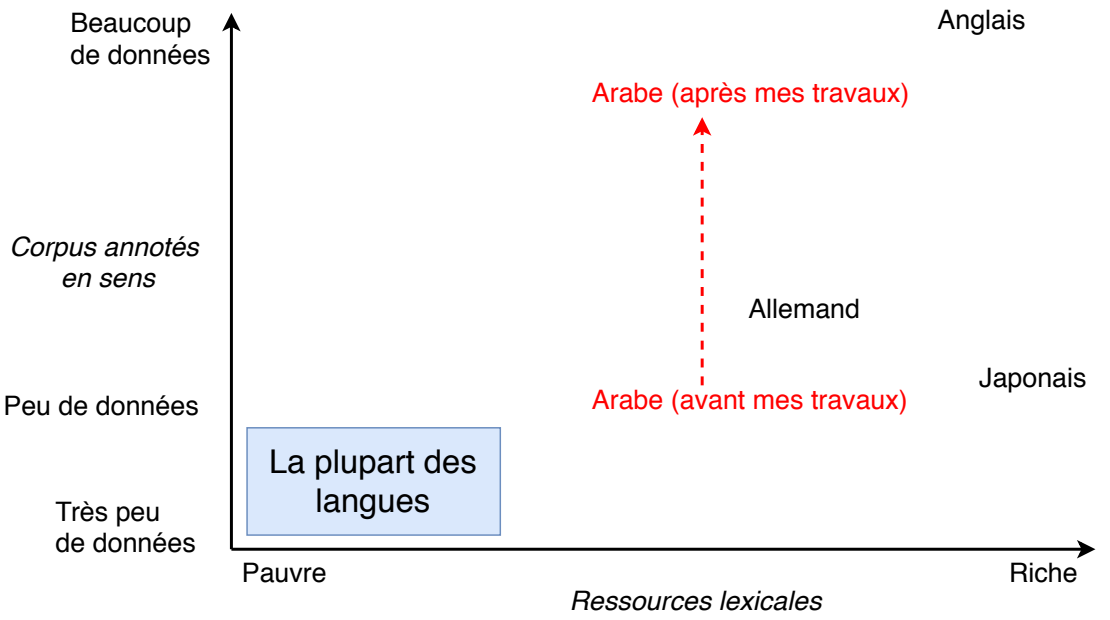

FIGURE 2.3: Ressources nécessaires à la désambiguïsation lexicale disponibles pour la langue arabe

Par ailleurs, les diacritiques manquantes dans les textes arabes est la caractéristique la plus difficile pour la désambiguïsation lexicale, car elle augmente le nombre de sens possibles d'un mot et rend la tâche de désambiguïsation plus difficile. De nombreuses publications sur la désambiguïsation lexicale de l'arabe considèrent souvent que la tâche est compliquée [Diab, 2004, Elmougy et al., 2008] à cause de certaines caractéristiques de la langue arabe que nous avons exposées dans le premier chapitre.

2.3.2 Méthodes de désambiguïsation lexicales appliquées à l'arabe

La littérature sur la désambiguïsation lexicale pour la langue arabe est relativement foisonnante. Nous ne cherchons pas ici à en faire un rapport exhaustif mais à présenter les principales approches et leurs limites. Parmi les travaux proposés pour faire face au problème de la désambiguïsation lexicale de l'arabe, nous pouvons citer le travail de Elmougy et al. [2008] qui ont utilisé un algorithme de racinisation ainsi qu'un classifieur bayésien naif afin de désambiguïser des mots non voyellés en arabe. Ils ont appliqué des étapes de prétraitement telles que la suppression des mots vides et la racinisation des mots à l'aide de l'algorithme de Al-Serhan et al. [2003]. Ils ont prédifini un ensemble de mots ambigus ainsi que les sens de chacun de ces derniers. Ils ont utilisé un dictionnaire

et ont collecté l'ensemble de leurs données d'entraînement à partir du Web. Pour chaque mot ambigu ils ont recueilli 10 échantillons d'entraînement et 10 échantillons de test pour l'évaluation. En utilisant l'algorithme de racinisation ils ont amélioré la précision de 53% (de 20% sans racinisation à 73% avec racinisation).

Eid et al. [2010] ont utilisé le classifieur Rocchio pour la désambiguïsation lexicale arabe. Ils ont comparé la performance de cette méthode de classification avec d'autres algorithmes d'apprentissage supervisé tels que le sens le plus fréquent (MFS), le classifieur bayésien naif (NBC) ainsi que la machine à vecteurs de support (SVM) afin de prouver son efficacité pour la désambiguïsation lexicale. Dans leur expériences, les auteurs ont utilisé un corpus lexical (issu de la littérature) de 5 noms arabes, ayant chacun 2 ou 3 sens. Les résultats ont montré que le classificateur Rocchio a atteint un taux de précision de 88% par rapport au NBC (86%), SVM (82%) et MFS (57,5%) avec une meilleure performance en 3 mots sur 5 pris en compte.

Zouaghi et al. [2011] ont utilisé le *WordNet arabe* (AWN) comme ressource générique. La ressource élaborée à partir d'AWN implique un sac des mots de la définition et le graphe des relations entre *synsets* afin de calculer diverses mesures de similarité classiques [Lesk, 1986, Resnik, 2011] pour effectuer la désambiguïsation lexicale.

Dans leur travaux, Merhben et al. [2012] utilisent des corpus non annotés en sens et quelques annotations de sens du dictionnaire *Lissan al arab* comme ressource générique. Les annotations sont ensuite utilisées comme bootstrap pour construire la ressource élaborée sous la forme de classifieurs (algorithme Naïve Bayes, listes de décisions, ...). La ressource élaborée est ensuite utilisée pour annoter de nouvelles parties du corpus. Le processus est répété jusqu'à ce qu'aucune nouvelle partie du corpus ne soit non-annotée.

Par ailleurs, Diab [2004] a présenté et évalué une approche non supervisée de désambiguïsation lexicale, nommée SALAAM (Sense Annotations Leveraging Alignments And Multilinguality). Cette méthode consiste à annoter les mots arabes avec leurs sens à partir du WordNet anglais en utilisant un corpus parallèle arabe-anglais basé sur des correspondances de traduction entre mots arabes et anglais. Ils ont crée un corpus de test, en traduisant un ensemble de corpus anglais annotés manuellement en sens avec WordNet v.1.7, vers l'arabe à l'aide des deux systèmes de traduction automatique existants (Tarjim et Almisbar), en fusionnant les résultats des deux sorties des systèmes et en portant les annotations à l'aide de l'outil d'alignement Giza++ [Och and Ney, 2003]. L'approche utilisé a atteint 56,9% en termes de précision, évaluée sur des mots arabes

(1071 noms) annotés en sens.

Toutefois, les ensembles de données utilisés dans les travaux que nous avons cités ne sont ni disponibles ni standardisés comme référence utile pour l'évaluation des systèmes de désambiguïsation lexicale arabes. Les auteurs ont testé leurs approches en utilisant leurs propres données (dictionnaire, corpus, etc.), ce qui empêche l'étude comparative entre toutes les approches. Ainsi le problème principal pour réaliser une désambiguïsation lexicale efficace pour l'arabe reste le manque de corpus annotés en sens.

2.4 Apport de la désambiguïsation lexicale en traduction automatique

La tâche de désambiguïsation lexicale peut permettre d'améliorer diverses applications telles que la traduction automatique, la recherche d'information, le traitement de texte, etc.
La traduction automatique consiste à traduire un texte rédigé dans une langue source vers un texte rédigé dans une langue cible tout en passant par une étape de compréhension de la langue source avant de procéder à sa traduction. De ce fait, un système de désambiguïsation lexicale peut jouer un rôle important pour lever l'ambiguïté des mots (ayant plusieurs sens) que ce soit ceux de la langue source ou encore ceux de la cible. Par exemple, le mot français «Vouloir» peut avoir diverses significations (des mots proches) comme «Souhaiter», «Désirer» etc. Plus précisément, si un système de traduction automatique connaît le sens correct des mots de la langue source, alors il pourrait déterminer d'une manière plus efficace les mots appropriés correspondants dans la langue cible. Par conséquent, l'annotation d'un texte grâce à la désambiguïsation lexicale peut aider un système de traduction automatique à présélectionner le sens le plus pertinent d'un mot ambigu tiré à partir de ses données d'entraînement.

Dans notre travail, nous nous intéressons à étudier l'apport de la désambiguïsation lexicale en traduction automatique, ce qui nécessite de travailler sur des données où le sens des mots est présent. Pour atteindre ces objectifs, nous avons besoin de corpus parallèles anglais-arabe annotés en sens. Ainsi, une question se pose : Est ce que la tâche de désambiguïsation lexicale améliore effectivement la qualité de traduction automatique, notamment pour une langue morphologiquement riche et complexe comme l'arabe ? Il

s'agit d'une évaluation ***In vivo*** de notre système de désambiguïsation lexicale arabe en fonction de sa contribution à la performance de la traduction automatique.

Parmi les rares travaux qui on été effectués en vue d'améliorer la tâche de traduction automatique à l'aide de la désambiguïsation lexicale nous citons les travaux de Carpuat and Wu [2007], qui ont montré que l'intégration des prédictions d'un système de désambiguïsation lexicale dans un système de traduction automatique statistique basé sur les segments, améliore la qualité de la traduction en l'évaluant sur les trois ensembles de tests chinois-anglais IWSLT06 différents. De même, ils ont obtenu des améliorations sur la grande tâche de traduction automatique chinoise et anglaise du NIST. Ils ont évalué la performance du système de traduction automatique en termes du score BLEU et d'autres mesures d'évaluation automatique.

En outre, Nguyen et al. [2018] ont présenté une étude de cas utilisant la désambiguïsation lexicale du coréen ainsi que des informations morphologiques afin d'améliorer la performance des systèmes de traduction neuronaux. Tout d'abord, ils ont construit un réseau lexico-sémantique coréen (LSN) comme base de connaissances lexico-sémantiques à grande échelle. Ensuite, en se basant sur le LSN coréen, ils ont crée un système de désambiguïsation lexicale coréen qui peut annoter le sens correct des mots coréens dans le corpus d'entraînement. Enfin, ils ont mené des expériences de traduction en utilisant des paires de langues coréenne-anglaise, coréenne-française, coréenne-espagnole et coréenne-japonaise. Les résultats expérimentaux montrent que l'utilisation du système de désambiguïsation lexicale coréen ainsi que des informations morphologiques ont amélioré la qualité de traduction des systèmes de traduction automatique neuronaux (pour toutes les combinaisons linguistiques) en termes de score BLEU [5] (de 2,94 points en moyenne), TER (de 4,04 points en moyenne) et DLRATIO (de 4,51 points).

2.5 Conclusion

Dans ce chapitre nous avons définit la tâche de désambiguïsation lexicale en général en présentant le processus de mise en oeuvre, les ressources utiles, ainsi que les approches pour la mise en oeuvre. Nous avons identifié les deux manières pour évaluer un système de désambiguïsation lexicale. En outre, nous avons présenté les méthodes de désambiguïsation lexicale appliquées à la langue arabe et son apport en traduction au-

5. La formule du score BLEU est définie dans la section 3.3.3

tomatique statistique. Par ailleurs, comme nous l'avons mentionné, de nombreux chercheurs ont constaté l'absence de corpus de référence dans la langue arabe et ils justifient ainsi la constitution de leur propre corpus, mais ils n'ont pas produit à ce jour de corpus de référence en accès libre construit sur une base de ressources communes.

Deuxième partie

Contributions

3

Cadre expérimental

3.1 Introduction

Dans ce chapitre, nous proposons dans un premier temps une méthode pour la création de corpus annotés en sens arabes pour la désambiguïsation lexicale, à l'aide de la traduction automatique et du portage des annotations. Nous citons quelques travaux de la bibliographie qui utilisent la technique de transfert d'annotations pour d'autres tâches dans le domaine du TAL.

Dans un second temps, nous présentons la tâche de traduction automatique dans les deux approches statistique et neuronale. Nous présentons également, la métrique BLEU la plus utilisée pour évaluer des systèmes de traduction automatique. Nous passons aussi en revue quelques travaux existants sur la traduction automatique de l'arabe, puis certains corpus parallèles anglais-arabe (dont deux obtenus grâce à une bourse d'étude de 6 000 dollars du LDC (Linguistic Data Consortium) pour utiliser les ressources qu'il distribue.

3.2 Méthode envisagée

Dans notre travail, nous nous intéressons à la désambiguïsation lexicale supervisée de la langue arabe. Étant donné que la création de ressources manuellement annotées est une tâche coûteuse et qui prend beaucoup du temps à réaliser, nous proposons une méthode afin de construire rapidement un système de désambiguïsation lexicale de l'arabe, pourvu que nous disposions d'un système de traduction automatique anglais-arabe pour traduire des grands corpus annotés en sens et porter leurs annotations.

Afin d'évaluer notre système de désambiguïsation lexicale, nous proposons deux approches illustrées par les deux figures 3.1 et 3.2 :

— Évaluation *in vitro* : en l'absence des corpus d'évaluation, nous envisageons d'enrichir une ressource lexicale existante nommée *OnotoNotes Release 5.0* (colorée en rouge dans la figure 3.1).

— Évaluation *in vivo* : nous souhaitons utiliser notre système de désambiguïsation lexicale pour l'amélioration de la traduction automatique. Plus précisément, nous proposons d'exploiter les hypothèses (prédictions) du système de désam-

biguïsation lexicale pour améliorer la qualité de systèmes de traduction arabe-anglais et anglais-arabe.

Toute réalisation dans cette direction va mettre en place une boucle vertueuse, améliorant la traduction automatique qui en retour pourra améliorer la désambiguïsation lexicale.

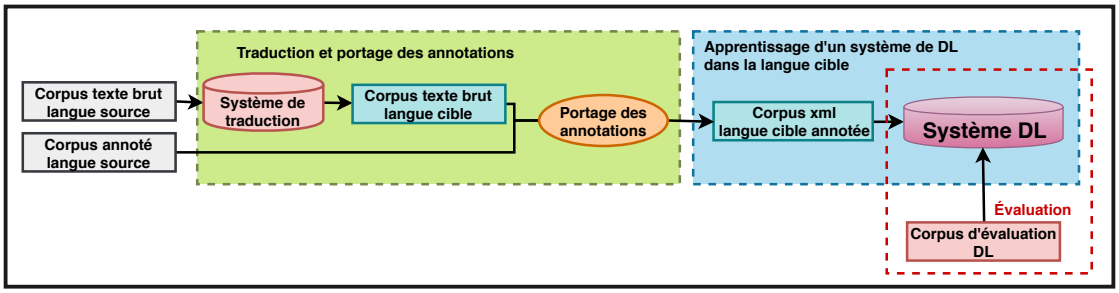

FIGURE 3.1: Évaluation *in vitro* de la désambiguïsation lexicale

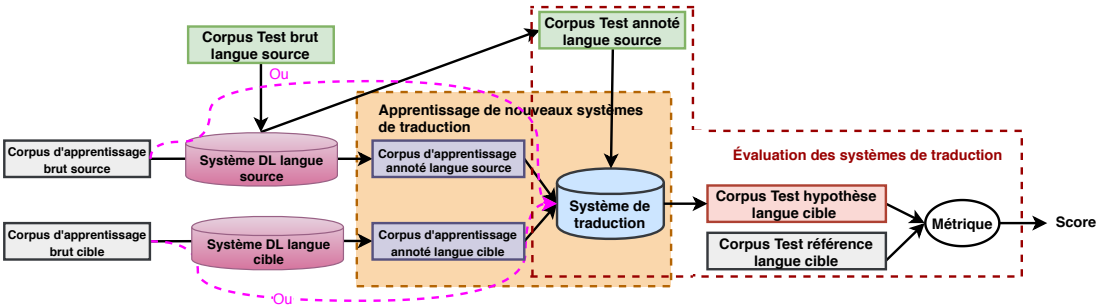

FIGURE 3.2: Évaluation *in vivo* de la désambiguïsation lexicale

La figure 3.1 décrit le processus de création d'un système de désambiguïsation lexicale en se basant sur la traduction automatique et sur une technique de portage des annotations. Disposant d'un corpus annoté en sens (issus du *Princeton WordNet*anglais) dans une langue source (ici l'anglais), nous proposons de traduire ce dernier vers une langue cible (ici l'arabe) tout en portant ses annotations. Ensuite, à l'aide de ce grand corpus arabe annoté en sens, nous allons créer un système de désambiguïsation lexicale supervisé que nous évaluons en utilisant un corpus de référence (à enrichir).

Par ailleurs, la figure 3.2 présente la seconde partie de notre travail. Nous proposons d'annoter le corpus parallèle d'entraînement de traduction automatique à l'aide de deux systèmes de désambiguïsation lexicale supervisés : le premier est le système créé pour la langue arabe, et le second est un système de désambiguïsation lexicale anglais construit au sein de l'équipe GETALP du LIG par Vial et al. [2018b].

3.2.1 Travaux existants sur le portage des annotations

Le transfert d'annotations linguistiques est utilisé depuis les années 1990 [Brown et al., 1991], où des corpus parallèles (source, cible) annotés en langue source sont alignés pour transférer des annotations vers la langue cible.

Yarowsky et al. [2001] ont utilisé un corpus parallèle afin d'adapter des outils monolingues à de nouvelles langues, tels que des analyseurs morphologiques, des analyseurs morpho-syntaxiques etc. Pour réaliser le transfert entre les langues, ils ont utilisé des alignements au niveau des mots entre les phrases d'un corpus parallèle.

Apidianaki [2008] a présenté une méthode d'induction de sens, qui combine des informations contextuelles et de traduction provenant d'un corpus d'entraînement parallèle bilingue afin d'identifier les sens des mots polysémiques de la langue source. Les sens induits peuvent être utilisés pour établir des correspondances sémantiques entre ces mots et leurs équivalents de traduction dans le corpus

Padó and Lapata [2009] ont utilisé le transfert des annotations afin obtenir automatiquement des annotations FrameNet pour des nouvelles langues, en utilisant les ressources disponibles pour l'anglais et en exploitant des corpus parallèles. Leur objectif était de transférer des rôles sémantiques de l'anglais vers des langues moins riches en ressources. Ils ont réalisé une évaluation expérimentale sur un corpus parallèle anglais-allemand qui prouve la faisabilité de leur approche.

Kim et al. [2010] ont présenté une méthode de transfert d'annotations multilingue afin d'extraire des relations dans une langue pauvre en ressources. Pour cela, ils ont proposé des méthodes de propagation d'annotations d'une langue riche en ressources vers une langue cible en utilisant des corpus parallèles. Ils ont introduit trois stratégies pour réduire le bruit de projection d'annotations (filtrage d'alignement basé sur une heuristique, correction d'alignement basée sur le dictionnaire et sélection d'instance basée sur l'évaluation). Ils ont appliqué leurs méthodes à la tâche de détection des relations en coréen et ont montré que les instances projetées à partir d'un corpus parallèle anglais-coréen améliorent la performance de la tâche lorsque les stratégies de réduction du bruit sont adoptées.

Tiedemann et al. [2014] ont utilisé la traduction automatique statistique et le portage des annotations pour créer des données d'entraînement synthétiques à partir des annotations sources originales. Ils ont appliqué cette technique à l'analyse des dépendances, en exploitant le *Universal Dependency Treebank v1* [McDonald et al., 2013] pour une évaluation adéquate.

Dans un autre travail, Zennaki et al. [2015] ont proposé une méthode pour construire automatiquement des outils d'analyse basés sur la projection multilingue d'annotations linguistiques en utilisant des corpus parallèles. Ils ont utilisé les réseaux neuronaux récurrents comme outil d'analyse multilingue.

van der Plas and Apidianaki [2014] ont proposé une méthode de transfert d'annotations sémantiques (étiquettes sur les prédicats), d'une langue à l'autre sur la base de corpus parallèles. Cette approche consiste à agréger des informations repérées dans l'ensemble du corpus parallèle. Ils ont annoté le corpus Europarl anglais à l'aide d'un système automatique entrainé sur PropBank, ensuite ils ont utilisé les alignements du corpus afin d'obtenir un corpus français automatiquement annoté en rôles PropBank.

La projection multilingue a également été adaptée avec succès pour transférer les annotations de sens des mots. Citons les travaux de Bentivogli et al. [2004], qui ont présenté une approche visant à créer des ressources annotées linguistiquement de haute qualité basées sur l'exploitation de corpus parallèles alignés. Ils ont créé un corpus parallèle anglais/italien (appelé MultiSemCor corpus).

Par ailleurs, Nasiruddin et al. [2015] ont construit un système de transfert d'annotation, afin de créer un système de désambiguïsation lexicale supervisé pour le français et le bengali.

Notre méthode diffère de celles décrites précédemment en plusieurs points : nous utilisons un système de traduction automatique (statistique et puis neuronal) associé à des traitements (prétraitement et post-traitement) favorisant la qualité de l'annotation et nous effectuons la tâche de transfert d'annotation après avoir traduit les corpus de la source vers la langue cible. Notre approche gère également les problèmes d'alignement des mots liés au transfert d'annotation. L'ensemble des programmes et ressources créés sont disponibles gratuitement et librement.

3.3 Traduction automatique

La traduction automatique (TA) consiste à traduire un texte d'un langage naturel sans intervention d'un humain. En effet, il existe différentes approches de traduction automatique telles que les approches statistiques (hiérarchiques, à base de mots, à base de segments, etc) ou encore les approches neuronales (à base de réseaux de neurones récurrents, à base de réseaux de neurones convolutifs, etc). Dans notre travail, nous nous intéressons aux méthodes de traduction automatique les plus utilisées que sont les

approches à base de segments et celles basées sur les réseaux de neurones récurrents.

3.3.1 Traduction automatique statistique

La traduction automatique statistique (TAS) est le processus qui consiste à traduire un texte rédigé dans une langue source vers un texte dans une langue cible, en se basant sur l'apprentissage de modèles statistiques à partir de corpus parallèles. En effet, comme il est montré dans la figure 3.3, la traduction automatique statistique se base essentiellement sur : un modèle de langage (ML), un modèle de traduction (MT) et un décodeur.

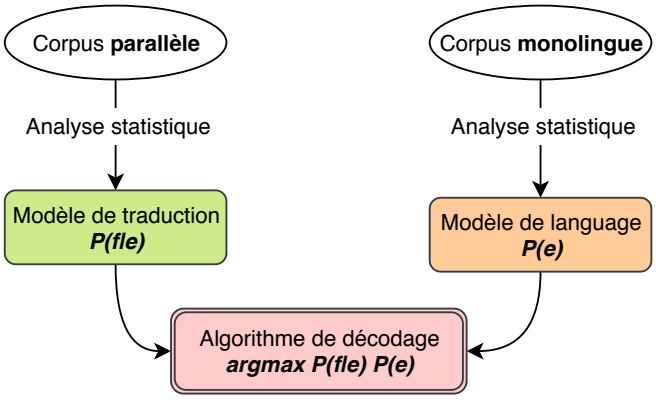

FIGURE 3.3: Processus de la traduction automatique statistique, où $P(e)$ la probabilité qu'une séquence de mots e soit vraisemblable et $P(f|e)$ la probabilité de la séquence cible e la plus probable sachant la phrase source f

3.3.1.1 Modèle de langue

Parmi les modèles de langue employés pour la construction des systèmes de traduction automatique statistique nous trouvons le modèle Cache [Kuhn and De Mori, 1990] qui se base sur les dépendances des mots non contigus, le modèle Trigger [Lau et al., 1993] qui consiste à déterminer le couple de mots (X,Y) où la présence de X dans l'historique déclenche l'apparition de Y, ou encore le modèle n-gramme ($1 \leq n \leq 5$) qui est le plus exploité dans les systèmes de traduction statistique notamment le modèle trigramme (3-gramme) pour le traitement des langues européennes. Le modèle n-gramme

permet d'estimer la vraisemblance d'une suite de mots en lui attribuant une probabilité. Soit $e = w_1 w_2 ... w_k$ une séquence de k mots dans une langue donnée et n la taille maximale des n-gramme ($1 \leq n \leq 5$) , $P(e)$ se définit comme :

$$P(e) = \prod_{i=1}^{k} (w_i | w_{i-1} w_{i-2} ... w_{i-n+1}) \tag{3.1}$$

IRSTLM

Plusieurs outils ont été proposés dans la littérature pour créer des modèles de langage de type n-grammes tels que le IRSTLM [Federico et al., 2008]. Il s'agit d'une boîte à outils utilisée pour la construction des modèles de langage statistiques. L'avantage de cette boîte à outils est de réduire les besoins de stockage ainsi que la mémoire lors de décodage. Par conséquent, cet outil nous permet de gagner du temps pour le chargement du modèle de langage.

3.3.1.2 Modèle de traduction à base de segments

Le modèle de traduction à base de segments (*phrase-based*) est souvent utilisé pour la tâche de traduction automatique statistique. L'objectif est d'estimer la probabilité $P(f|e)$ qui est définit comme suit :

$$P(f|e) = \prod_{i=1}^{I} \phi(\bar{f}_i | \bar{e}_i) d(a_i - b_{i-1}) \tag{3.2}$$

$\phi(\bar{f}_i|\bar{e}_i)$ présente la probabilité de traduction de la phrase hypothèse $\bar{e}_i$ avec la phrase source $\bar{f}_i$. $d(a_i - b_{i-1})$ indique la distribution de probabilité de distorsion avec a_i et b_{i-1} sont respectivement les positions de départ et de fin du segment dans la langue source traduit en segment hypothèse dans la langue cible.

Comme illustré dans la figure 3.4, afin de construire un modèle de traduction à base de segments [Koehn et al., 2003] , il est nécessaire de passer par trois étapes indispensables :

— Segmentation de la phrase en séquences de mots
— Traduction des séquences de mots en se fondant sur la table de traduction
— Ré-ordonnancement des séquences de mots à l'aide d'un modèle de distorsion

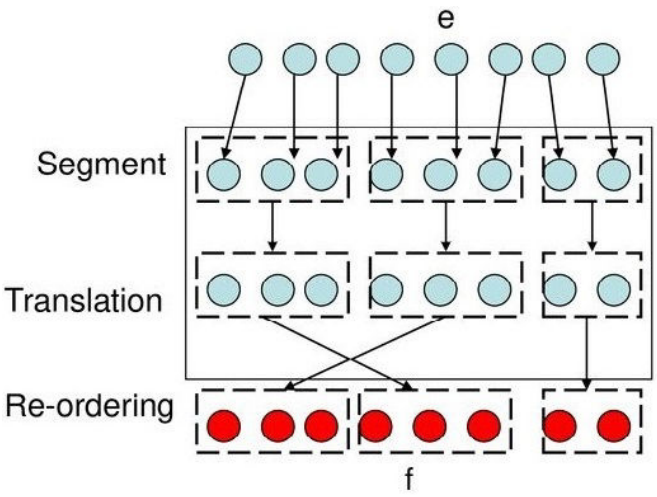

FIGURE 3.4: Exemple d'alignement à base de segments [Amr, 2008]

3.3.1.3 Décodeur

Moses [Koehn et al., 2007] est une boîte à outils disponible sous licence libre GPL, basée sur des approches statistiques de traduction automatique. En effet, Moses nous permet de développer et manipuler un système de traduction selon nos besoins grâce à ses nombreuses caractéristiques, telle que la production du modèle de traduction et le modèle de réordonnancement à partir des corpus volumineux.
Parmi les principaux modules du Moses, on trouve :

- **Train** : permet de construire des modèles de traduction ainsi que des modèles de réordonnancement.
- **Mert** : permet d'ajuster les poids des différents modèles afin d'optimiser et maximiser la qualité de traduction en utilisant les données de développement (Dev) .
- **Décodage** : ce module contient des scripts et des exécutables permettant de trouver la traduction la plus probable d'une phrase source en consultant les modèles du module Train.

3.3.2 Traduction automatique neuronale

Comme décrit précédemment, la traduction automatique statistique est fondée sur des modèles probabilistes (modèle de langage, modèle de traduction ainsi qu'un modèle de réordonnance) pour apprendre un système de traduction automatique . La per-

tinence de ces modèles a une forte influence sur la qualité du système de traduction automatique produit.

Récemment, Sutskever et al. [2014] ont proposé une méthode d'apprentissage profond «End-to-End» générique pour toute tâche «sequence to sequence» (en utilisant des réseaux de neurones récurrents) et qui peut être exploitée pour apprendre des systèmes de traduction automatique. Il s'agit d'un premier réseau neuronal récurrent (encodeur) qui prend en entrée une séquence de mots (dans la langue source) et l'encode en un vecteur de représentations de taille fixe qui sert à modéliser la langue cible à l'aide d'un second réseau de neurone récurrent (décodeur), tel qu'il est présenté dans la figure 3.5.

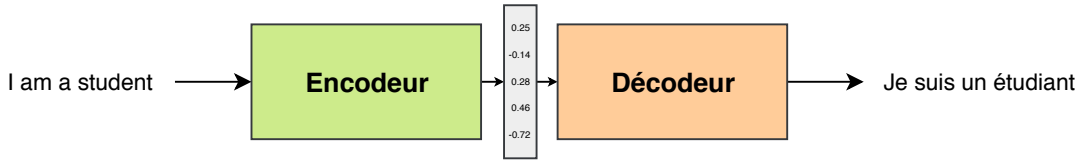

FIGURE 3.5: Processus d'apprentissage

Cependant, Bahdanau et al. [2015] ont amélioré cette méthode en ajoutant le mécanisme d'attention afin de résoudre un problème de l'approche encodeur-décodeur lié aux longues séquences, car l'encodeur doit compresser toutes les informations nécessaires d'une phrase source en un vecteur de longueur fixe.

Par conséquent, un réseau neuronal profond lit une séquence de mots dans la langue source et produit une séquence de mots dans la langue cible, tout en optimisant les paramètres au moment de l'apprentissage du système sur un corpus d'entraînement parallèle (sans données monolingues). La sortie du réseau est obtenue en appliquant des fonctions Softmax permettant de prédire les mots les plus probables. La fonction Softmax prend en entrée un vecteur $x = (x_1 \ldots x_n)$ de N nombre réels et fournit en sortie un vecteur $\sigma(x)$ de N nombres réels positifs ayant une somme égale à 1. Elle est définit comme suit :

$$\sigma(x)_i = \frac{e^{x_i}}{\sum_{n=1}^{N} e^{x_n}}, \forall i \in 1,\ldots,N \tag{3.3}$$

Dans ce qui suit, nous présentons brièvement la notion de réseaux de neurones, et, nous décrivons plus en détail les deux approches de réseaux de neurones récurrents et de mécanisme d'attention pour la tâche de traduction automatique.

3.3.2.1 Introduction aux réseaux de neurones

Les réseaux de neurones artificiels sont inspirés du neurone biologique [McCulloch and Pitts, 1943]. En 1957, Rosenblatt [1957] a inventé le modèle algorithmique d'un neurone artificiel qui a servit de base aux premiers réseaux de neurones. Notant que un perceptron est un très simple réseau de neurone artificiel, constitué d'un seul neurone.

Depuis 2012, les réseaux de neurones (perceptrons à plusieurs couches cachées) bénéficient de la révolution technique avec le développement de l'informatique et le rassemblement d'énormes quantités de données. De nos jours, ils constituent l'une des techniques les plus performantes et les plus utilisées pour la prédiction, la classification ainsi que l'analyse des données dans plusieurs tâches. Les réseaux de neurones sont des estimateurs universels de fonction. En outre, la vague actuelle de regain d'intérêt pour les réseaux de neurones récurrents est venue par le domaine de l'analyse d'images. Étant donné que les réseaux de neurones sont des estimateurs universels, ils peuvent être utilisés pour n'importe qu'elle tâche, d'où le nombre croissant de leurs domaines d'application.

Comme décrit dans la figure 3.6, un neurone artificiel est composé généralement de plusieurs entrées $x_1 \ldots x_n$. À chaque entrée est associée un poids synaptique w_i où $1 \leq i \leq n$. La somme des vecteurs d'entrées $X = x_1 \ldots x_n$ pondérés par le vecteur de poids $W = w_1 \ldots w_n$ et un paramètre $b \in \mathbb{R}$ représentant le biais, est calculée par le neurone et sera passée par la fonction d'activation φ afin de produire une seule sortie y. Celle-ci peut être exploitée comme entrée pour d'autres neurones et se définit par l'équation 3.4 :

$$y = \varphi(W.X + b) \tag{3.4}$$

La fonction d'activation φ peut se définir de différentes manières, où $Z = W.X + b$:

— identité (ou linéaire) :

$$\varphi(Z) = Z \rightarrow \ \in (-\infty, +\infty) \tag{3.5}$$

— Sigmoïde :

$$\varphi(Z) = \frac{1}{1 + e^{-Z}} \rightarrow \ \in (0, 1) \tag{3.6}$$

— Tangente hyperbolique :

$$\varphi(Z) = \tanh(Z) = \frac{e^{Z} - e^{-Z}}{e^{Z} + e^{-Z}} \rightarrow \ \in (-1, 1) \tag{3.7}$$

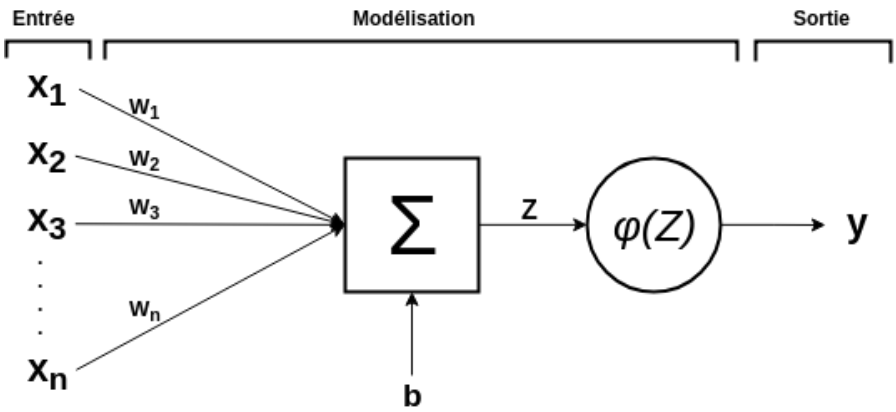

FIGURE 3.6: Structure d'un neurone artificiel

— ReLu (Rectified linear unit) :

$$\varphi(Z) = Z^{+} = max(0,Z) \rightarrow \ \in [0,+\infty[\tag{3.8}$$

3.3.2.2 Les réseaux de neurones récurrents

Contrairement aux réseaux de neurones classiques qui ne peuvent pas conserver des informations contextuelles sur leur entrée, les réseaux de neurones récurrents ou *Recurrent Neural Network* en anglais (RNN) [Medsker and Jain, 1999] peuvent traiter une entrée de taille variable et s'attaquent au problème de mémoire avec des boucles permettant de faire passer l'information d'une étape précédente du réseau à l'autre.

3.3.2.2.1 Réseau Elman

Un simple réseau récurrent ou encore appelé réseau Elman [Elman, 1990] est un réseau qui comporte trois couches et qui prend en entrée une séquence de vecteurs $x_1 \ ... \ x_T \in \mathbb{R}^{T*m}$. Le réseau est composé d'un état $s_t \in \mathbb{R}^n$ mis à jour à chaque nouvelle entrée x_t tout en encodant les informations des entrées passées.

La figure décrit 3.7 un exemple de boucle d'un réseau de neurones récurrent, où X_t et h_t représentent respectivement l'entrée et la sortie de la couche cachée A.

L'état d'un RNN est mis à jour comme suit :

$$s_t = \tanh(W_{rec}s_{t-1} + W_{in}x_t + b) \tag{3.9}$$

où $W_{rec} \in \mathbb{R}^{n*n}$, $W_{in} \in \mathbb{R}^{n*m}$ et $b \in \mathbb{R}^n$ représentent des paramètres entraînables du RNN.

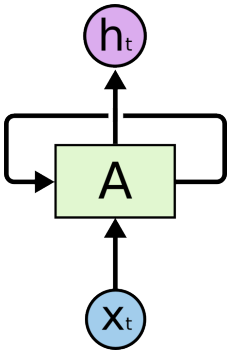

FIGURE 3.7: Exemple de boucle d'un réseau de neurones récurrent

Par ailleurs, tout comme les réseaux de neurones classiques, les RNN utilisent la technique de rétropropagation du gradient pour l'entraînement du réseau. Il s'agit de calculer le gradient de l'erreur pour chacun des neurones. Toutefois, il est difficile d'entraîner un RNN avec de longues séquences, ou lorsque le décalage entre les évènements et le retour d'erreur (gradient) correspondant est important [Hochreiter and Schmidhuber, 1997]. De même, les RNN ont une difficulté à stocker des informations à long terme. Cela est dû à la variation exponentielle du gradient, car à chaque étape, l'erreur est multiplié par W_{rec}, ce qui peut faire disparaître ou exploser le gradient. Pour faire face à ce problème, plusieurs architectures ont été proposées parmi lesquelles nous trouvons celle la plus utilisée nommée LSTM.

3.3.2.2.2 LSTM

La mémoire à long et court terme (LSTM) ou *Long Short Term Memory networks* en anglais (dites cellules «à mémoire» ou cellules «récurrentes»), est un type de réseaux de neurones récurrents [Hochreiter and Schmidhuber, 1997, Graves et al., 2013], capable d'apprendre les dépendances à long terme. Les LSTM sont généralement utilisés dans la traduction neuronale et de manière générale pour toute tâche ayant à gérer des dépendances non locales entre les unités constituant les données d'entrée.

Comme illustré dans la figure 3.8, l'architecture d'une cellule LSTM simple est caractérisée par : une observation X_t à l'instant t et l'observation de l'état précédent h_{t-1}, une porte d'oubli (*Forget Gate*) permettant de sélectionner les informations pertinentes de l'état caché h_{t-1} en oubliant les informations inutiles, une porte de modulation *Modulation Gate* pour ajuster l'état de la cellule avec une fonction d'activation *tanh*, une porte d'entrée qui détermine les informations qui doivent être entrées dans la cellule

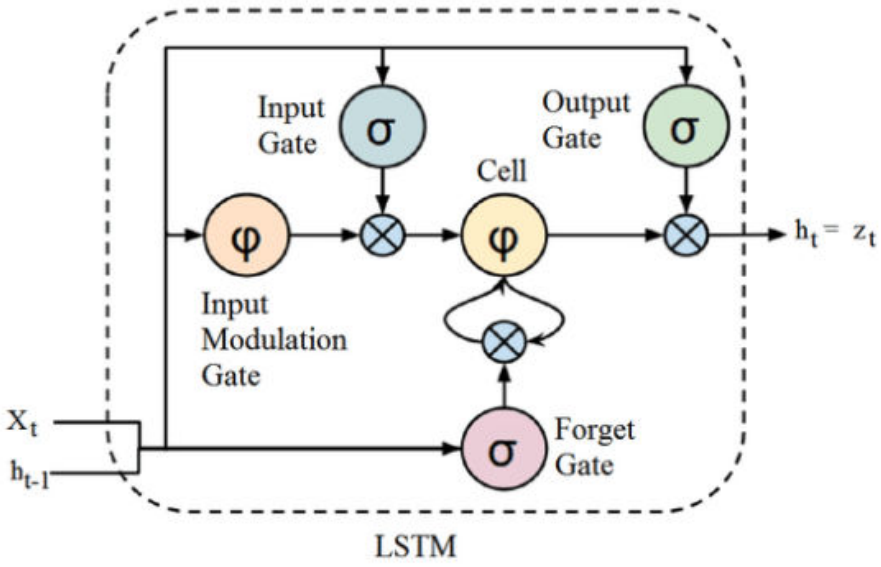

FIGURE 3.8: Une cellule LSTM [Kang, 2017]

(*Cell*) et une porte de sortie (*Output Gate*) qui détermine les informations pertinentes qui doivent passer au prochain état caché.

3.3.2.2.3 Encodeur bidirectionnel

Bahdanau et al. [2015] utilisent un encodeur RNN bidirectionnel. Il s'agit de deux RNN indépendants superposés qui lisent la séquence d'entrée dans les deux sens (de gauche à droite et de droite à gauche) au lieu d'un seul RNN. Les états cachés de l'encodeur sont généralement une concaténation des sorties des deux RNNs.

Comme décrit dans la a figure 3.9, le RNN bidirectionnel est composé de deux couches cachées qui présentent respectivement un RNN avant A (*forward*) et le RNN arrière A' (*backword*) permettant d'encoder l'entrée $X = x_0, x_1, \ldots, x_i$. Les couches A et A' contiennent des informations pertinentes sur l'historique des mots précédents et suivants afin de produire correctement en sortie une séquence correspondante $Y = y_0, y_1, \ldots, y_i$.

3.3.2.3 Mécanisme d'attention

Comme mentionné précédemment, le mécanisme d'attention [Bahdanau et al., 2015] est une extension de l'architecture RNN de type encodeur-décodeur de base. En effet, il permet d'ignorer les informations non pertinentes de l'entrée afin de ne pas endommager les prédictions ultérieures. À chaque étape, il génère un vecteur de

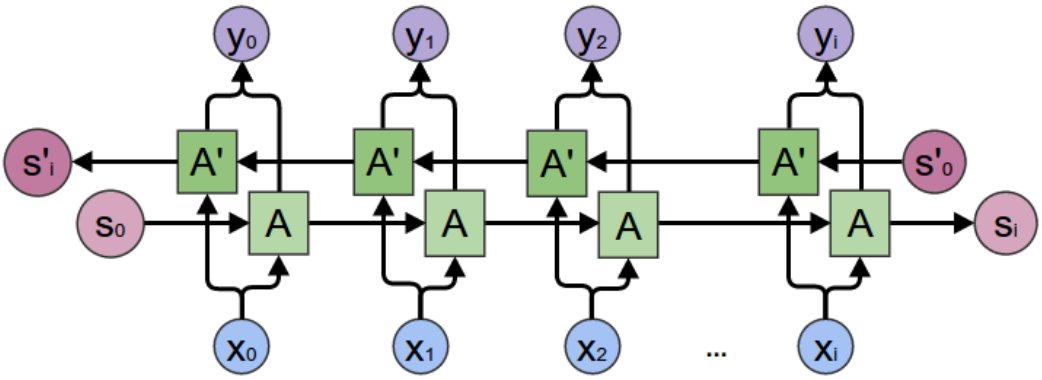

FIGURE 3.9: Structure générale des réseaux neuronaux récurrents bidirectionnels [Olah, 2015]

contexte et permet au décodeur d'utiliser une représentation différente de la séquence d'entrée. Il est généralement utilisé dans les modèles «sequence-to-sequence». Ainsi, en traduction automatique, le mécanisme d'attention cherche les informations les plus pertinentes d'une phrase source afin de prédire un mot cible, permettant de traduire de longues phrases contrairement à l'approche encodeur-décodeur de base.

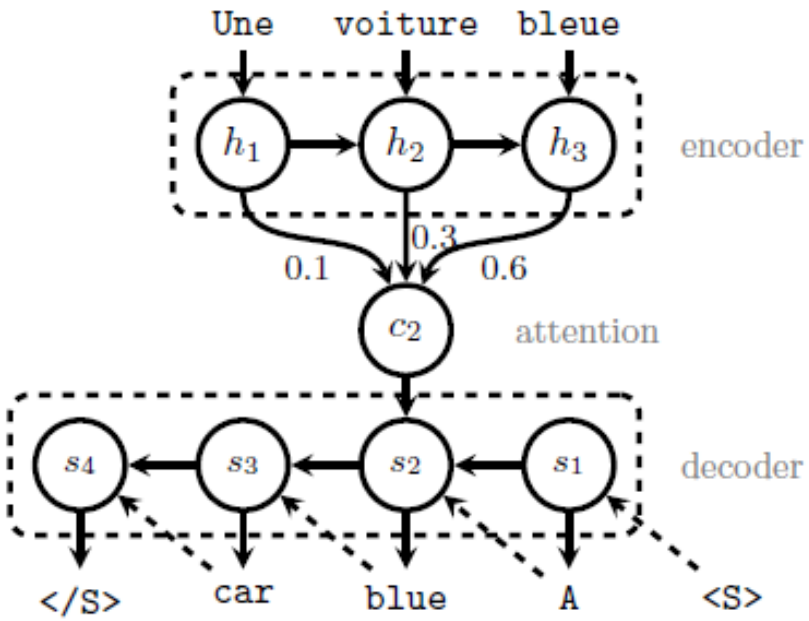

FIGURE 3.10: Illustration d'un modèle «sequence-to-sequence» avec Mécanisme d'attention [Bérard, 2018]

3.3.3 BLEU : Métrique d'évaluation automatique

Le score BLEU (en anglais : Bilingual Evaluation Understudy) a initialement été proposé par Papineni et al. [2002]. C'est un algorithme utilisé en vue d'évaluer la qualité des hypothèses de sortie produites par un système de traduction automatique.

En effet, le concept est fondé sur l'idée de comparer l'hypothèse de traduction avec une ou plusieurs références au niveau des séquences de mots n-grammes de longueurs variables ($1 \leq n \leq N$, $N = 4$ par défaut) en comptant le nombre de correspondances.

Le score BLEU est définit alors comme suit :

$$BLEU = BP \cdot \exp(\sum_{n=1}^{N} w_n \; log \; p_n) \quad \text{avec} \quad BP = \begin{cases} 1 & \text{si } c > r \\ e^{1-r/c} & \text{si } c \leq r \end{cases} \tag{3.10}$$

Où BP la pénalité de brièveté, c le nombre de mots de l'hypothèse, r le nombre de mots de la référence, p_n la précision d'une séquence de mots n-grammes, $N = 4$ la longueur maximale d'une séquence de mots et $w_n = \frac{1}{N}$ un ensemble de poids positifs uniformes ayant une somme égale à 1.

Le score BLEU est normalisé entre 0 et 1, et il est exprimé généralement en pourcentage. Notons qu'une traduction humaine peut parfois obtenir un mauvais score BLEU, si elle s'écarte de la référence.

3.3.4 Travaux connexes sur la traduction automatique arabe

Dans le cadre de la traduction automatique statistique arabe, Ghaffar and Fakhr [2011] ont proposé un système de traduction automatique statistique anglais-arabe, construit en utilisant le décodeur Moses et l'outil Giza++ pour l'alignement, ainsi que le corpus parallèle LDC2004T18 (ressource payante) comme données d'entraînement. L'objectif était d'améliorer les performances de leur système, en effectuant un prétraitement pour les deux langues (source et cible), en utilisant l'outil MADA pour la segmentation morphologique de l'arabe. Le prétraitement appliqué implique le regroupement des nombres, des dates, ainsi que des noms propres des personnes. Toutefois, l'outil utilisé a mal segmenté les noms des personnes, ce qui a introduit plus d'ambiguïté et a affecté négativement la qualité de l'alignement et du modèle de langage. Ils ont obtenu 19.1% et 27.39% en termes de score BLEU respectivement sur un système de base (normalisation de base) et un système optimisé (avec toutes les chaînes de normalisation proposées).

Dans leurs travaux, Hadla et al. [2014] ont comparé l'efficacité de deux systèmes de traduction automatique disponibles en ligne (Google Translate et Babylon) pour traduire de l'arabe vers l'anglais. Pour cela, ils ont construit un corpus de test, contenant plus que 1000 phrases arabes avec deux traductions anglaises de référence pour chacune des phrases dans la langue source. Ce corpus se compose de 4169 mots arabes, où le nombre de mots arabes uniques est de 2539. Ces phrases arabes sont réparties en quatre types de base (déclarative, interrogative, exclamative et impérative). Les résultats expérimentaux montrent que le système de traduction automatique Google est meilleur que le système de traduction automatique Babylon, ayant respectivement des performances moyennes de 44.96% et 39.84% (pour les quatre types de phrases) en termes de score BLEU.

Par ailleurs, la traduction automatique neuronale est de plus en plus utilisée. Depuis 2016, Google (Google translate) a commencé à utiliser la traduction neuronale pour huit langues, ainsi que Microsoft (Skype Translator). Almahairi et al. [2016] se sont concentrés à la traduction automatique neuronale arabe dans les deux sens (EN↔AR) et ont comparé leur système à un système de traduction basé sur des segments, en utilisant différentes configurations pour le prétraitement de l'arabe. Ils ont constaté que les deux systèmes sont presque similaires en termes de performance, mais le système de traduction automatique neuronal est généralement meilleur sur un ensemble de test hors domaine. Dans le cas de la traduction EN→AR, et en utilisant la normalisation et la tokenisation, leur système de traduction automatique neuronale atteint respectivement 33,62% et 24,46% en termes de score BLEU, sur l'ensemble de test dans le domaine (MT05) et l'ensemble de test hors domaine (MEDAR).

D'autre part, diverses approches ont été proposées pour faire face aux problèmes d'ambiguïté morphologique en arabe lors de la tokenisation. Dans l'un des premiers ouvrages, et d'ailleurs l'un des plus connus dans ce domaine, Habash and Sadat [2006] ont présenté différents schémas de tokenisation pour le pré-traitement de l'arabe en vue de voir quelle est la méthode de segmentation la plus utile pour la traduction automatique statistique et ils ont trouvé que *ATB* (décrit dans la section 1.3.2.1) est la meilleure technqiue de tokenisation des textes arabes pour la traduction automatique statistique. Ces schémas de tokenisation sont disponibles dans l'outil MADAMIRA (présenté dans la section 1.3.2.1). En outre, Bouamor et al. [2014] ont présenté des résultats d'évaluation humaine avec une qualité d'annotation élevée ainsi qu'une métrique adaptée pour l'évaluation de la traduction automatique arabe. Pour ce faire, ils ont utilisé les données annotées pour adapter le score BLEU pour l'arabe.

3.3.5 Corpus parallèles anglais-arabe

3.3.5.1 Ummah

Le corpus Ummah [1] est un corpus d'articles de journaux arabe aligné avec des traductions en anglais collectées via le service de presse *Ummah* de Janvier 2001 à Septembre 2004. Il a été produit par *LDC* (Linguistic Data Consortium) sous le numéro de catalogue LDC2004T18.

Il totalise 8 439 paires histoire, 68 685 paires de phrases, de mots arabes et 2M mots 2,5M anglais. Le corpus est aligné au niveau des phrases. Tous les fichiers de données sont des documents SGML.

Langue	Nombre de mots	Nombre de lignes
Arabe	1 626K	57K
Anglais	1 991K	57K

TABLE 3.1: Description des corpus Ummah

3.3.5.2 News

Le corpus News [2] (Arabic News Translation Text Part 1) a été produit par le *LDC* (Linguistic Data Consortium) sous le numéro de catalogue LDC2004T17. Trois sources de texte journalistique arabe ont été sélectionnées pour produire ce corpus arabe :

— Service de brèves journalistiques *AFP* : 250 brèves journalistiques, 44 193 mots arabes, octobre 1998 - décembre 1998.

— Service de brèves journalistiques *Xinhua* : 670 brèves journalistiques, 99 514 mots arabes, Novembre 2001 - Mars 2002

— An Nahar : 606 brèves journalistiques, 297 533 mots arabes, de Octobre 2001 - Décembre 2002

Langue	Nombre de mots	Nombre de lignes
Arabe	389K	16K
Anglais	519K	16K

TABLE 3.2: Description des corpus LDC-News

1. https://catalog.ldc.upenn.edu/LDC2004T187
2. https://catalog.ldc.upenn.edu/LDC2004T17

3.3.5.3 News Commentary

Le corpus News-Commentary [3] est un corpus parallèle aligné au niveau des phrases. Ce corpus contient des extraits de diverses publications de presse et de commentaires du projet *Syndicate* et il est disponible dans plusieurs langues (arabe, anglais, français, espagnol, allemand, et tchèque, etc).

Langue	Nombre de mots	Nombre de lignes
Arabe	2 499K	54K
Anglais	2 499K	54K

TABLE 3.3: Description du corpus News-Commentary

3.3.5.4 MultiUN

Le corpus MultiUN [4] est un ensemble de documents traduits des *United Nations* élaborés à la base par Eisele and Chen [2010]. Il est disponible dans 7 langues : anglais, français, allemand, arabe, espagnol, russe et chinois.

Langue	Nombre de mots	Nombre de lignes
Arabe	223 893K	9 131K
Anglais	253 254K	9 131K

TABLE 3.4: Description du corpus Multi-UN

3.3.5.5 TED

Le corpus TED [5] est un ensemble de transcriptions des conférences en anglais présentés sous format vidéo sur le site officiel de TED. Ces transcriptions ont été traduites par les bénévoles pour plus de 70 autres langues (arabe, français, italien, coréen, portugais, etc.).

Langue	Nombre de mots	Nombre de lignes
Arabe	2 302K	150K
Anglais	2 925K	150K

3. http://opus.nlpl.eu/News-Commentary11.php
4. http://opus.nlpl.eu/MultiUN.php
5. http://opus.nlpl.eu/TED2013.php

TABLE 3.5: Description du corpus TED

3.4 Conclusion

Dans ce chapitre, nous avons décrit le scénario envisagé pour la création des corpus annotés en sens arabes nécessaires pour la tâche de désambiguïsation lexicale. Nous avons également présenté certains travaux qui ont exploité la technique de portage des annotations pour d'autres applications.

Par ailleurs, nous avons présenté les deux approches de traduction automatique statistique (à base de segments) et neuronale (plus particulièrement en utilisant les réseaux de neurones récurrents), ainsi que la métrique la plus connue pour l'évaluation de la traduction automatique. De plus, nous avons cité quelques travaux sur la traduction automatique de l'anglais vers l'arabe et vice-versa. Nous avons finalement présenté certains corpus parallèles anglais-arabe que nous allons utiliser dans notre travail.

Dans le chapitre suivant, nous présentons le corpus de référence nécessaire pour l'évaluation de la désambiguïsation lexicale de l'arabe. De même, nous présentons d'une manière détaillée notre méthode de construction de corpus annotés en sens à l'aide d'une traduction automatique statistique et neuronale, et le transfert direct des annotations d'une langue source riche en corpus annotés comme l'anglais vers une langue cible moins bien dotée (ici l'arabe).

4

Production de ressources

4.1 Introduction

Dans les travaux existants sur la désambiguïsation lexicale de l'arabe, nous remarquons que la plupart des articles insistent sur le fait que le manque de diacritiques dans les textes arabes rend la tâche de désambiguïsation lexicale plus difficile pour l'arabe que pour d'autres langues comme le français ou l'anglais. Selon nous, le problème le plus important est le manque d'un corpus standard pour l'évaluation de l'arabe. Nous pouvons le constater dans la littérature où chaque système est évalué sur un corpus différent, réalisé en interne et non rendu disponible à la communauté. Dans leurs travaux, les auteurs comparent ensuite des résultats obtenus sur d'autres corpus. De notre point de vue, la validité scientifique d'une telle démarche est discutable. C'est pour cette raison que nous avons cherché un corpus d'évaluation facilement disponible et nous avons trouvé l'OntoNotes 5.0 annoté en sens issus de *Princeton WordNet* pour ses parties anglaises et chinoises seulement. Cependant, il manquait le lien entre les annotations OntoNotes et le *Princeton WordNet* pour sa partie arabe.

Ainsi, ce chapitre est organisé comme suit : nous présentons tout d'abord le corpus OntoNotes 5.0 qui comporte trois langues : anglais, chinois et arabe (qui n'était pas complet). Nous présentons alors les différentes manières d'aligner des ressources interlingues ainsi que la méthode et les différentes étapes suivies afin d'enrichir et mettre à jour la partie arabe d'OntoNotes 5.0. Nous décrivons ensuite nos systèmes de traduction automatique anglais-arabe crées ainsi que le processus de traduction des corpus et portage des annotations. Nous détaillons enfin les corpus arabes que nous avons produits pour la tâche de désambiguïsation lexicale de l'arabe.

4.2 Corpus d'évaluation commun pour l'arabe : OntoNotes Release 5.0

4.2.1 OntoNotes Release 5.0

Le projet OntoNotes [Weischedel et al., 2015] est le résultat d'un travail collaboratif entre BBN Technologies, l'Université du Colorado, l'Université de Pennsylvanie et l'Institut des sciences de l'information de l'Université de Californie du Sud. *OntoNotes Release 5.0* est la dernière version du corpus proposée par ce projet. C'est un grand corpus annoté libre de droit, construit à 90% d'accord inter-annotateur avec des infor-

mations structurelles (syntaxe et structures prédicat-arguments) et sémantiques superficielles (sens du mot lié à une ontologie et co-référence). Le corpus contient plusieurs genres de textes (Brèves journalistiques, conversations téléphoniques, weblogs, usenet newsgroups, broadcast, talk shows) en anglais et chinois et uniquement des données journalistiques pour la partie arabe, comme le montre le tableau 4.1. La partie arabe d'*OntoNotes Release 5.0* comprend 300K mots du corpus arabe *An-Nahar Newswire*. C'est sur cette partie que nous évaluons notre système tandis que nous l'entraînons en partie sur les traductions de la partie anglaise. Il n'y a pas de biais car les parties arabes et anglaises ne sont pas des traductions l'une de l'autre.

Les données du projet OntoNotes peuvent être utilisées dans diverses tâches dans le domaine de traitement automatique des langues telles que l'extraction d'informations, la recherche d'informations, la désambiguïsation lexicale, etc.

Type	**Arabe**	**Anglais**	**Chinois**
News	300k	625k	250k
BN	n/a	200k	250k
BC	n/a	200k	150k
Web	n/a	300k	150k
Tele	n/a	120k	100k
Pivot	n/a	n/a	300k

TABLE 4.1: Description d'OntoNotes Release 5.0 pour chaque langue disponible

Dans ce qui suit nous allons présenter les méthodes possibles pour la construction de ressources langagières multilingues notamment les ressources lexicales et sémantiques.

4.2.2 Alignement de ressources interlingues

Il existe trois types de méthodes de construction de ressources lexicales et/ou sémantiques électroniques : les méthodes de construction manuelle, les méthodes de construction automatique et les méthodes de construction semi-automatique. Nous nous intéressons particulièrement aux ressources lexicales multilingues et leurs interopérabilité. Pour chacune des catégories de ressources nous passons en revue un exemple d'une ressource existante.

4.2.2.1 Ressources construites manuellement

Ce type de construction de ressources est généralement réalisé par des experts du domaine. Cependant cette méthode est très coûteuse et prend beaucoup de temps. Ainsi, les ressources construites avec cette approche restent limitées en taille ainsi qu'en nombre de langues couvertes ; de plus elles ne sont pas tous disponibles et libres de droit. Parmi ces ressources, nous trouvons le *Princeton WordNet* ainsi que l'*WordNet arabe* (décrits dans la section 2.2.2.1.1).

4.2.2.2 Ressources construites automatiquement

Il s'agit de ressources créées automatiquement à partir d'autres ressources électroniques existantes, telles que *BabelNet* [Navigli and Ponzetto, 2012] qui est une ressource lexicale à grande échelle construite par alignement automatique des *synsets*, issus de *Princeton WordNet* et de pages Wikipedia correspondant aux entrées. Dans sa dernière version publiée en août 2016, la 3.7 [1], comprend 271 langues (parmi lesquelles l'arabe), 13 801 844 *Babel synsets*, 745 859 932 sens, 380 239 084 relations lexico-sémantiques et 40 709 194 définitions textuelles. Dans BabelNet arabe, il y a 3 082 624 entrées, 2 767 303 *Babel synsets*, 3 501 238 sens de mots et un degré de polysémie de 1,27.

4.2.2.3 Ressources construites semi-automatiquement

La construction de ce type de ressources s'effectue d'une manière semi-automatique. Pour appliquer cette méthode, on passe par une première étape automatique pour extraire des informations à partir d'une ressource existante donnée, cette étape est suivie d'une intervention humaine afin de vérifier/modifier et valider les informations produites automatiquement. Par exemple, Shi and Mihalcea [2005] ont présenté une approche semi-automatique pour intégrer les trois ressources lexicales différentes : *FrameNet* [Johnson et al., 2002], *VerbNet* [Kipper et al., 2000] et *WordNet*, dans une base de connaissances unifiée et plus riche, permettant une analyse sémantique plus robuste. Ils ont utilisé *VerbNet* comme un lien entre *FrameNet* et *WordNet* pour aligner les verbes.

1. L'ensemble de ces statistiques vient de la page `http://babelnet.org/stats` consultée le 23/09/2017

4.2.3 Enrichissement de la partie arabe de l'OntoNotes Release 5.0

Afin d'évaluer un système de désambiguïsation lexicale de l'arabe, il est nécessaire d'avoir un corpus arabe annoté en sens. C'est le cas de l'OntoNotes pour ses parties anglaises et chinoises puisqu'elles sont annotées avec des sens issus du *Princeton WordNet*. Malheureusement, le projet n'avait pas été mené jusqu'au bout sur la partie arabe et le lien entre les annotations OntoNotes et le *Princeton WordNet* sont absentes.

Nous proposons ainsi une mise à jour de la partie arabe de OntoNotes Release 5.0 d'une manière semi-automatique pour obtenir des mots annotés en sens avec le *Princeton WordNet* 3.0.

En pratique, pour la partie arabe de *Ontonotes Release 5.0*, les informations d'annotation en sens sont fournies dans des fichiers ayant l'extension *.sense*. Ces fichiers sont organisés comme suit :

<**chemin**> <**id_phrase**> <**id_mot**> <**lemme**> <**groupe**> <**id_sens**>

— <**id_phrase**> **:** l'identifiant de la phrase arabe dans le corpus.
— <**id_mot**> **:** l'identifiant du mot arabe dans la phrase contexte.
— <**lemme**> **:** le lemme du mot.
— <**id_sens**> **:** l'identifiant du sens du mot.

Chaque lemme est associé à un fichier *xml* qui contient la liste des sens liés à ce lemme et identifié par un *id*. La Figure 4.1 présente le premier sens du lemme en arabe transcrit en caractères latins *EuDow* (Membre en français). Ces fichiers contiennent essentiellement les informations suivantes :

— La balise <**sens**> contient les informations *Groupe*, *identifiant du lemme*, le sens du lemme en anglais ainsi que le type.
— La balise <**example**> donne un exemple en anglais et/ou une traduction en arabe contenant le mot ambigu.
— Une balise <**wn**> qui est vide dans la version 5 de l'OntoNotes et qui désigne les clés des sens anglais possibles du mot dans *Wordnet*.

Comme il est indiqué dans la figure 4.2, nous proposons un traitement semi-automatique afin de compléter les balises <*wn*> pour réaliser la correspondance avec le *Princeton WordNet* 3.0. En effet, pour chaque mot annoté, nous mettons en œuvre les étapes suivantes :

1. Récupérer le lemme avec le sens exact du mot

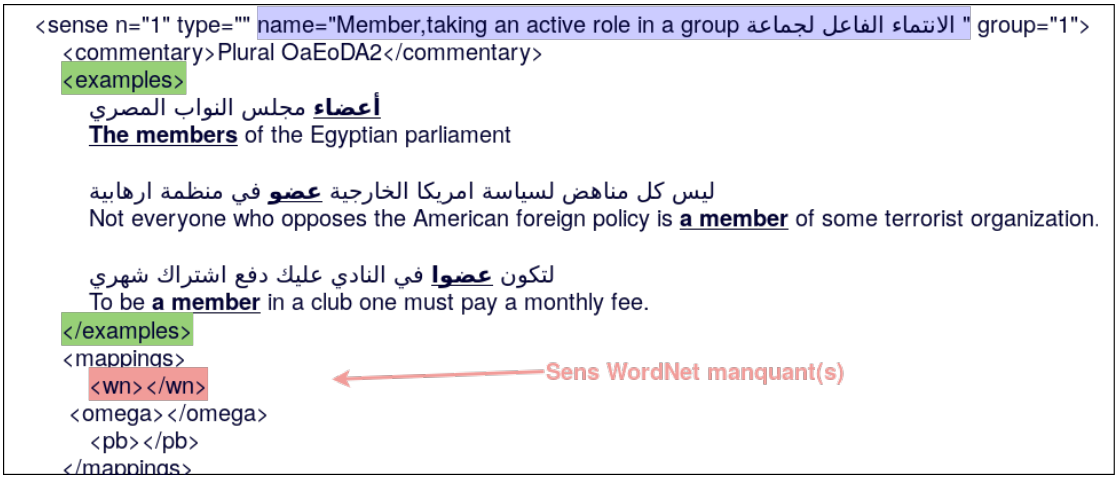

FIGURE 4.1: Exemple du premier sens dans le document original EuDow-n.xml (Membre)

2. Extraire l'information <*name*> ainsi que le contenu de la balise <*example*>, depuis le fichier *lemme.xml*
3. Trouver une ou plusieurs traductions en anglais selon le contexte du mot
4. Pour chaque lemme *l* d'un exemple de traduction en anglais trouvé, nous suivons deux étapes :
   — Si le lemme *l* existe dans la partie anglaise d'OntoNotes, notre outil d'annotation nous renvoie les synonymes possibles, tout en sélectionnant manuellement les synonymes corrects selon le contexte de l'exemple dans la phrase arabe
   — Sinon, notre outil d'annotation nous renvoie les synonymes possibles, leurs descriptions ainsi que leurs exemples d'utilisation à partir du *Princeton WordNet* anglais afin de sélectionner manuellement les sens corrects selon le contexte Avec ce traitement, 60% des sens ont été récupérés depuis la partie anglaise d'OntoNotes et 40% ont été récupérés manuellement depuis *Princeton WordNet*.
5. Mettre les clés (Keys) *Princeton WordNet* dans la balise <*wn*>

La figure 4.3 présente l'exemple d'ajout des sens WordNet manquants dans la balise <wn> du sens 1 du lemme *EuDow*, après avoir mis à jour la partie arabe de *OntoNotes Release 5.0*.

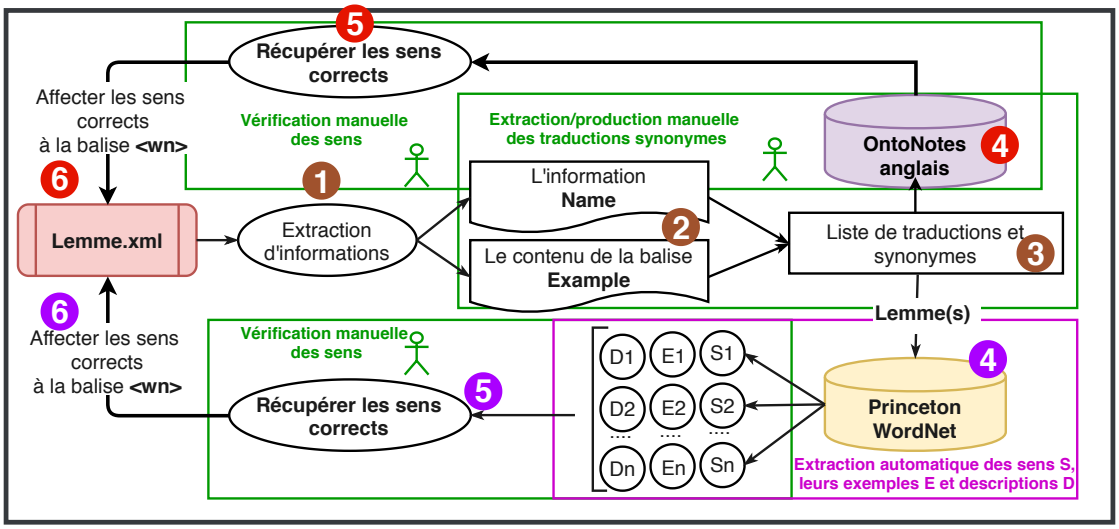

FIGURE 4.2: Étapes pour ajouter les sens *Princeton WordNet* dans Ontonotes

```
<sense n="1" type="" name="Member,taking an active role in a group الانتماء الفاعل لجماعة " group="1">
    <commentary>Plural OaEoDA2</commentary>
    <examples>
        أعضاء مجلس النواب المصري
        The members of the Egyptian parliament

        ليس كل مناهض لسياسة امريكا الخارجية عضو في منظمة ارهابية
        Not everyone who opposes the American foreign policy is a member of some terrorist organization.

        لتكون عضوا في النادي عليك دفع اشتراك شهري
        To be a member in a club one must pay a monthly fee.
    </examples>
    <mappings>
    <wn version="3">
                    member%1:18:00::        <-- Sens WordNet ajouté(s)
                    member%1:14:00::
                    fellow_member%1:18:00::
    </wn>
     <omega></omega>
       <pb></pb>
    </mappings>
```

FIGURE 4.3: Exemple du premier sens dans le document mis à jour EuDow-n.xml (Membre)

Ce traitement semi-automatique d'annotation et de vérification réalisé s'est avéré coûteux en temps (quatre mois de travail). Le tableau 4.2 présente la description d'*OntoNotes Release 5.0* ainsi que le nombre de correspondances$_{WordNet}$ uniques ajoutées.

	#Lemmes	#Lemmes uniques	#Sens uniques	#Correspondances$_{WordNet}$ uniques
Verbes	3 990	150	642	4 182
Noms	8 534	111	463	1 376
Total	12 524	261	1 105	5 558

TABLE 4.2: Description d'*OntoNotes Release 5.0* après l'ajout des correspondances vers le *Princeton WordNet* 3.0

Après avoir enrichi le corpus de référence OntoNotes Release 5.0, nous procédons à la mise en œuvre de notre méthode proposée pour construire des corpus annotés par traduction automatique afin de créer rapidement un système de désambiguïsation lexicale supervisé de l'arabe et qui consiste à traduire les 12 corpus anglais UFSAC (présentés dans le tableau 4.4) annotés manuellement en sens et à porter leurs annotations grâce à un système de traduction de l'anglais (une langue riche en corpus annotés) vers une langue cible (ici l'arabe).

Ainsi, nous présentons dans ce qui suit les systèmes de traduction automatique anglais-arabe (statistique et neuronale) que nous avons mis en place.

4.3 Systèmes de traduction anglais-arabe

Comme illustré dans la figure 4.4, notre méthode de traduction et portage des annotations se compose de trois étapes principales : (1) pré-traitement des corpus anglais, (2) traduction et portage de ses annotations vers l'arabe, (3) post-traitement qui, appliqué aux traductions produites, résout des problèmes induits par la traduction. Chacune de ces étapes est mise en œuvre au moyen d'un script écrit en *Python*.

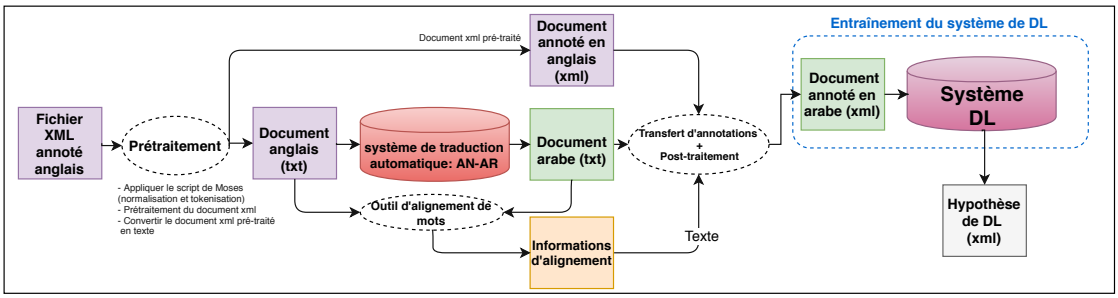

FIGURE 4.4: Notre projection interlingue du protocole d'annotation des sens

Par ailleurs, nous utilisons l'ensemble des données décrites dans la section 3.3.5 comme données d'entraînement afin de construire nos systèmes de traduction automatique statistique et neuronal anglais-arabe. Le tableau 4.3 présente la taille des différents corpus pour le Train, le Dev et le Test.

Corpus	**#Lignes**	**#Mots anglais**	**#Mots arabes**
MultiUN	9 464K	253 284K	223893K
TED	150K	2 925K	2 302K
News-Commentary	54K	2 499K	2 879K
Ummah	57K	1 991K	1 626K
News	16K	519K	389K

TABLE 4.3: Description des corpus parallèles utilisés pour l'entraînement de notre système de traduction automatique

4.3.1 Prétraitement des corpus

En arabe nous trouvons plusieurs clitiques qui se collent au mot, conduisant à des ambiguïtés morphologiques et orthographiques comme il est indiqué dans le mot source de la Figure 4.5.

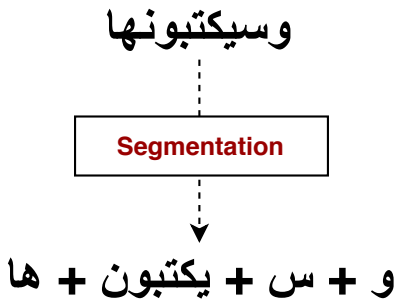

FIGURE 4.5: Exemple de segmentation d'un mot arabe au niveau des clitiques

Ainsi, pour construire un système de traduction anglais-arabe, il est nécessaire de passer par une étape de segmentation du corpus au niveau des mots en pré-traitement De ce fait, il est important de trouver le bon schéma de tokenisation à suivre qui ne se trompe pas en détectant le token et les clitiques. Dans notre travail, nous nous intéressons aux schémas de tokenisation, proposés par Habash and Sadat [2006] pour le pré-traitement de l'arabe. Nous utilisons la technique *ATB* (présentée dans la section 1.3.2.1) qu'ils jugent comme étant le meilleur schéma de tokenisation.

Afin d'avoir des données propres, nous avons prétraité nos corpus d'entraînement comme suit :

— Garder les phrases contenant entre 5 et 50 mots pour les corpus arabe et anglais
— Convertir les entités HTML
— Supprimer les balises XML et les lignes contenant que des liens ou des chiffres
— Normalisation des ponctuations
— Convertir en minuscule et normaliser le corpus Anglais : (it's ⇒ it 's)
— Normalisation des caractères arabes (ا ،إ ،أ ،آ «Alif», ى ،ي «Ya», etc)
— Enlever la voyelisation pour les caractères arabes.

D'autre part, afin de traduire des données à l'aide de notre système de traduction automatique, celles-ci doivent être normalisées pour être dans le même format que les données d'entraînement du système. Nous trouvons en effet dans les corpus à traduire des mots composés avec tiret bas, des mots non tokenisés, des mots commençant par une majuscule au début d'une phrase, etc. Ainsi, nous proposons une normalisation pour les données à traduire, et qui se réalise en trois étapes :

— Segmenter les mots composés (effacer le tiret bas)

written_language ⇒ written language

— Appliquer la tokenisation Moses (ajouter des espaces entre mots et ponctuation :

People's ⇒ People 's

— Mettre chaque mot du corpus dans une balise en suivant le format du corpus en lui affectant un identificateur (id) unique.

La Figure 4.6 présente un exemple de normalisation appliquée au mot composé "written_language" :

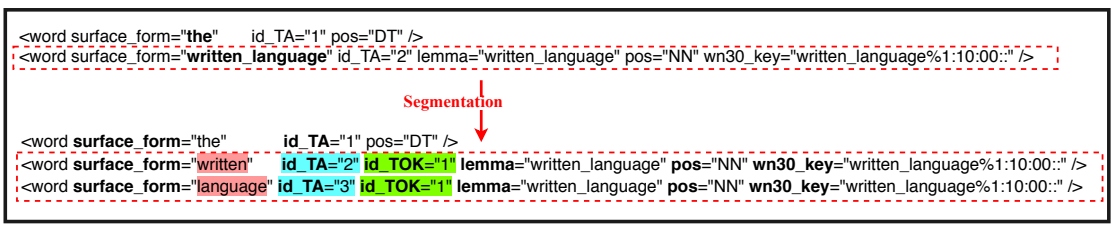

FIGURE 4.6: Exemple de normalisation du mot composé "written_language"

4.3.2 Traduction et portage des annotations

Dans cette section, nous décrivons les deux systèmes de traduction automatique anglais-arabe en utilisant une approche statistique et une approche neuronale que nous avons construits. Ces systèmes ont été appris, optimisés et évalués (en termes de score BLEU) respectivement sur les corpus Train, Dev et Test. Le tableau 4.3 décrit les corpus parallèles utilisés pour l'apprentissage. Les corpus Dev et Test présentent 800 lignes de chaque corpus. Les corpus Train/Dev/Test sont normalisés avec le schéma de tokenisation *ATB*.

4.3.2.1 Approche statistique

Le système de traduction automatique statistique anglais-arabe a été produit à l'aide de la boîte à outils Moses (voir la section 3.3.1.3), l'outil IRSTLM (voir la section 3.3.1.1) pour créer un modèle de langage 5-grammes (appris sur la partie arabe du corpus Train) et l'outil d'alignement Giza++. En évaluant le système produit sur le corpus de Test, nous obtenons 27,51% en termes de score BLEU.

Ainsi, le système créé a été utilisé pour traduire les 12 corpus UFSAC de l'anglais vers l'arabe. Nous exploitons ensuite les informations d'alignement des mots cible-source fournis par Moses pour transférer les annotations d'un mot source anglais vers son correspondant dans la traduction arabe. La Figure 4.7 présente un exemple de traduction et portage des annotations en arabe d'un mot composé pré-traité.

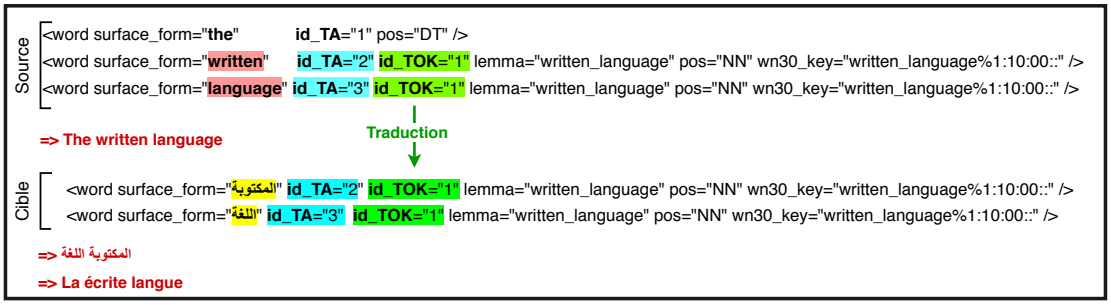

FIGURE 4.7: Exempde traduction et portage d'annotations du mot composé "written_language"

4.3.2.2 Approche neuronale

Le système de traduction automatique neuronal anglais-arabe a été appris sur le corpus Train et optimisé sur le corpus Dev à l'aide de l'outil seq2seq [2] pour construire notre système NMT en exploitant les données d'entraînement décrites dans le tableau 4.3. Nous avons évalué notre système NMT sur le corpus de Test qui a obtenu 28.66% en termes du BLEU.

Pour le transfert des annotations, il est nécessaire de produire des informations d'alignement. Pour ce faire, nous avons utilisé l'outil *fast_align* [3] Dyer et al. [2013], qui est un modèle d'alignement open-source [4]. Avant l'entraînement, nous avons appliqué un traitement spécifique pour toutes les données monolingues en utilisant les symboles de sous-mots [Sennrich et al., 2015, Chung et al., 2016, Luong and Manning, 2016].

Ainsi, nous utilisons les informations d'alignement de mots source-cible fournies par *fast_align* afin de transférer des annotations d'un texte source anglais à son correspondant dans le texte cible arabe. Plus précisément, comme décrit dans la figure 4.4, après avoir effectuer le pré-traitement de notre corpus, nous effectuons les étapes de traduction et de transfert d'annotations suivantes :

1. Extraire les données textuelles du corpus XML pré-traité.

2. Traduire le corpus textuel de l'anglais vers l'arabe, en utilisant notre système de traduction automatique neuronal.

3. Construire des informations d'alignement avec *fast_align* en utilisant à la fois les corpus de la langue source et de la langue cible.

4. Appliquer la chaîne de traitement que nous avons proposé pour le transfert d'annotation afin d'obtenir un fichier XML traduit pour la tâche de désambiguisation lexicale.

2. Disponible en ligne :https://github.com/eske/seq2seq.git. Les paramètres sont : Adam comme optimiseur ; taille d'embeddings=600 ; batch_size=64 ; la longueur maximale des séquences d'entrée et de sortie=80 ; 2 couches LSTM avec une taille de cellule=512 ; taille de la couche d'attention=512.

3. https://github.com/clab/fast_align

4. Nous avons choisi de ne pas utiliser l'information soft-alignment donnée par le mécanisme d'attention du système MT neuronal car nous avons remarqué manuellement que la qualité de l'alignement produit était pire, probablement en raison de la segmentation spécifique effectuée sur la langue arabe.

4.3.3 Post-traitement

L'exemple de traduction présenté dans la figure 4.7 montre clairement l'un des problèmes posé par l'étape de traduction. Il est nécessaire de passer par un post-traitement pour obtenir des résultats les plus pertinents possibles. Ainsi, l'outil de portage d'annotations produit parfois des traductions mal ordonnées (Figure 4.7) ou dupliquées car il se base seulement sur les alignements en mots fournis par le décodeur ou par l'outil d'alignement *fast_align* et en aucun cas sur la sortie de traduction. Ainsi, afin de résoudre ces problèmes, nous avons développé un outil permettant de compiler une chaîne de post-traitement sur la sortie de traduction, et qui enchaîne les trois étapes suivantes :

— Réordonnancement et suppression des mots ajoutés au moment de l'alignement : afin de détecter les erreurs d'alignement en mot fournis par l'outil d'alignement, nous avons commencé par vérifier le positionnement de chaque mot annoté suivant la cible de traduction. Ainsi, si nous détectons un problème sur l'ordre de mots, nous faisons le réordonnancement sinon, nous supprimons le mot ajouté. La Figure 4.8 présente un exemple de réordonnancement d'une traduction mal ordonnée (Figure 4.6).

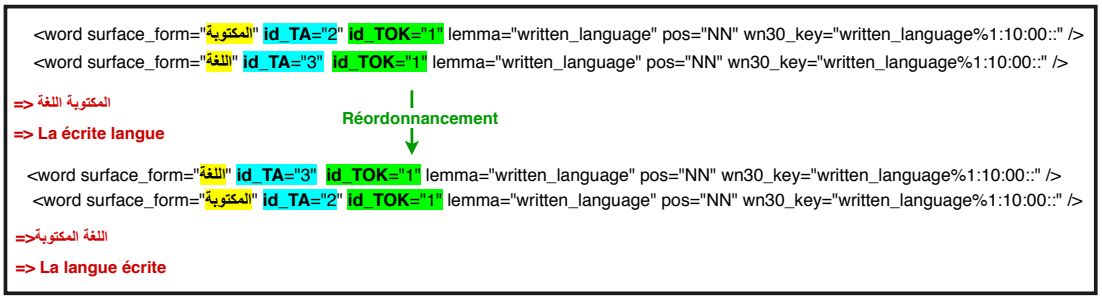

FIGURE 4.8: Réordonnancement des mots suivant la cible

— Concaténation : suite à la tokenisation des mots composés (ayant le même id) dans la chaîne de pré-traitement du SemCor, nous concaténons ces derniers afin d'obtenir des id uniques. Dans la Figure 4.9, nous donnons un exemple précis de concaténation de mots ayant l'id=3 après le réordonnancement de la traduction en arabe du mot composé «written_language»

— Segmentation *D3* (voir la section 1.3.2.1 : cette étape nous permet de séparer les proclitiques, les clitiques PART, ainsi que touts les articles et enclitiques du mot afin d'avoir la forme fléchie en sortie.
La figure 4.10 présente un exemple de segmentation du mot الوقت afin de séparer

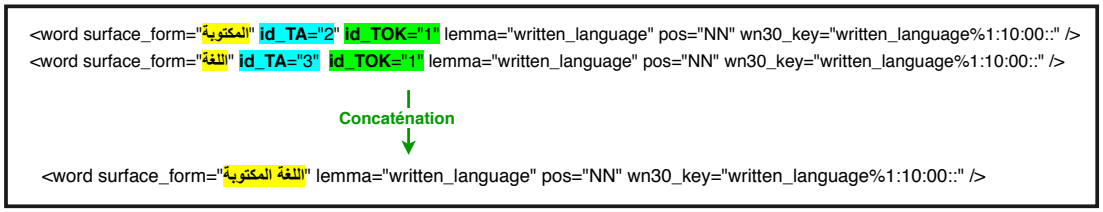

FIGURE 4.9: Concaténation des mots suivant la cible

l'article défini الـ du forme fléchie وقت.

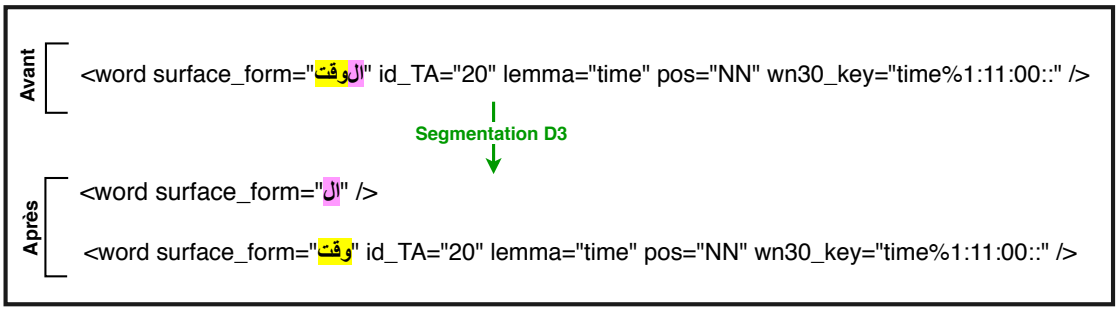

FIGURE 4.10: Segmentation D3 en utilisant MADAMIRA

Par ailleurs, comme il est indiqué précédemment, à cause des informations d'alignement fournies par l'outil d'alignement, l'outil de portage d'annotation utilisé produit non seulement des traductions mal ordonnées, mais aussi des traductions parfois dupliquées comme il est présenté dans la figure 4.11.

La figure montre que notre outil de post-traitement a détecté que la séquence de mots مدينة مدينة اطلنطا (en appliquant le processus de Concaténation) n'existe pas dans la sortie de traduction, et que le mot a été dupliqué à cause des informations d'alignement Ainsi, il a cherché la séquence de mots exacte dans la cible et a supprimé le mot dupliqué ayant l'identifiant id_TA="29".

Cette méthode a été appliquée aux douze corpus d'*UFSAC* et nous fournirons à la communauté les 11 dont la licence permet le partage et le code permettant de convertir le douzième (le DSO) pour ceux qui l'auraient acquit.

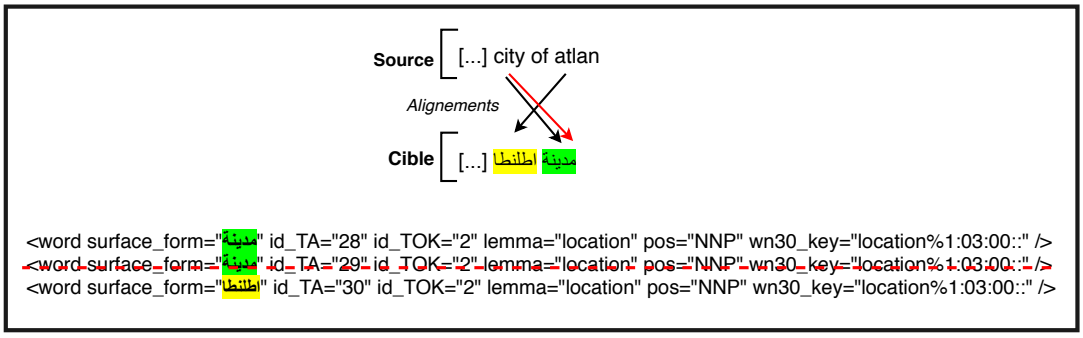

FIGURE 4.11: Processus de post-traitement pour les mots dupliqués

4.4 Production de corpus arabes pour la désambiguïsation lexicale supervisée

Dans cette section, nous présentons l'ensemble de nos corpus UFSAC en arabe annotés en sens (nommé UFSAC-ara) après avoir appliqué notre méthode de création de ressources en utilisant un système de traduction automatique neuronal et le transfert direct d'annotations d'une langue source riche en corpus annotés comme l'anglais vers une langue cible moins bien dotée. Sachant que, cette méthode pourrait être utilisée pour toute langue, du moment que l'on dispose d'une traduction de l'anglais vers cette langue. Le tableau 4.4 présente les informations relatives aux corpus UFSAC-eng originaux tandis que le tableau 4.5 présente les informations relatives aux corpus UFSAC-ara.

Ressource	Phrases	Mots		Parties du discours annotées			
		Total	Annotés	Noms	Verbes	Adjectifs	Adverbes
SemCor	37 176	778 587	229 533	87 581	89 051	33 752	19 149
DSO	101 004	2 705 190	176 197	105 245	70 952	0	0
WNGT	117 659	1 634 691	496 776	287 798	77 234	107 135	24 609
MASC	31 760	585 354	113 546	49 474	39 356	12 894	11 822
OMSTI	820 084	35 800 061	920 357	476 692	253 555	190 110	0
OntoNotes	124 851	2 475 926	233 616	79 765	153 851	0	0
SemEval 2007 task 07	245	5 637	2 261	1 108	591	356	206
SemEval 2007 task 17	126	3 438	455	159	296	0	0
SemEval 2 013 task 12	306	8 142	1 644	1 644	0	0	0
SemEval 2015 task 13	138	2 637	1 053	554	251	166	82
Senseval 2	238	5 589	2 301	1 061	541	422	277
Senseval 3 task 1	300	5 507	1 957	886	723	336	12
Total	1 233 649	44 010 759	2 179 696	1 091 967	686 401	345 171	56 157

TABLE 4.4: Informations relatives aux corpus UFSAC-eng Vial et al. [2017]

Ressource	Phrases	Mots		Parties du discours annotées			
		Totaux	Annotés	Noms	Verbes	Adjectifs	Adverbes
SemCor	37 176	767 415	208 142	80 552	80 079	29 977	17 534
DSO	101 004	2 494 012	166 436	99 933	66 503	0	0
WNGT	117 659	1 586 199	456 880	267 985	69 886	96 299	22 710
MASC	31 760	548 645	102 614	45 093	35 283	11 543	10 695
OMSTI	820 084	28 455 324	842 999	437 487	229 976	175 536	0
OntoNotes	124 851	2 331 961	216 283	75 354	140 929	0	0
SemEval 2007 Task 7	245	5 589	1 985	997	503	305	180
SemEval 2007 task 17	126	3 012	380	135	245	0	0
SemEval 2013 task 12	306	7 709	1 439	1 439	0	0	0
SemEval 2015 task 13	138	2 677	959	504	235	144	76
SensEval 2	238	5 741	2 063	973	492	364	234
SensEval 3	300	5 493	1 738	806	640	281	11
Total	1 233 649	36 213 777	2 001 918	1 011 258	624 771	314 449	51 440

TABLE 4.5: Informations relatives à notre ensemble de corpus en langue arabe annotés en sens

Par conséquent, en comparant les deux tableaux 4.4 et 4.5, nous remarquons que après la traduction et le transfert des annotations des 12 corpus UFSAC de l'anglais vers l'arabe, il y a une diminution en termes de quantité de données pour la totalité des mots (de 44M à 36M mots) ainsi que ceux qui sont annotés.

4.5 Conclusion

Dans ce chapitre nous avons d'abord, présenté le corpus OntoNotes 5.0 manuellement annoté en sens, contenant comme langues l'anglais, le chinois ainsi que l'arabe. Cependant, le travail d'annotation de la partie arabe n'était pas complet, puisqu'il manquaient le lien vers la ressource lexicale WordNet anglais. Ainsi, nous avons décrit la méthode suivie pour mettre à jour la partie arabe du corpus, après avoir indiqué les diverses approches d'alignement de ressources interlingues. Ensuite, nous avons présenté les systèmes de traduction automatique statistique et neuronal anglais-arabe construits, ainsi que les étapes de prétraitement, de traduction et portage d'annotations et de post-raitement que nous avons appliquées sur nos données. Enfin, nous avons introduit l'ensemble des 12 corpus arabes annotés en sens que nous avons produits, nécessaires pour la tâche de désambiguïsation lexicale de l'arabe.

5

Utilisation des ressources

5.1 Introduction

Après avoir produit l'ensemble de nos corpus UFSAC arabes annotés en sens, nous procédons à la mise en œuvre des systèmes de désambiguïsation lexicale pour l'arabe en exploitant nos ressources. Dans un premier temps, nous présentons deux systèmes de désambiguïsation lexicale pour l'arabe : un système basé sur les séparateurs à vaste marge (SVM), et un autre basé sur les réseaux de neurones (LSTM), entraînés sur les corpus que nous avons produits. Ces deux systèmes sont évalués sur le corpus OntoNotes 5.0 arabe annoté manuellement en sens *Princeton WordNet*. En outre, afin de prouver l'efficacité de notre méthode de traduction et portage des annotations, nous allons nous comparer à un système de désambiguïsation lexicale anglais SVM et un autre neuronal existants développés au sein de l'équipe GETALP du LIG par Vial et al. [2018a], entraînés sur les corpus UFSAC anglais originaux, et évalués sur différents corpus. Dans un deuxième temps, nous étudions l'impact de la tâche de désambiguïsation lexicale de l'arabe sur la traduction automatique.

Dans ce chapitre, nous nous intéressons aussi à créer des nouveaux systèmes de traduction factorisés où les modèles génèrent à la fois le mot ainsi que les facteurs (caractéristiques) associés, afin d'évaluer l'effet des informations de sens (issues des systèmes de désambiguïstion lexicale de l'arabe et de l'anglais) et des informations morphologiques (lemme et partie de discours). Ces systèmes seront comparés par la suite avec un nouveau système de traduction neuronal de base entraîné sur les 4 corpus : TED, News-Commentary, Ummah et News.

5.2 Système de désambiguïsation basé sur les séparateurs à vaste marge

5.2.1 Méthodologie

L'apprentissage automatique consiste à entraîner un classifieur pour chaque mot cible dans le but de prédire le sens le plus pertinent dans son contexte. Les algorithmes supervisés entraînent un classifieur sur les corpus annotés en sens : classifieur séparateurs à vaste marge (NUS-PT Chan et al. [2007]), classifieur naïfs bayésiens (NUS-ML, Cai et al. [2007]), combinaison de séparateurs à vaste marge, entropie maximale (LCC-WSD, Novischi et al. [2007]). Nous ne pouvons pas vraiment affirmer que tel ou tel

classifieur soit meilleur qu'un autre (avant l'émergence des réseaux de neurones) et ce qui différencie les performances des systèmes est principalement et directement lié à la taille des données annotées [Navigli et al., 2007, Schwab, 2017].

Pour ce travail [1], nous avons implémenté le classifieur utilisé dans le système NUS-PT qui était le système supervisé état de l'art avant l'émergence des réseaux de neurones profonds. Nous avons fait ce choix pour deux raisons : premièrement afin de prouver la pertinence de l'approche, et deuxièmement, pour utiliser un système de calcul moins gourmand en ressources et donc accessible à un plus grand nombre de chercheurs.

Notre classifieur se base sur trois ensembles de traits pour assigner un sens à un mot donné :

1. Les parties du discours des mots voisins (P_i), 7 traits sont extraits, qui correspondent aux labels de partie du discours des trois mots à gauche (P_{-3}, P_{-2}, P_{-1}) trois mots à droite (P_1, P_2, P_3) et à celui du mot cible (P_0) ;
2. les collocations locales ($C_{i,j}$), qui correspondent à la suite ordonnée des mots entre les index i et j relativement au mot cible, mis en lettres minuscules. 11 traits sont ainsi extraits : $C_{-1,-1}$, $C_{1,1}$, $C_{-2,-2}$, $C_{2,2}$, $C_{-2,-1}$, $C_{-1,1}$, $C_{1,2}$, $C_{-3,-1}$, $C_{-2,1}$, $C_{-1,2}$ et $C_{1,3}$;
3. le contexte voisin, ce trait correspond à un vecteur de la taille du nombre de lemmes différents observés pendant l'entraînement. Chaque composante du vecteur correspond ainsi à un lemme, et sa valeur est mise à 1 si le lemme d'un des mots présent dans la même phrase que le mot cible correspond au lemme de cette composante. Elle vaut 0 sinon.

5.2.2 Évaluation

Pour l'évaluation, nous utilisons la partie arabe que nous avons mise à jour d'OntoNotes Release 5.0 et les mesures d'évaluation classiques de la désambiguïsation lexicale telle qu'utilisées dans SemEval 2013 : précision P, rappel R et score F1.

5.2.3 Résultats et analyse basée sur les SVM

Dans le précédant chapitre, nous avons complété le corpus de référence OntoNotes Release 5.0 pour la désambiguïsation lexicale de l'arabe. Ce corpus sera maintenant

1. https://hal.archives-ouvertes.fr/hal-01781185/document

exploité pour comparer notre système de désambiguïsation lexicale de l'arabe basé sur les séparateurs à vaste marge (entraîné sur les données UFSAC arabes annotés en sens) avec un système de désambiguïsation lexicale de l'anglais maison créé au sein de l'équipe GETALP du LIG (entraîné sur les données UFSAC anglais originales) et évalué sur différents corpus. Cette approche permettra de prouver l'efficacité de notre méthode et ainsi évaluer l'influence de l'étape de traduction et le portage des annotations sur la qualité de la désambiguïsation lexicale de l'arabe.

5.2.3.1 Résultats de la désambiguïsation lexicale SVM anglais

Le système de désambiguisation lexicale SVM anglais nommé SVM-UFSAC-eng a été entraîné sur les corpus UFSAC originaux en anglais (Semcor, DSO, WNGT, MASC, OMSTI et OntoNotes anglais) comportant 1,99M mots annotés. Ensuite, ce système a été évalué sur différents corpus (*SensEval*2, *SensEval*3*task*1 et *semeval*2007*task*17) pour comparer les résultats.

Système	Tâche	Précision	Rappel	Score F1
SVM-UFSAC-eng	SensEval2	71,34	69,57	70,45
SVM-UFSAC-eng +repli	SensEval2	71,88	71,88	**71,88**
IMS+emb	SensEval2	68,3	68,3	68,3
SVM-UFSAC-eng	SensEval3task1	65,32	59,58	62,31
SVM-UFSAC-eng+repli	SensEval3task1	65,50	65.50	**65.50**
IMS+emb	SensEval3task1	682	68,2	68,2
SVM-UFSAC-eng	semeval2007task17	60,92	60,65	60,79
SVM-UFSAC-eng+repli	semeval2007task17	60,87	60,87	**60,87**
IMS+emb	semeval2007task17	68,2	68,2	59,7

TABLE 5.1: Performances du système de désambiguïsation lexicale sur l'anglais. Le repli est effectué sur le premier sens dans WordNet.

D'après le tableau ci-dessous, nous remarquons que le système SVM-UFSAC-eng a obtenu les meilleurs performances en termes de score F1 sur les corpus *SensEval*2, *SensEval*3*task*1 et *semeval*2007*task*17, respectivement 71,88%, 65,50% et 60,87%, en ajoutant le repli vers le premier sens.

D'autre part, nous avons reporté dans le tableau les résultats du meilleur système de désambiguïsation supervisé état de l'art [Iacobacci et al., 2016] (nommé IMS+emb). Nous observons que les performances de SVM-UFSAC-eng sont comparables.

Dans ce qui suit, nous présentons nos expériences sur la langue arabe, nous montrons que nos résultats sont convenables et encourageants comparés à l'anglais.

5.2.3.2 Résultats de la désambiguïsation lexicale SVM arabe

Nous avons réalisé l'entraînement de notre algorithme de désambiguïsation lexicale SVM (voir section 5.2.1) sur les douze corpus traduits depuis l'anglais. Nous appelons ainsi ce système SVM-UFSAC. Le tableau 5.2 présente les résultats de notre système. Comme beaucoup d'algorithmes de désambiguïsation lexicale, SVM-UFSAC n'annote pas l'ensemble des termes. Par exemple, si les corpus annotés ne contiennent pas d'exemple pour un mots à étiqueter, il ne peut pas réaliser cette opération. Nous utilisons alors l'heuristique classique qui consiste à choisir le premier sens du *Princeton WordNet*.

	Précision	Rappel	Score F1
SVM-UFSAC	68.60%	62.14%	65.21%
SVM-UFSAC+ repli premier sens	67.55%	62.74%	65.06%
SVM-UFSAC+Post-traitement	**70.86**%	64.20%	**67.36**%
SVM-UFSAC+Post-traitement + repli premier sens	69.75%	**64.79**%	67.18%

TABLE 5.2: Performance de notre système de désambiguïsation lexicale arabe

Notre système de désambiguïsation arabe a été évalué en termes de *Précision, Rappel* et *Score F1* sur le corpus de référence OntoNotes arabe que nous avons mis à jour. Il convient de noter que notre système a réussi donc à désambiguïser tous les mots annotés (12 524 mots).

En outre, en ajoutant le repli vers le premier sens, notre système de désambiguïsation obtient une meilleure performance en termes de score F1 56.62% (+6.01%), plus précisément, notre système a été capable de désambiguïser correctement 7 073 mots parmi les 12 457 mots désambiguïsés.

Comme le montre le tableau 5.2, nous pouvons remarquer tout d'abord, qu'en ajoutant le repli vers le premier sens pour les mots non annotés, le système SVM-UFSAC a obtenu 67.55% en termes de précision et et 65.06% en termes de score F1. En outre, en appliquant les différentes étapes de post-traitement décrites précédemment sur nos données traduites en arabe, notre système de désambiguïsation obtient une meilleure performance en termes de précision 70.86% (+3.31%) , c'est-à-dire qu'il a été capable

de désambiguïser correctement 8 040 mots parmi les 12 524 mots annotés, et en termes de score F1 67.36% (+2.30%).

Par conséquent, nous pouvons dire que nous avons obtenu des résultats de désambiguïsation lexicale arabe similaires aux résultats obtenus pour l'anglais sachant que les corpus d'entraînement utilisés pour les deux langues ont presque la même taille, ce qui prouve l'efficacité de notre méthode.

5.3 Système de désambiguïsation basé sur les réseaux neuronaux

5.3.1 Architecture du réseau neuronal pour la désambiguïsation lexicale

Parmi les ouvrages existants sur les approches neuronales pour la désambiguïsation lexicale, et d'ailleurs ceux les plus connus, Yuan et al. [2016] ont utilisé un réseau neuronal à base de *LSTM* comme modèle de langue, afin de prédire le mot contenu dans une séquence en fonction de son contexte. Ils ont réalisé un apprentissage supervisé sur des corpus annotés en sens pour entraîner le système à distinguer les différents sens d'un mot en fonction des mots prédits par leur modèle de langue. Ensuite, ils ont proposé une méthode de propagation des étiquettes afin d'avoir plus de données annotées en sens et obtenir des meilleurs résultats.

Raganato et al. [2017] ont proposé quant à eux un modèle fondé sur un réseau RNN de type *LSTM* permettant de prédire une étiquette pour chacun des mots en entrée. L'étiquette à prédire appartient à un ensemble comportant tous les sens possibles dans un dictionnaire ainsi que tous les mots observés au cours de l'apprentissage. Ils ont enrichi ensuite leur architecture avec une couche d'attention, et ont réalisé un entraînement multi-tâches dans lequel leur réseau prédit à la fois un sens ou un mot, une étiquette de partie du discours, ainsi qu'une étiquette sémantique.

Au sein de notre équipe GETALP du LIG, Vial et al. [2018b] ont proposé une nouvelle architecture à base de réseaux de neurones pour la désambiguïsation lexicale. Ils ont créé un système de désambiguïsation lexicale anglais moins complexe à entraîner que les systèmes neuronaux existants tout en produisant des résultats état de l'art sur la plupart des tâches d'évaluation de la désambiguïsation lexicale en anglais. Ils ont consi-

déré la désambiguïsation lexicale comme un problème de classification dans lequel à chaque mot est assigné une étiquette. Pour cela, ils ont utilisé un modèle de vecteurs de mots, un corpus d'apprentissage et d'évaluation librement accessibles. Ils ont simplifié le modèle de Raganato et al. [2017] en considérant une étiquette comme appartenant uniquement à l'ensemble de tous les sens possibles de leur inventaire de sens.

Dans notre travail, nous avons suivi la même approche que celle utilisée pour le système de désambiguïsation lexicale de l'anglais de Vial et al. [2018b] afin de construire notre système de désambiguïsation lexicale neuronale de l'arabe. Cette approche possède l'architecture illustrée dans la figure 5.1 et qui repose sur les trois couches suivantes :

— Une couche d'entrée : qui prend en entrée les représentations vectorielles (Word-Embeddings) des mots pré-entrainés. Le modèle Word-Embeddings a été appris sur la partie mono-lingue arabe de notre corpus de traduction automatique à l'aide l'outil Word2Vec [Mikolov et al., 2013b].

— Une couche cachée : composée de cellules LSTM (voir la section 3.3.2.2.2) bi-directionnelles.

— Une couche de sortie, permettant de générer une distribution de probabilités sur tous les sens possibles du dictionnaire, pour chaque mot en entrée, tout en utilisant la fonction Softmax (décrite dans la section 3.3.2) qui permet d'assurer que la somme de toutes les probabilités est égale à 1.

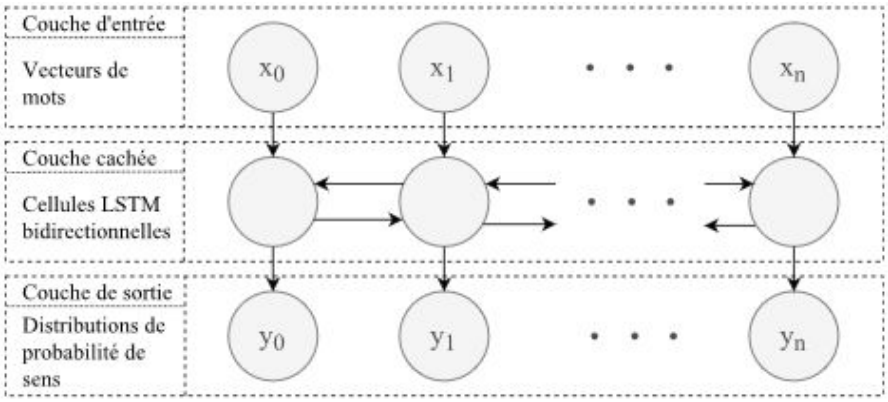

FIGURE 5.1: Architecture neuronale pour la désambiguïsation lexicale (Vial et al. [2018b])

Le modèle a pour objectif de prédire un sens pour chaque mot en entrée du réseau, voire les mots outils ou les mots non annotés dans le corpus d'entraînement. Sachant

que, pour traiter ces derniers, nous utilisons le symbole spécial <*skip*> qui permet d'ignorer les prédictions réalisées par le modèle et de ne pas les considérer au cours de la phase de rétro-propagation durant l'apprentissage.

5.3.2 Protocole expérimental

Dans cette section nous présentons le processus d'entraînement utilisé dans la désambiguïsation de l'arabe et de l'anglais, ainsi que la méthodologie utilisée pour désambiguïser un document.

5.3.2.1 Processus d'entraînement et détails de mise en œuvre

Tout comme pour le système de désambiguïsation lexicale anglais [Vial et al., 2018b], nous avons utilisé des cellules LSTM de taille de 1000 par direction (2000 au total) pour la couche cachée de neurones récurrents. Également, nous avons appliqué une régularisation de type *Dropout* [Srivastava et al., 2014] à 50% entre la couche cachée et la couche de sortie, afin d'empêcher le surapprentissage pendant l'entraînement et rendre ainsi le modèle plus robuste.

Nous présentons ainsi les paramètres utilisés pour l'apprentissage :

— La fonction objectif à minimiser est l'entropie croisée entre la prédiction faite par le modèle, et un vecteur type *one-hot* pour lequel toutes les composantes sont à 0, sauf à l'index du sens à prédire où elle est à 1
— La méthode d'optimisation *Adam* [Kingma and Ba, 2014a], en utilisant les paramètres standards
— Des mini-lots de taille 30
— Les phrases ont été réduites à 50 mots, en ajoutant des vecteurs nuls pour les phrases ayant un nombre de mots inférieur à 50.

Le réseau neuronal a été construit à l'aide de l'outil *PyTorch* [2]. Nous avons effectué l'entraînement pendant 20 *epochs*. Une *epoch* correspondant à une passe complète sur nos données d'entraînement. Nous avons évalué périodiquement (tous les 2000 mini-lots) notre modèle sur le corpus de développement, et nous avons conservé uniquement le modèle ayant obtenu le plus grand score F1 de désambiguïsation.

2. http://pytorch.org/

5.3.2.2 Processus de désambiguïsation neuronale

Nous avons utilisé l'approche décrite ci-dessous afin d'effectuer la désambiguïsation lexicale d'une séquence de mots en utilisant le réseau entraîné :

1. Convertir chaque mot en entrée du réseau en une représentation vectorielle à l'aide du modèle de Word-Embeddings.
2. Retourner en sortie le sens le plus probable en fonction de son lemme et sa catégorie grammaticale depuis la ressource *Princeton WordNet*.
3. Appliquer la technique du repli vers le premier sens si aucun sens n'est attribué. Cette technique consiste à attribuer le sens le plus fréquent dans *Princeton WordNet*.

5.3.3 Évaluation

Nous évaluons notre approche *in vitro* (en termes de précision P, de rappel R et de score F1) en nous basant sur un système de désambiguïsation lexicale neuronal, tout en exploitant la partie arabe du corpus OntoNotes Release 5.0 (annoté en sens anglais) que nous avons enrichi comme corpus de référence.

5.3.4 Résultats et analyse de la désambiguïsation lexicale neuronale

Dans cette section, nous nous intéressons à comparer notre système de désambiguïsation lexicale neuronal de l'arabe entraîné sur les corpus UFSAC-ara avec le système de désambiguïsation lexicale neuronal de l'anglais maison entraîné sur les corpus originaux UFSAC-eng. Dans ce qui suit, nous présentons les systèmes ainsi que les résultats de la désambiguïsation lexicale neuronale de l'anglais et de celle de l'arabe.

5.3.4.1 Résultats de la désambiguïsation lexicale neuronale de l'anglais

Vial et al. [2018b] ont construit un système de désambiguïsation lexicale de l'anglais entraîné sur les 6 corpus issus de UFSAC anglais (présentés dans la section 4) : SemCor, DSO, WordNet Gloss Tagged, OMSTI, MASC et OntoNotes anglais. Le corpus de SemEval 2015 task 13 a été exploité comme corpus de développement durant

l'entraînement, pour éviter le surapprentissage des données d'entraînement. Comme, ils ont utilisé le modèle ayant obtenu le meilleur score F1 de désambiguïsation lexicale sur le corpus de développement pour l'évaluation de leur système de désambiguïsation lexicale sur les différents corpus : SensEval 2, SensEval 3, SemEval 2007 task 07, SemEval 2007 task 17, et SemEval 2013 task 12.

Ils ont également évalué leur modèle sur tous les corpus d'évaluation communément utilisés en désambiguïsation lexicale , à savoir les tâches de désambiguïsation lexicale des campagnes d'évaluation SensEval/SemEval. Ils ont comparé les résultats de leur système aux systèmes semblables de l'état de l'art à base de réseaux de neurones [Yuan et al., 2016, Raganato et al., 2017], ainsi que l'étalon du sens le plus fréquent, et du meilleur système avant l'émergence des réseaux de neurones en désambiguïsation lexicale [Iacobacci et al., 2016]. Le tableau 5.3 présente les différents scores.

Système	SE2	SE3	SE07 (07)	SE07 (17)	SE13 (12)	SE15 (13)
Notre système	**73.75%**	70.31%	**83.59%**	60.22%	68.98%	*73.98%
Yuan et al. [2016] (LSTM)	73.6%	69.2%	82.8%	**64.2%**	67.0%	72.1%
Yuan et al. [2016] (LSTM + LP)	**73.8%**	**71.8%**	**83.6%**	63.5%	**69.5%**	**72.6%**
Raganato et al. [2017] (BLSTM)	71.4%	68.8%	-	*61.8%	65.6%	69.2%
Raganato et al. [2017] (BLSTM + att. + LEX + POS)	72.0%	69.1%	83.1%	*64.8%	66.9%	71.5%
Sens le plus fréquent	65.6%	66.0%	78.89%	54.5%	63.8%	67.1%
Iacobacci et al. [2016]	68.3%	68.2%	-	59.1%	-	-

TABLE 5.3: Scores F1 obtenus par le système de Vial et al. [2018b] sur les tâches de désambiguïsation lexicale de l'anglais des campagnes d'évaluation SensEval 2 (SE2), SensEval 3 (SE3), SemEval 2007 (SE07) task 07 et 17, SemEval 2013 (SE13) task 12 et SemEval 2015 (SE15) task 13. Les résultats précédés d'une étoile (*) sont obtenus sur le corpus de développement utilisé lors de l'entraînement

Le premier système de Yuan et al. [2016] obtient des résultats comparables au système de Vial et al. [2018b], sachant que leur modèle de langue est entraîné sur un corpus privé contenant 100 milliards de mots provenant de nouvelles, ce qui rend la reproductibilité de leurs résultats très difficile. Ensuite, ils ont ajouté une étape de propagation d'étiquettes dans leur second système (LSTM + LP) afin d'augmenter automatiquement leurs données d'entraînement annotées en sens en recherchant des phrases similaires aux phrases annotées dans une grande quantité de textes non annotés et en portant les annotations de sens depuis les phrases annotées vers les phrases non annotées. Ceci leur a permis d'obtenir des meilleurs résultats. Toutefois, ils ont recueillis 1000 phrases issues du Web pour chaque lemme de leurs données non annotées, ce qui rend la repro-

ductibilité des résultats encore plus difficile.

Raganato et al. [2017], ayant un système de désambiguïsation lexicale très semblable au système de désambiguïsation lexicale de Vial et al. [2018b], ont obtenu des résultats moins élevés malgré leur modèle plus complexe. Ils ont utilisé de plus 2 couches de cellules LSTM bidirectionnelles de taille 2048 (1024 par direction). Dans leur deuxième système (BLTM + att. + LEX + POS), les auteurs ont ajouté une couche d'attention à leur réseau, et ils ont effectué de l'apprentissage multi-tâche, c'est à dire que leur réseau apprend à la fois à prédire une étiquette du mot ou du sens, ainsi que la partie du discours (POS) du mot, ainsi que son annotation sémantique dans WordNet (LEX), la tâche est rendue ainsi plus complexe.

5.3.4.2 Résultats de la désambiguïsation lexicale neuronale de l'arabe

Nous avons utilisé la ressource UFSAC-ara que nous avons créée, décrite dans la section 4.4, comme donnée d'apprentissage pour le système désambiguïsation lexicale arabe, et le corpus OntoNotes arabe décrit dans 4.2 comme corpus d'évaluation. Les données d'apprentissage sont constituées de 11 corpus parmi les 12 corpus de l'UFSAC traduits de l'anglais vers l'arabe, éliminant le corpus de tâche SemEval 2015 que nous avons utilisé comme corpus de développement (appelé Dev) pour éviter le surapprentissage (959 mots annotés). Nous avons utilisé toute la partie annotée de sens de OntoNotes en arabe (contenant 12524 mots annotés), en tant que données d'évaluation (appelée Test), afin de permettre à la communauté de recherche de comparer avec nos résultats en tant que baseline. Nous avons appliqué sur chaque corpus, le schéma de tokenisation *D3* de MADAMIRA (voir 1.3.2.1), qui consiste à tokeniser les proclitiques QUES, CONJ, les clitiques PART, ainsi que tous les articles et enclitiques. De plus, ce schéma normalise les caractères ALIF et YA après avoir supprimé les voyelles des caractères arabes. Le vocabulaire est donc réduit et les mots sont traités uniformément sans préfixes ni suffixes. L'ensemble de données utilisé pour l'apprentissage du système MT neuronal décrit dans le tableau 4.3 a également été segmenté avec le schéma de tokenisation *D3* pour être cohérent avec les ensembles Train, Dev et Test de notre système désambiguïsation lexicale.

Nous avons construit un modèle de word embeddings avec les données d'entraînement décrites dans le tableau 4.3, en utilisant Word2Vec [Mikolov et al., 2013a] avec 100 dimensions que nous avons utilisées comme entrée du système désambiguïsation lexicale neuronal. Le tableau 5.4 montre les résultats de notre système. Comme la plupart

des algorithmes de désambiguïsation lexicale supervisés, nous avons utilisé l'heuristique classique de choisir le premier sens de WordNet si notre système n'annote pas un terme. Cela est utile, par exemple, si les corpus annotés ne contiennent aucun exemple pour un mot à étiqueter.

Système	Précision	Rappel	Score F1
Dev (SE15 automatiquement traduit de l'anglais)			
Notre système	70.06%	69.55%	69.81%
Notre système + repli premier sens	70.28%	70.28%	70.28%
Baseline aléatoire	36.92%	36.92%	36.92%
Baseline du sens le plus fréquent	67.25%	67.25%	67.25%
Test (OntoNotes arabe)			
Notre système	**71.52%**	**71.32%**	**71.42%**
Notre système + repli premier sens	**71.52%**	**71.32%**	**71.42%**
Baseline aléatoire	40.26%	40.04%	40.15%
Baseline du sens le plus fréquent	58.11%	57.95%	58.03 %

TABLE 5.4: Performance de notre système de désambiguïsation lexicale arabe sur les ensembles Dev et Test. Sur l'ensemble de test, la stratégie de backoff est inutile car notre système annote déjà tous les mots possibles.

5.3.5 Analyse des erreurs

DL-Neuronal	Nb-Mots	S-WN	S-T	S-E-T	S-N-E-T	Exp-Mots
Mots mal annotés	3549	7.64	3.04	5.30	0.84	1.30M
Mots bien annotés	8908	6.26	2.39	9.25	1.36	2.22M
Total	12457	5.32	2.16	10.02	1.50	3.52M

TABLE 5.5: Analyse des erreurs de notre système de désambiguïsation neuronale arabe évalué sur le corpus OntoNotes arabe

Dans cette partie, nous analysons la sortie de notre système de désambiguïsation lexicale neuronale de l'arabe pour essayer de voir comment nous pourrions l'améliorer. À notre connaissance, c'est la première fois qu'une telle analyse est effectuée en désambiguïsation lexicale. Notre analyse est réalisée avec :

— **Nb-Mots** : le nombre total des mots annotés

— **S-WN** : la moyenne des sens des mots uniques annotés dans *Princeton WordNet*

— **S-T** : la moyenne des sens des mots annotés uniques dans le corpus d'entraînement
— **S-E-T** : la moyenne des sens des mots annotés uniques trouvés dans le corpus de Test et existant dans le corpus d'entraînement
— **S-N-E-T** : la moyenne des sens des mots annotés uniques trouvés dans le corpus de Test et non existant dans le corpus d'entraînement
— **Exp-Mots** : le nombre d'exemples des mots annotés existant dans le corpus d'entraînement

Le tableau 5.5 présente une analyse des erreurs produites par notre système de désambiguïsation neuronale pour les mots bien/mal annotés et pour la totalité du corpus. Il est important de noter que la ligne total pour les colonnes qui correspondent à des moyennes ne présente pas forcément une valeur située entre celles des mots bien/mal annotés car plusieurs mots sont parfois bien annotés et parfois mal annotés. Dans ce tableau, *Exp-Mots* montre que les exemples des mots bien annotés sont plus nombreux que les exemples des mots mal annotés dans le corpus d'apprentissage (1,30M Vs 2,22M), ce qui influence directement la qualité de désambiguïsation.

Nous pouvons également noter que la moyenne des sens *Princeton WordNet*, *S-WN*, est plus élevée que celle des sens existant dans nos données d'entraînement *S-T* (5,32 Vs 2,16). De plus, La colonne *S-E-T* montre que la moyenne des sens des mots bien annotés est supérieure à celle des sens des mots mal annotés dans le corpus d'entraînement. Cela signifie que le système ne peut prédire que 2,16 sens (appris) parmi 5,32 (trouvés dans *Princeton WordNet*) sur la totalité du corpus. C'est d'ailleurs le cas de 0,84 sens en moyenne (S-N-E-T) que nous ne pouvions trouver car il ne sont pas dans le corpus arabe (et ne correspondent pas au premier sens en anglais).

Parmi les pistes d'amélioration, nous pouvons ainsi chercher à augmenter le nombre d'exemples pour les sens peu ou pas annotés. Une méthode serait d'annoter des corpus non annotés grâce à ce système en ne conservant que le données dont nous sommes les plus sûr comme le font Yuan et al. [2016] ou Vial et al. [2018c].

5.4 Désambiguïsation lexicale pour la traduction automatique

Dans cette section, nous nous intéressons à évaluer l'impact de la désambiguïsation lexicale pour l'arabe sur la qualité des systèmes de traduction automatique. Cela nous permet également d'évaluer notre système de désambiguïsation lexicale de l'arabe par la traduction automatique avec une méthode d'évaluation *in vivo* (voir la section 2.2.4).

Notre objectif est d'exploiter notre meilleur système de désambiguïsation lexicale de l'arabe neuronal pour annoter automatiquement en sens la partie monolingue (arabe) de nos données dédiées pour la tâche de traduction automatique (Train/Dev/-Test). Les corpus produits seront par la suite utilisés pour créer des systèmes de traduction afin d'étudier l'effet de la désambiguïsation lexicale sur le comportement des systèmes de traduction au moment de l'apprentissage et du décodage.

Dans les sections suivantes, nous présentons les différentes étapes effectuées ainsi que les systèmes de traduction produits et leurs performances.

5.4.1 Corpus d'apprentissage

Dans le chapitre précédent nous avons construit un système de traduction automatique neuronal appris sur les 5 corpus TED, News-Commentary, Ummah, News et MultiUN soit 9 000K phrases pour l'apprentissage afin de traduire les 12 corpus UFSAC de l'anglais vers l'arabe. Toutefois, la désambiguïsation lexicale automatique du corpus MultiUN est très coûteuse en termes de temps et il n'était pas possible de tester le système complet dans le temps imparti. En effet, pour les 270 000 phrases des 4 premiers corpus, le système neuronal a mis 45 jours pour tout annoter [3] or le corpus MultiUN possède 9 millions de phrases, il demanderait ainsi environ 1500 jours pour être annoté soit plus de 4 années. Ainsi, nous allons créer un nouveau système de traduction automatique neuronal de base anglais-arabe ayant la même architecture que le système précédent mais axé sur les news en utilisant seulement les 4 corpus TED, News-Commentary, Ummah et News soit 277K phrases comme données d'entraînement. Il est donc important de noter que l'ancien et le nouveau système EN-AR ne sont pas comparables.

3. Les caractéristiques de la machine utilisée sont : système d'exploitation Linux, 16 processeurs, 80Go de RAM, 2 cartes GPU de 24Go, etc.

5.4.2 Désambiguïsation lexicale des données d'entraînement de traduction automatique

Afin d'obtenir les annotations en sens pour chacun des corpus de l'ensemble Train/-Dev/Test, nous avons appliqué le processus d'annotation automatique suivant :

1. Convertir les données textes brutes en format UFSAC
2. Appliquer la chaîne de pré-traitement décrite dans la section 4.3.1 sur l'ensemble des données arabes
3. Appliquer un traitement spécifique pour chaque langue pour obtenir le lemme en anglais et le POS :
   — Anglais : nous avons utilisé l'outil TreeTagger pour obtenir le lemme et le POS de chaque mot des corpus Train/Dev/Test.
   — Arabe : apprendre un modèle d'alignement sur l'ensemble des corpus parallèles Train/Dev/Test en utilisant l'outil Giza++. Notre objectif est de trouver le lemme de la traduction en anglais correspondante de chaque mot arabe.
4. Lancer la désambiguïsation lexicale de l'arabe sur chaque corpus (décrite dans la section 5.3.4.2)
5. Appliquer la chaîne de post-traitement (voir section 4.3.3)

La figure 5.2 montre un exemple d'une phrase arabe annotée en sens WordNet anglais à l'aide de notre système de désambiguïsation lexicale neuronal.

5.4.3 Apport de la désambiguïsation lexicale de l'arabe pour la traduction automatique

Étant donné que les données (de désambiguïsation lexicale) sont segmentées par le schéma *D3* de MADAMIRA (voir 1.3.2.1) et que les systèmes de traduction sont plus performants en utilisant le schéma de segmentation *ATB*, nous avons converti les corpus du schéma *D3* vers *ATB*.

Dans la section suivante, nous présentons les systèmes de traduction produits en exploitant les données arabes annotées en sens automatiquement.

```
<word surface_form="الا" lemma_ar="الا" pos="PRT" />
<word surface_form="ان" lemma_ar="ان" pos="p" />
<word surface_form="ه+" lemma_ar="ه" pos="p" />
<word surface_form="في" lemma_ar="في" pos="p" />
<word surface_form="ضوء" lemma_ar="ضوء" pos="n" />
<word surface_form="تطور" lemma_ar="تطور" pos="v" />
<word surface_form="حركة" id="d0038.s3.t6" lemma="business"   pos="n" wsd_test="business%1:04:01::" />
<word surface_form="ال+" lemma_ar="ال" pos="p" />
<word surface_form="عرض" lemma_ar="عرض" pos="n" />
<word surface_form="و+" lemma_ar="و" pos="p" />
<word surface_form="ال+" lemma_ar="ال" pos="p" />
<word surface_form="طلب" id="d0038.s3.t9" p="demand" wsd_test="demand%1:10:00::" />
<word surface_form="على" lemma_ar="على" pos="PRT" />
<word surface_form="ه+" lemma_ar="ه" pos="PROP" />
<word surface_form="," lemma_ar="," pos="PNX" />
<word surface_form="جري" id="d0038.s3.t13" lemma="happen"  pos="v" wsd_test="happen%2:30:00::" />
<word surface_form="تداول" lemma_ar="تداول" pos="n" />
<word surface_form="ه+" lemma_ar="ه" pos="p" />
<word surface_form="فعليا" lemma_ar="فعلي" pos="n" />
<word surface_form="في" lemma_ar="في" pos="p" />
<word surface_form="ال+" lemma_ar="ال" pos="p" />
<word surface_form="عمليات" id="d0038.s3.t18" lemma="procedure" pos="n" wsd_test="procedure%1:04:00::" />
<word surface_form="ال+" lemma_ar="ال" pos="p" />
<word surface_form="مصرفية" lemma_ar="مصرفي" pos="p" />
<word surface_form="عند" lemma_ar="عند" pos="p" />
```

FIGURE 5.2: Exemple de sortie de notre système de désambiguïsation lexicale

5.4.3.1 Système de traduction automatique neuronal factorisé

Comme l'arabe est une langue riche morphologiquement et qu'elle devient très ambigüe si le texte n'est pas voyellé. Notre hypothèse est donc que le système de traduction automatique doit être capable d'encoder et d'extraire les caractéristiques pertinentes à partir d'une suite de mots ainsi que leurs sens afin de les traduire correctement en anglais.

En utilisant la boite à outils OpenNMT [4] [Klein et al., 2017] (qui est un modèle «sequence-to-sequence»), nous avons développé des systèmes de traduction neuronaux de l'arabe vers l'anglais : un premier système de base (nommé baseline) mis en place en utilisant uniquement les textes source et cible bruts, et un deuxième système factorisé (nommé NMT-DL) qui utilise les textes source et cible dont les mots sont été annotés en sens. Les représentations vectorielles (*embeddings*) des mots et des sens ont été concaténées lors de l'apprentissage. Nos systèmes se composent de deux couches LSTM bidirectionnelles (voir la section 3.3.2.2.3) de taille 512, ainsi que des représentations vectorielles (*embeddings*) de taille 512 pour les mots et leurs sens. L'apprentissage est

4. https://github.com/OpenNMT/OpenNMT

effectué sur 25 epochs à l'aide de l'algorithme d'apprentissage *Adam* [Kingma and Ba, 2014b] et l'entropie-croisée comme fonction de perte (loss). La perplexité [Jelinek et al., 1977] (PPL) a été également calculée à chaque epoch.

La perplexité consiste à évaluer la modélisation linguistique du système et sa capacité à prédire les bonnes hypothèses. Elle est calculée en mesurant l'incertitude H d'un jeu de données S de n phrases (avec $S = s_1, s_2, ..., s_n$).

$$H = -\frac{1}{n}\sum_{i=1}^{n} \log_2 P(s_i) \tag{5.1}$$

Ainsi, la perplexité (PPL) est définie comme suit :

$$PPL = 2^H \tag{5.2}$$

Plus que la perplexité obtenue est faible, plus le modèle est pertinent.

5.4.3.2 Résultats et Analyse

Afin d'étudier l'impact de la désambiguïsation lexicale sur la tâche de traduction automatique, nous avons créé 4 systèmes de traduction pour chaque paire de langues arabe vers anglais (AR-EN) et anglais vers arabe (EN-AR) avec différentes configurations :

1. **Mot→Mot** : présente le système de base (baseline) dont l'entrée et la sortie du système sont des données brutes (mots).
2. **Mot+Sens→Mot** : le système de traduction est appris sur des données annotées (des mots et leurs sens). L'encodeur du système doit être capable de trouver les bonnes représentations des mots avec leurs sens pour produire en sortie une séquence de mots hypothèse.
3. **Mot→Mot+Sens** : le système de traduction est appris sur des données brutes (mots), et optimisé au moment de l'apprentissage pour produire des corpus annotés (des mots et leurs sens) au niveau du décodeur.
4. **Mot+Sens→Mot+Sens** : le système de traduction prend en entrée des données annotées (des mots et leurs sens), et fournit en sortie une séquence de mots annotés (des mots et leurs sens).

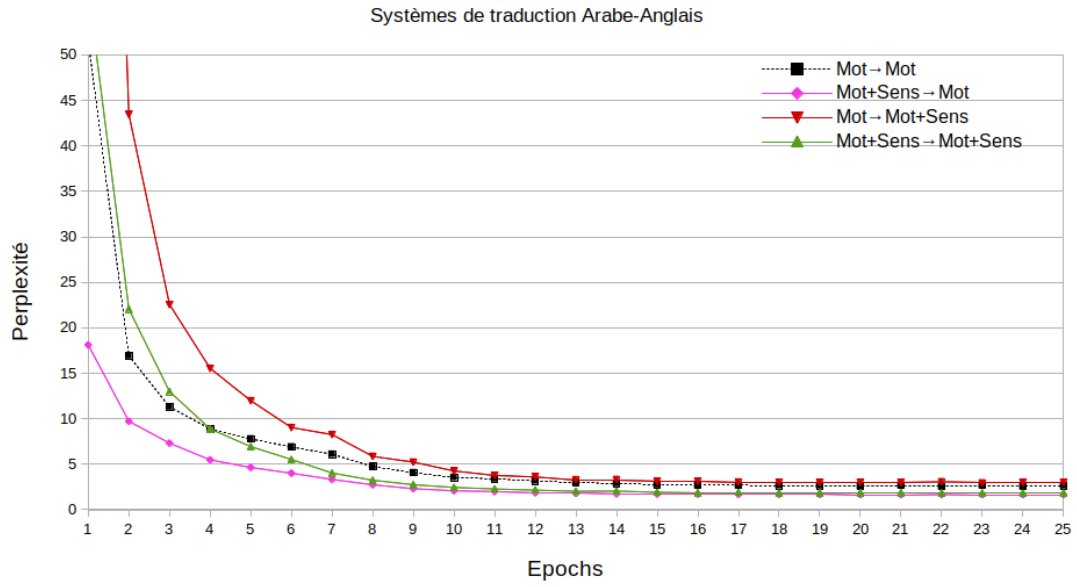

FIGURE 5.3: Impact de la désambiguïsation lexicale sur les systèmes de traduction **AR-EN** au moment de l'apprentissage en termes de perplexité

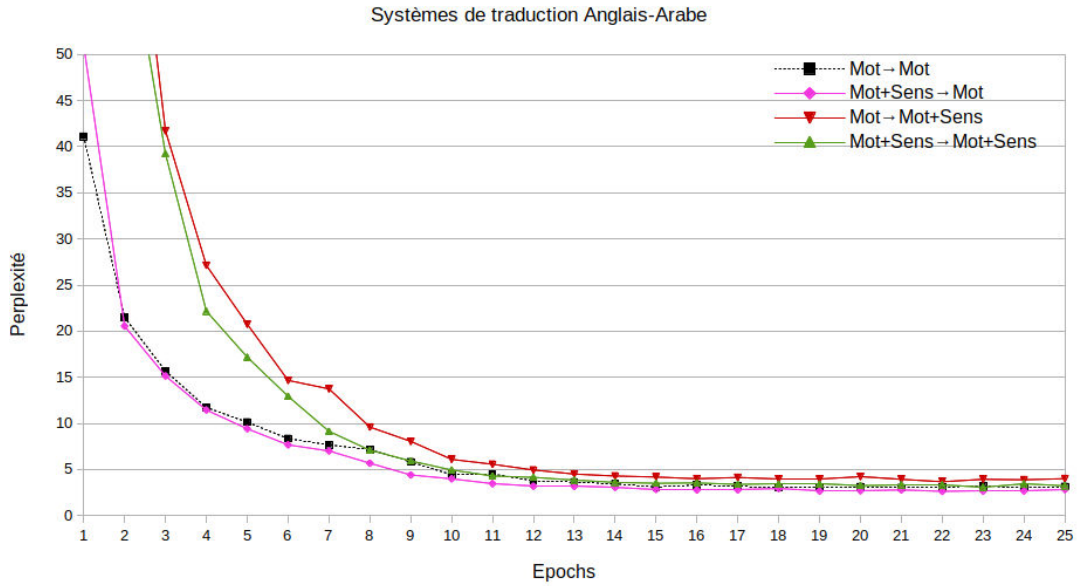

FIGURE 5.4: Impact de la désambiguïsation lexicale sur les systèmes de traduction **EN-AR** au moment de l'apprentissage en termes de perplexité

5.4.3.2.1 Effet de la désambiguïsation lexicale sur l'apprentissage

Comme illustré dans les deux figures 5.3 et 5.4, nous avons évalué l'effet de la désambiguïsation lexicale au moment de l'apprentissage des systèmes de traduction AR-EN et EN-AR par rapport au système de base **Mot**→**Mot** (ligne pointillée et colorée en noir) en termes de perplexité pour les 25 epochs. Tout d'abord, nous remarquons que la désambiguïsation lexicale influence la qualité des systèmes de traduction automatique lors de l'apprentissage, que les systèmes **Mot+Sens**→**Mot** (AR-EN et EN-AR) possèdent les meilleures courbes d'apprentissage (colorées en rose) et que les systèmes **Mot**→**Mot+Sens** (colorés en rouge) présentent les systèmes les moins performants. De plus, nous constatons que le comportement des systèmes AR-EN et EN-AR n'est pas le même sur les 10 premières epochs. Par exemple sur les systèmes AR-EN, l'exploitation des annotations en sens (des mots arabes) en entrée du système **Mot+Sens**→**Mot**s améliore la perplexité de 51,55 à 18,11 dès la première epoch par rapport au système de base (**Mot**→**Mot**). Sur le système EN-AR, la perplexité s'améliore légèrement à partir de l'epoch 2. Enfin, nous concluons que les annotations en sens sont plus pertinentes lorsqu'elles sont à l'entrée du système de traduction (au niveau de l'encodage) et que l'arabe (langue morphologiquement riche) est la langue source.

5.4.3.2.2 Performance des systèmes en termes de score BLEU

Les tableaux 5.6 et 5.7 résument les performances des systèmes de traduction évalués sur l'ensemble des données Test (Ummah+News) ainsi que sur ses différentes sources de sous corpus Ummah et News (800 lignes de chacun) en termes de score BLEU.

Tout d'abord, conformément aux observations sus-mentionnées à propos de l'apprentissage, nous constatons également que **Mot+Sens**→**Mot** (AR-EN et EN-AR) sont les plus performants et que les systèmes **Mot**→**Mot+Sens** (AR-EN et EN-AR) sont les moins performants en termes de score BLEU.

Dans le tableau 5.6, nous remarquons que les annotations en sens des mots arabes influencent énormément la qualité des systèmes AR-EN lorsqu'elles sont exploitées en entrée des systèmes. Nous obtenons respectivement une différence importante de +4.17% et +3,12% entre les systèmes factorisés (**Mot+Sens**→**Mot** et **Mot+Sens**→**Mot+Sens**) et le système de base (**Mot**→**Mot**) en termes de score BLEU. De plus, nous constatons une dégradation légère (-0,26%) des performances du système **Mot**→**Mot+Sens** par rapport au système de base (**Mot**→**Mot**).

Dans le tableau 5.7, nous constatons que les annotations de sens des mots anglais

Système NMT AR-EN		Corpus d'évaluation		
Entrée (AR)	**Sortie (EN)**	**Ummah**	**News**	**Test**
Système de base				
Mot	**Mot**	21.25	28.13	24.73
Systèmes factorisés				
Mot+Sens	**Mot**	**26.03**	**31.73**	**28.90**
Mot	**Mot+Sens**	22.07	26.84	24.47
Mot+Sens	**Mot+Sens**	25.06	30.60	27.85

TABLE 5.6: Évaluation des systèmes de traduction neuronale **AR-EN** sur le corpus Test (Ummah et News) en termes de score BLEU

Système NMT EN-AR		Corpus d'évaluation		
Entrée (EN)	**Sortie (AR)**	**Ummah**	**News**	**Test**
Système de base				
Mot	**Mot**	**24.58**	28.01	26.28
Systèmes factorisé				
Mot+Sens	**Mot**	24.34	**29.70**	**27.01**
Mot	**Mot+Sens**	21.74	24.79	23.25
Mot+Sens	**Mot+Sens**	22.66	28.59	25.60

TABLE 5.7: Évaluation des systèmes de traduction neuronale **EN-AR** sur le corpus Test (Ummah et News) en termes de score BLEU

améliorent légèrement les performances (+0,73%) du système **Mot+Sens→Mot** EN-AR en termes de score BLEU. Toutefois, nous remarquons une dégradation de performances lorsque la sortie des systèmes de traduction est annotée (**Mot+Sens** arabes). Cela signifie que les annotations en sens des mots arabes ne sont pas pertinentes lorsqu'elles sont exploitées en sortie des systèmes de traduction EN-AR.

Ainsi, après cette analyse de performances, nous concluons que les annotations de sens ont une forte influence sur la qualité des systèmes de traduction lorsque l'arabe est la langue source. De plus, ces annotations sont plus pertinentes lorsqu'elles sont à l'entrée du système de traduction. Aussi, nous concluons que l'utilisation des données annotées pour une langue morphologiquement riche (comme l'arabe) est pertinente pour la tâche de traduction automatique afin de réduire l'ambiguïté des données.

5.4.3.2.3 Exploitation de données plus riches

En outre, nous avons exploité les informations morphologiques **Lemme** et Partie de discours (**POS**) afin d'améliorer nos systèmes de traduction automatique factorisés. Ces informations ont été obtenues à l'aide de l'outil MADAMIRA pour les corpus arabes et l'outil TreeTagger pour l'anglais. Ensuite, nous avons enrichi nos ensembles de données Train/Dev/Test avec les informations Sens, **Lemme** et **POS**. Suite à l'hypothèse tirée précédemment, nous intégrons ces informations uniquement dans la langue source.

Système NMT		**Corpus d'évaluation**		
Entrée	**Sortie**	**Ummah**	**News**	**Test**
Système de base				
Mot	**Mot**	21.25	28.13	24.73
Systèmes factorisés				
Mot+Sens	**Mot**	26.03	31.73	28.90
Mot+Lemme+POS	**Mot**	25.40	31.42	28.43
Mot+Sens+Lemme+POS	**Mot**	**26.49**	**32.94**	**29.74**

TABLE 5.8: Évaluation de l'effet des informations morphologiques (Lemme+POS) sur les systèmes de traduction neuronale **AR-EN** sur le corpus Test (Ummah et News) en termes de score BLEU

Système NMT		**Corpus d'évaluation**		
Entrée	**Sortie**	**Ummah**	**News**	**Test**
Système de base				
Mot	**Mot**	24.58	28.01	26.28
Systèmes factorisés				
Mot+Sens	**Mot**	24.34	29.70	27.01
Mot+Lemme+POS	**Mot**	**24.72**	30.30	27.48
Mot+Sens+Lemme+POS	**Mot**	24.46	**31.15**	**27.78**

TABLE 5.9: Évaluation de l'effet des informations morphologiques (Lemme+POS) sur les meilleurs systèmes de traduction neuronale **EN-AR** sur le corpus Test (Ummah et News) en termes de score BLEU

Les tableaux 5.8 et 5.9 présentent les performances obtenues en intégrant les infor-

mations morphologiques dans des systèmes de traduction ayant une entrée factorisée. En comparant les performances obtenues sur les systèmes de base et les systèmes factorisés, nous constatons que les informations Lemme et **POS** améliorent significativement la qualité de traduction pour les systèmes AR-EN et EN-AR.

Sur les systèmes AR-EN, en exploitant les informations morphologiques avec les mots (**Mot**+**Lemme**+**POS**), nous constatons dans le tableau 5.8 que ces informations ont la même importance que les informations de désambiguïsation lexicale **Sens**. Le système **Mot**+**Sens**→**Mot** est légèrement meilleur que le système **Mot**+**Lemme**+**POS**→**Mot** avec une différence de +0,47% en termes de score BLEU sur l'ensemble du corpus de Test.

En utilisant les informations de désambiguïsation lexicale ainsi que les informations morphologiques, nous améliorons significativement notre système de +5,01% par rapport au système de base en termes de BLEU sur notre corpus de Test. Sur les systèmes EN-AR, les performances obtenues dans le tableau 5.9 montrent que les informations morphologiques sont légèrement importantes par rapport à l'information **Sens**. La différence obtenue est à -0,47% en termes de score BLEU obtenu entre les systèmes **Mot**+**Sens**→**Mot** et **Mot**+**Lemme**+**POS**→**Mot**. En exploitant les deux types d'informations (désambiguïsation lexicale et morphologie), nous obtenons respectivement une différence de +1,50% par rapport au système de base en termes de score BLEU sur le corpus de Test.

Nous concluons que les informations de désambiguïsation lexicale et morphologiques sont plus pertinentes lorsque l'arabe est une langue source.

Les figures 5.5 et 5.6 présentente des exemples de traduction automatique obtenues avec nos différents systèmes (systèmes de base et systèmes factorisés).

5.5 Conclusion

Dans ce chapitre, nous avons exploité nos 12 corpus arabes traduits automatiquement annotés en sens avec *Princeton WordNet* pour le processus d'apprentissage de deux systèmes de désambiguïsation lexicale de l'arabe, avec une architecture à base de séparateurs à vaste marge et une architecture neuronale. Les résultats sur l'arabe prouvent l'efficacité de notre méthode et sont très encourageants pour l'avenir de la désambiguïsation lexicale sur les langues ayant une petite quantité de données annotées en sens. Nous avons également proposé une grille d'analyse des résultats de

	Données
Source: Mot	و+ طالبت الاحزاب السياسية المعارضة امس ب+ استقالة يلماظ اثر نشر هذا الشريط .
Source: Mot+Sens	و+|و+ طالبت|طالبت الاحزاب|الاحزاب ::party%1:14:01 السياسية|السياسية ::political%3:00:00 المعارضة|المعارضة ::opposition%1:14:01 امس|امس ::yesterday%1:28:01 ب+|ب+ استقالة|استقالة ::resignation%1:10:00 يلماظ|يلماظ اثر|اثر نشر|نشر هذا|هذا الشريط|الشريط ::airing%1:10:00 .|.
Source: Mot+Sens+Lemme+POS	و+|و+|وَ+ p طالبت|طالبت|طالَب v الاحزاب|الاحزاب ::party%1:14:01|حِزْب n السياسية|السياسية ::political%3:00:00|سِياسِيّ n المعارضة|المعارضة ::opposition%1:14:01|مُعارِض n امس|امس ::yesterday%1:28:01|أَمْس n ب+|ب+|بِ+ p استقالة|استقالة ::resignation%1:10:00|اِسْتِقالَة n يلماظ|يلماظ|يِلْماظ p اثر|اثر|أَثَر n نشر|نشر|نَشْر n هذا|هذا|هٰذا n الشريط|الشريط ::airing%1:10:00|شَرِيط px .|.|. n
Référence	opposition political parties demanded yilmaz 's resignation yesterday , following the airing of this tape .

Système	Bleu	Hypothèse
		Système de base
Mot→Mot	**58.42**	opposition parties called on opposition political parties yesterday , in the aftermath of the publication of this tape .
		Systèmes factorisés
Mot+Sens→Mot	**66.82**	opposition political parties asked yesterday by yilmaz 's resignation after that tape was published .
Mot+Sens +Lemme+POS →Mot	**84.87**	opposition political parties demanded yesterday the yilmaz 's resignation following the publication of this tape .
Mot→Mot+Sens	**57.99**	the|the opposition|opposition%1:14:01:: parties|party%1:14:01:: called|called yesterday|yesterday%1:28:01:: the|the opposition|opposition%1:14:01:: 's|'s resignation|resignation%1:10:00:: to|to the|the resignation|resignation%1:10:00:: of|of yilmaz|yilmaz 's|'s resignation|resignation%1:10:00:: .|.
Mot+Sens→Mot+Sens	**66.59**	the|the opposition|opposition%1:14:01:: political|political%3:00:00:: parties|party%1:14:01:: called|called yesterday|yesterday%1:28:01:: at|at the|the resignation|resignation%1:10:00:: of|of yilmaz|yilmaz ,|, that|that the|the tape|tape%1:06:00:: was|was published|published .|.

FIGURE 5.5: Exemple d'amélioration de la traduction automatique (AR-EN)

désambiguïsation lexicale supervisée permettant d'aider à comprendre comment pallier les erreurs réalisées par le système. Par ailleurs, nous avons exploité les prédictions de notre système de désambiguïsation lexicale neuronal de l'arabe pour l'entraînement d'un nouveau système de traduction automatique. Les performances obtenues montrent que la désambiguïsation lexicale améliore la qualité de traduction automatique respectivement de +4,17% et +0,73% sur les systèmes AR-EN et EN-AR en termes de score BLEU. En exploitant les informations morphologiques **Lemme** et **POS** avec l'information Sens, nous améliorons nos systèmes de traduction avec une différence de +5,01% et +1,50% sur les systèmes AR-EN et EN-AR par rapport aux systèmes de base en termes de score BLEU. Cela confirme notre intuition que disposer des informations relatives au sens (**Sens**) et à la morphologie (**Lemme** et **POS**) sont des informations très intéressantes lorsque la langue source est ambigüe et à morphologie très riche par rapport à la langue cible.

	Données
Source: Mot	exporters object to white list registration conditions
Source: Mot+Sens	exporters|**exporter%1:18:00::** object|**object%2:32:00::** to|**to** white|**white%3:00:01::** list|**list%1:10:00::** registration|**registration%1:04:00::** conditions|**condition%1:26:00::**
Source: Mot+Sens+Lemme+POS	exporters|**exporter%1:18:00::**|**exporter**|**n** object|**object%2:32:00::**|**object**|**v** to|**to**|**to**|**n** white|**white%3:00:01::**|**white**|**n** list|**list%1:10:00::**|**list**|**n** registration|**registration%1:04:00::**|**registration**|**n** conditions|**condition%1:26:00::**|**condition**|**n**
Référence	المصدرون يعترضون علي شروط القيد بالقائمة البيضاء

Système	Bleu	Hypothèse
		Système de base
Mot→Mot	**28.29**	طلبات المصدرين لشروط التسجيل الابيض
		Systèmes factorisés
Mot+Sens→Mot	**57.22**	المصدرون يعترض علي لائحة اللائحة البيضاء لالقيد
Mot+Sens +Lemme+POS →Mot	**72.57**	المصدرون يعترض علي شروط تسجيل القائمة البيضاء
Mot→Mot+Sens	**38.56**	المصدرون|**exporter%1:18:00**|**exporter%1:18:00::** ل+|**لشروط**|::condition%1:10:02 السوق|**السوق** البيضاء|**البيضاء**
Mot+Sens→Mot+Sens	**35.9**	المصدرون|**exporter%1:18:00::** تحتج|**تحتج** على|**علي** شروط|**::condition%1:10:02** عشوائية|**عشوائية** ب+|**بعشوائية**|عشوائية

FIGURE 5.6: Exemple d'amélioration de la traduction automatique (EN-AR)

6

Conclusion et perspectives

6.1 Conclusion

Le travail présenté dans ce manuscrit porte sur la désambiguïsation lexicale de la langue arabe que l'on considère comme une langue peu dotée pour cette tâche.

D'une part, notre objectif a été de mettre en œuvre une méthode générique (qui puisse être appliquée à n'importe quelle langue) afin de construire un système de désambiguïsation lexicale supervisé pour une langue L en utilisant un bon système de traduction automatique d'une langue riche en corpus annotés, telle que l'anglais, vers une langue moins bien dotée L, telle que l'arabe, et portant des annotations de la langue riche vers la langue moins bien dotée L.

D'autre part, nous nous sommes intéressée à l'évaluation de notre système de désambiguïsation lexicale créé en suivant deux approches : une évaluation *in vitro* où notre système de désambiguïsation est évalué sur un corpus annoté de référence que nous avons enrichi semi-automatiquement, et une évaluation *in vivo* où notre système de désambiguïsation est évalué sur sa contribution à la performance de systèmes de traduction automatique. À notre connaissance, ce type d'évaluation n'a pas encore jamais été appliqué dans les campagnes d'évaluation.

La désambiguïsation lexicale est une tâche qui a pour but d'associer le sens le plus approprié à chaque mot d'un texte parmi les sens proposés par un inventaire de sens prédéfini. Dans une approche supervisée, pour entraîner un système de désambiguïsation lexicale, il est nécessaire de disposer d'une grande quantité de données annotées en sens. Il est possible de faire appel à un ou plusieurs experts linguistes afin d'annoter manuellement, avec les sens appropriés, les données d'entraînement. Une telle tâche est très coûteuse en temps et en main d'œuvre. Pour évaluer un système de désambiguïsation lexicale, il faut disposer un corpus annoté de référence. Afin que différents systèmes et différentes approches puissent être évaluée comparativement, il est nécessaire qu'un corpus de référence soit disponible pour la communauté et gratuit. La disponibilité de données d'entraînement et d'un corpus d'évaluation de référence sont des questions bien plus pertinentes pour des langues qui ne possèdent pratiquement aucune ressource textuelle annotées en sens libre de droit. La langue arabe fait partie de cette catégorie de langues comportant très peu de ressources disponibles pour la tâche qui nous intéresse ici.

Ce travail de thèse repose sur plusieurs axes. Dans un premier temps, nous avons décrit les difficultés liées aux langues moins dotées que l'anglais pour la construction de système de désambiguïsation lexicale. La difficulté est principalement due au manque de corpus annotés en sens. Nous fournissons ainsi à la communauté 12 corpus [1] traduits automatiquement en arabe et annotés en sens avec *Princeton WordNet*. Ces corpus peuvent être utilisés lors de l'étape d'apprentissage d'un système de désambiguïsation lexicale. Nous l'avons montré en les exploitant avec une architecture à base de machine à vecteurs de support et une architecture neuronale.

Dans un second temps, pour une évaluation *in vitro* de la désambiguïsation lexicale de l'arabe, nous avons mis à jour la partie arabe du corpus OntoNotes Release 5.0, et créé notre propre système de traduction anglais-arabe, pour créer un système de désambiguïsation de l'arabe grâce au portage d'annotations depuis l'anglais. Nous avons ainsi montré la faisabilité et la généricité de l'approche. Ensuite, nous avons évalué les résultats d'une méthode de désambiguïsation automatique supervisée.

Les résultats sur l'arabe prouvent l'efficacité de notre méthode et sont très encourageants pour l'avenir de la désambiguïsation lexicale des langues ayant une petite quantité de données annotées en sens. Par conséquent, il est maintenant possible de produire de bons systèmes de désambiguïsation lexicale pour n'importe quelle langue pourvu que l'on dispose d'un bon système de traduction automatique de l'anglais vers celle-ci. L'ensemble des 12 corpus arabes, ainsi que les scripts utilisés pour les produire sont mis à la disposition de la communauté. Nous avons également proposé une grille d'analyse des résultats de désambiguïsation lexicale supervisée permettant d'aider à comprendre comment pallier les erreurs réalisées par le système.

Par ailleurs, nous avons effectué une évaluation *in vivo* de notre système de désambiguïsation lexicale de l'arabe en le comparant à un système de désambiguïsation lexical de anglais, développé au sein de notre équipe, pour l'annotation de grands corpus bruts alignés (données d'entraînement de notre système de TA anglais-arabe). Ensuite, nous avons utilisé ces grands corpus anglais et arabe comme données d'entraînement de systèmes de traduction neuronaux factorisés (données d'entraînement annotées en sens) arabe-anglais (AR-EN) et anglais-arabe (EN-AR).

Nous avons évalué ces systèmes factorisés en termes de score BLEU. Les résultats confirment l'hypothèse d'amélioration de la qualité des résultats de traduction automatique sur des entrées désambiguïsées lexicalement. Nous avons amélioré les perfor-

1. https://github.com/getalp/UFSAC-ara/

mances des systèmes de traduction automatique neuronaux de +4,17% pour le système AR-EN et de +0,73% pour le système EN-AR en termes de score BLEU.

De plus, en ajoutant les informations morphologiques **Lemme** et **POS** à l'information Sens, nous avons réussi à améliorer encore plus la qualité de la traduction automatique avec une différence respective de +5,01% et de +1,50% en termes de score BLEU pour les systèmes AR-EN et EN-AR en les comparant avec les systèmes de base (systèmes neuronaux à base de mots). Ce qui semble montrer que les informations additionnelles Sens, **Lemme** et **POS** sont un apport important lorsque la langue source est une langue ambiguë à morphologie très riche comparée à la langue cible.

En outre, il est important de mentionner que nous pouvons obtenir des grands corpus arabes annotés en sens issus de *Princeton WordNet* via notre système de traduction neuronal factorisé **Mot+Sens**→**Mot+Sens** sans grande perte d'informations.

6.2 Perspectives

À partir des contributions présentées dans ce manuscrit, diverses perspectives peuvent êtres envisagées.

Tout d'abord, nous prévoyons d'augmenter la taille des données parallèles anglais-arabe annotées en sens. Les dernières améliorations à la fois conceptuelles (réduction du vocabulaire des sens) et implémentationnelles (tout est dorénavant uniquement en Python sans échange avec du Java) réalisées sur le système de désambiguïsation lexicale ont fortement accéléré les traitements. L'annotation des 9 Millions de phrases manquantes devient ainsi envisageable en temps raisonnable (de l'ordre du mois contre 4 ans auparavant).

Nous envisageons également d'exploiter les corpus alignés et leurs informations translingues pour les réinjecter dans des systèmes de traduction statistique factorisés, afin de comparer les hypothèses de sortie des systèmes de TA statistiques et neuronaux.

Une autre perspective importante serait d'exploiter les sorties des systèmes de traduction neuronaux factorisés anglais-arabe et arabe-anglais pour la création de nouveaux systèmes de désambiguïsation lexicale supervisés pour l'arabe et l'anglais afin d'améliorer la tâche de désambiguïsation lexicale. L'idée sous-jacente serait d'essayer de voir jusqu'à quel point il est possible d'obtenir un cercle vertueux où la traduction automatique s'améliore grâce à la désambiguïsation lexicale qui en retour s'améliore grâce à la traduction automatique.

Par ailleurs, nous proposons d'utiliser de nouveaux facteurs tels que les relations de synonymie afin d'améliorer les tâches de désambiguïsation lexicale et de traduction automatique.

Puisque la sortie de notre système de TA anglais-arabe factorisé est annotée avec des sens issus du *Princeton WordNet*, nous envisageons de faire intervenir des annotateurs humains pour annoter rapidement ces corpus arabes en validant ou en remplaçant les annotations hypothèses produites afin de confirmer que notre système de traduction factorisé peut jouer simultanément le rôle d'un système de traduction automatique et d'un système de désambiguïsation lexicale de l'arabe.

Ces travaux pourraient également permettre une étude contrastive des attributs pertinents pour la désambiguïsation d'une langue (arabe, français, etc.), étude qu'il n'était pas possible de mener avant l'existence de corpus pour autant de langues.

Enfin, ces travaux commencent à être exploités dans l'équipe dans le cadre d'autres applications que la traduction automatique. Nos méthodes ont été adaptées au français pour réaliser un système de génération de pictogrammes à partir d'énoncés oraux ou écrits dans lesquels le choix des pictogrammes se fait en levant les ambiguïtés lexicales. Ce type d'application est utile pour permettre d'établir une communication alternative par exemple pour des personnes en situation de polyhandicap ou pour des personnes maîtrisant pas ou peu la langue.

Bibliographie personnelle

M. Hadj Salah, L. Vial, H. Blanchon, M. Zrigui, B. Lecouteux, and D. Schwab. La désambiguïsation lexicale d'une langue moins bien dotée, l'exemple de l'arabe. In 25e conférence sur le Traitement Automatique des Langues Naturelles, 2018.

M. Hadj Salah, H. Blanchon, M. Zrigui, and D. Schwab. Un corpus en arabe annoté manuellement avec des sens wordnet. In 25e conférence sur le Traitement Automatique des Langues Naturelles, 2018.

M. Hadj Salah, H. Blanchon, M. Zrigui, and D. Schwab. Amélioration de la traduction automatique d'un corpus annoté. In JEP-TALN-RECITAL 2016, 2016.M. Hadj Salah, H. Blanchon, M. Zrigui, and D. Schwab. Un corpus en arabe annoté ma-nuellement avec des sens wordnet. In 25e conférence sur le Traitement Automatique des Langues Naturelles, 2018

L. Besacier, B. Lecouteux, N.-Q. Luong, K. Hour, and M. Hadj Salah. Word confidence estimation for speech translation. In International Workshop on Spoken Language Translation, 2014.

A

Annexes

```
<sentence>
  <word surface_form="The" pos="DT" />
  <word surface_form="Fulton_County_Grand_Jury" lemma="group" pos="NNP" wn16_key="group%1:03:00::"
wn30_key="group%1:03:00::" />
  <word surface_form="said" lemma="say" pos="VB" wn16_key="say%2:32:00::" wn30_key="say%2:32:00::" /
>
  <word surface_form="Friday" lemma="friday" pos="NN" wn16_key="friday%1:28:00::" wn30_key="friday%
1:28:00::" />
  <word surface_form="an" pos="DT" />
  <word surface_form="investigation" lemma="investigation" pos="NN" wn16_key="investigation%
1:09:00::" wn30_key="investigation%1:09:00::" />
  <word surface_form="of" pos="IN" />
  <word surface_form="Atlanta" lemma="atlanta" pos="NN" wn16_key="atlanta%1:15:00::"
wn30_key="atlanta%1:15:00::" />
  <word surface_form="&amp;apos;s" pos="POS" />
  <word surface_form="recent" lemma="recent" pos="JJ" wn16_key="recent%5:00:00:past:00"
wn30_key="recent%3:00:00:past:00" />
  <word surface_form="primary_election" lemma="primary_election" pos="NN" wn16_key="primary_election
%1:04:00::" wn30_key="primary_election%1:04:00::" />
  <word surface_form="produced" lemma="produce" pos="VB" wn16_key="produce%2:39:01::"
wn30_key="produce%2:39:01::" />
  <word surface_form="``" pos="``" />
  <word surface_form="no" pos="DT" />
  <word surface_form="evidence" lemma="evidence" pos="NN" wn16_key="evidence%1:09:00::"
wn30_key="evidence%1:09:00::" />
  <word surface_form="&amp;apos;&amp;apos;" pos="&amp;apos;&amp;apos;" />
  <word surface_form="that" pos="IN" />
  <word surface_form="any" pos="DT" />
  <word surface_form="irregularities" lemma="irregularity" pos="NN" wn16_key="irregularity%
1:04:00::" wn30_key="irregularity%1:04:00::" />
  <word surface_form="took_place" lemma="take_place" pos="VB" wn16_key="take_place%2:30:00::"
wn30_key="take_place%2:30:00::" />
  <word surface_form="." pos="." />
</sentence>
```

FIGURE 1.1: Exemple d'une phrase annotée du corpus UFSAC sans aucun traitement

```
<sentence>
  <word surface_form="the" id_TA="0" pos="DT" />
  <word surface_form="fulton" id_TA="1" id_TOK="1" lemma="group" pos="NNP" wn16_key="group%
1:03:00::" wn30_key="group%1:03:00::" />
  <word surface_form="county" id_TA="2" id_TOK="1" lemma="group" pos="NNP" wn16_key="group%
1:03:00::" wn30_key="group%1:03:00::" />
  <word surface_form="grand" id_TA="3" id_TOK="1" lemma="group" pos="NNP" wn16_key="group%
1:03:00::" wn30_key="group%1:03:00::" />
  <word surface_form="jury" id_TA="4" id_TOK="1" lemma="group" pos="NNP" wn16_key="group%1:03:00::"
wn30_key="group%1:03:00::" />
  <word surface_form="said" id_TA="5" lemma="say" pos="VB" wn16_key="say%2:32:00::" wn30_key="say%
2:32:00::" />
  <word surface_form="friday" id_TA="6" lemma="friday" pos="NN" wn16_key="friday%1:28:00::"
wn30_key="friday%1:28:00::" />
  <word surface_form="an" id_TA="7" pos="DT" />
  <word surface_form="investigation" id_TA="8" lemma="investigation" pos="NN"
wn16_key="investigation%1:09:00::" wn30_key="investigation%1:09:00::" />
  <word surface_form="of" id_TA="9" pos="IN" />
  <word surface_form="atlanta" id_TA="10" lemma="atlanta" pos="NN" wn16_key="atlanta%1:15:00::"
wn30_key="atlanta%1:15:00::" />
  <word surface_form="&apos;s" id_TA="11" pos="POS" />
  <word surface_form="recent" id_TA="12" lemma="recent" pos="JJ" wn16_key="recent%5:00:00:past:00"
wn30_key="recent%3:00:00:past:00" />
  <word surface_form="primary" id_TA="13" id_TOK="2" lemma="primary_election" pos="NN"
wn16_key="primary_election%1:04:00::" wn30_key="primary_election%1:04:00::" />
  <word surface_form="election" id_TA="14" id_TOK="2" lemma="primary_election" pos="NN"
wn16_key="primary_election%1:04:00::" wn30_key="primary_election%1:04:00::" />
  <word surface_form="produced" id_TA="15" lemma="produce" pos="VB" wn16_key="produce%2:39:01::"
wn30_key="produce%2:39:01::" />
  <word surface_form="``" id_TA="16" pos="``" />
  <word surface_form="no" id_TA="17" pos="DT" />
  <word surface_form="evidence" id_TA="18" lemma="evidence" pos="NN" wn16_key="evidence%1:09:00::"
wn30_key="evidence%1:09:00::" />
  <word surface_form="&apos;&apos;" id_TA="19" pos="&apos;&apos;" />
  <word surface_form="that" id_TA="20" pos="IN" />
  <word surface_form="any" id_TA="21" pos="DT" />
  <word surface_form="irregularities" id_TA="22" lemma="irregularity" pos="NN"
wn16_key="irregularity%1:04:00::" wn30_key="irregularity%1:04:00::" />
  <word surface_form="took" id_TA="23" id_TOK="3" lemma="take_place" pos="VB" wn16_key="take_place%
2:30:00::" wn30_key="take_place%2:30:00::" />
  <word surface_form="place" id_TA="24" id_TOK="3" lemma="take_place" pos="VB" wn16_key="take_place%
2:30:00::" wn30_key="take_place%2:30:00::" />
  <word surface_form="." id_TA="25" pos="." />
</sentence>
```

FIGURE 1.2: Exemple d'une phrase annotée du corpus UFSAC en anglais pré-traité

```
<sentence>
  <word surface_form="و+" id_TA="0" pos="DT" />
  <word surface_form="فولتون" id_TA="1" id_TOK="1" lemma="group" pos="NNP" wn16_key="group%
1:03:00::" wn30_key="group%1:03:00::" />
  <word surface_form="مقاطعة" id_TA="2" id_TOK="1" lemma="group" pos="NNP" wn16_key="group%
1:03:00::" wn30_key="group%1:03:00::" />
  <word surface_form="المحلفين" id_TA="3" id_TOK="1" lemma="group" pos="NNP" wn16_key="group%
1:03:00::" wn30_key="group%1:03:00::" />
  <word surface_form="في" id_TA="3" id_TOK="1" lemma="group" pos="NNP" wn16_key="group%1:03:00::"
wn30_key="group%1:03:00::" />
  <word surface_form="هيئة" id_TA="4" id_TOK="1" lemma="group" pos="NNP" wn16_key="group%1:03:00::"
wn30_key="group%1:03:00::" />
  <word surface_form="المحلفين" id_TA="4" id_TOK="1" lemma="group" pos="NNP" wn16_key="group%
1:03:00::" wn30_key="group%1:03:00::" />
  <word surface_form="قال" id_TA="5" lemma="say" pos="VB" wn16_key="say%2:32:00::" wn30_key="say%
2:32:00::" />
  <word surface_form="ان" id_TA="5" lemma="say" pos="VB" wn16_key="say%2:32:00::" wn30_key="say%
2:32:00::" />
  <word surface_form="فولتون" id_TA="6" lemma="friday" pos="NN" wn16_key="friday%1:28:00::"
wn30_key="friday%1:28:00::" />
  <word surface_form="في" id_TA="7" pos="DT" />
  <word surface_form="اجري" id_TA="8" lemma="investigation" pos="NN" wn16_key="investigation%
1:09:00::" wn30_key="investigation%1:09:00::" />
  <word surface_form="اجرت" id_TA="9" pos="IN" />
  <word surface_form="+ها" id_TA="11" pos="POS" />
  <word surface_form="في" id_TA="12" lemma="recent" pos="JJ" wn16_key="recent%5:00:00:past:00"
wn30_key="recent%3:00:00:past:00" />
  <word surface_form="الاونة" id_TA="12" lemma="recent" pos="JJ" wn16_key="recent%5:00:00:past:00"
wn30_key="recent%3:00:00:past:00" />
  <word surface_form="الاخيرة" id_TA="12" lemma="recent" pos="JJ" wn16_key="recent%5:00:00:past:00"
wn30_key="recent%3:00:00:past:00" />
  <word surface_form="الاولية" id_TA="13" id_TOK="2" lemma="primary_election" pos="NN"
wn16_key="primary_election%1:04:00::" wn30_key="primary_election%1:04:00::" />
  <word surface_form="الانتخابات" id_TA="14" id_TOK="2" lemma="primary_election" pos="NN"
wn16_key="primary_election%1:04:00::" wn30_key="primary_election%1:04:00::" />
  <word surface_form="التي" id_TA="15" lemma="produce" pos="VB" wn16_key="produce%2:39:01::"
wn30_key="produce%2:39:01::" />
  <word surface_form="لم" id_TA="17" pos="DT" />
  <word surface_form="ادلة" id_TA="18" lemma="evidence" pos="NN" wn16_key="evidence%1:09:00::"
wn30_key="evidence%1:09:00::" />
  <word surface_form="تثبت" id_TA="18" lemma="evidence" pos="NN" wn16_key="evidence%1:09:00::"
wn30_key="evidence%1:09:00::" />
  <word surface_form="عن" id_TA="19" pos="&amp;apos;&amp;apos;" />
  <word surface_form="تسفر" id_TA="20" pos="IN" />
  <word surface_form="عن" id_TA="20" pos="IN" />
  <word surface_form="اي" id_TA="21" pos="DT" />
  <word surface_form="حدوث" id_TA="21" pos="DT" />
  <word surface_form="اي" id_TA="21" pos="DT" />
  <word surface_form="مخالفات" id_TA="22" lemma="irregularity" pos="NN" wn16_key="irregularity%
1:04:00::" wn30_key="irregularity%1:04:00::" />
  <word surface_form="." id_TA="25" pos="." />
</sentence>
```

FIGURE 1.3: Exemple d'une phrase annotée du corpus UFSAC après **le processus de traduction (EN-AR)**, **le processus de portage d'annotations**

```
<sentence>
  <word surface_form="و+" id_TA="0" pos="DT" />
  <word surface_form="فولتون هيئة المحلفين في مقاطعة" id_TA="1"  lemma="group" pos="NNP"
wn16_key="group%1:03:00::" wn30_key="group%1:03:00::" />
  <word surface_form="قال" id_TA="5" lemma="say" pos="VB" wn16_key="say%2:32:00::" wn30_key="say%
2:32:00::" />
  <word surface_form="ان" pos="RP" />
  <word surface_form="فولتون" id_TA="6" lemma="friday" pos="NN" wn16_key="friday%1:28:00::"
wn30_key="friday%1:28:00::" />
  <word surface_form="في" id_TA="7" pos="DT" />
  <word surface_form="اجري" id_TA="8" lemma="investigation" pos="NN" wn16_key="investigation%
1:09:00::" wn30_key="investigation%1:09:00::" />
  <word surface_form="أجرت" id_TA="9" pos="IN" />
  <word surface_form="+ها" id_TA="11" pos="POS" />
  <word surface_form="في" pos="RP" />
  <word surface_form="الاونة الاخيرة" id_TA="12" lemma="recent" pos="JJ" wn16_key="recent%
5:00:00:past:00" wn30_key="recent%3:00:00:past:00" />
  <word surface_form="الانتخابات الاولية" id_TA="13"  lemma="primary_election" pos="NN"
wn16_key="primary_election%1:04:00::" wn30_key="primary_election%1:04:00::" />
  <word surface_form="التي" id_TA="15" lemma="produce" pos="VB" wn16_key="produce%2:39:01::"
wn30_key="produce%2:39:01::" />
  <word surface_form="لم" id_TA="17" pos="DT" />
  <word surface_form="أدلة تثبت" id_TA="18" lemma="evidence" pos="NN" wn16_key="evidence%1:09:00::"
wn30_key="evidence%1:09:00::" />
  <word surface_form="عن" id_TA="19" pos="&amp;apos;&amp;apos;" />
  <word surface_form="تسفر عن" id_TA="20" pos="IN" />
  <word surface_form="اي حدوث أي" id_TA="21" pos="DT" />
  <word surface_form="مخالفات" id_TA="22" lemma="irregularity" pos="NN" wn16_key="irregularity%
1:04:00::" wn30_key="irregularity%1:04:00::" />
  <word surface_form="." id_TA="25" pos="." />
</sentence>
```

FIGURE 1.4: Exemple d'une phrase annotée du corpus UFSAC après **le processus de traduction (EN-AR)**, **le processus de portage d'annotations** et **le processus de post-traitement**

 # Sentence pair (1) source length 6 target length 4 alignment score : 4.13487e-07
$ 10,000 gold ?
NULL ({ }) 4 ({ 3 }) ب+ ({ }) عشرة ({ 2 }) الاف ({ 1 }) دولار ? ({ }) الذهب

 # Sentence pair (2) source length 79 target length 43 alignment score : 7.21819e-88
with soaring deficits , and a rudderless fiscal policy , one does wonder whether a populist administration might recklessly turn to the printing press . and if you are really worried
about that , gold might indeed be the most reliable hedge .
NULL ({ 22 39 }) الاميركي ({ }) الدولار ({ }) لـ+ ({ }) الكامل ({ }) التساول ({ }) ب+ ({ }) الطبع ({ }) , ({ }) الانهيار ({ }) هذا ({ }) علي ({ }) الاجابات ({ }) بين ({ }) من ({ }) . (}) ف+ ({ }) في ({ }) ظل ({ 1 }) مستويات ({ }) العجز ({ 3 }) التي ({ }) ارتفعت ({ }) الي ({ }) عنان ({ 2 }) السماء ({ }) , ({ 4 }) و+ ({ 5 }) السياسة ({ 9 }) المالية ({ 8 }) التي ({ } { }) فقدت ({ }) الدفة ({ 7 }) التي ({ }) توجه ({ 6 }) +ها ({ }) , ({ 10 }) لا ({ 12 }) يملك ({ }) المرء ({ 11 }) الا ({ }) ان ({ }) يتساءل ({ 13 }) ما ({ }) اذا ({ 14 }) كانت ({ }) احدي ({ }) الادارات ({ 17 }) الشعوبية ({ 16 }) قد ({ 18 }) تلجا ({ 20 }) ب+ ({ }) تهور ({ 19 }) الي ({ 21 }) طباعة ({ 23 24 }) اوراق ({ }) النقد ({ }) . ({ 25 }) و+ ({ 26 }) ان ({ }) ك ({ 15 }) (نتم ({ 27 28 }) تشعرون ({ 29 30 }) ب+ ({ }) القلق ({ 31 }) ازاء ({ 32 }) هذا ({ 33 }) الاحتمال ({ }) ف+ ({ 34 }) ان ({ }) الذهب ({ 35 }) قد ({ 36 }) يكون ({ 38 }) حقا ({ 37 }) وسيلة { }) 43 ({ }) . ({ }) الثقة ({ }) ب+ ({ }) الاكثر ({ 42 }) جدارة ({ 40 }) التحوط ({ 41 })

 # Sentence pair (3) source length 65 target length 67 alignment score : 1.57404e-126
even so , the fact that very high inflation is possible does not make it probable , so one should be cautious in arguing that higher gold prices are being driven by inflation expectations . some have argued instead that gold 's long upward march has been partly driven by the development of new financial instruments that make it easier to trade and speculate in gold .
NULL ({ 4 51 53 }) تجعل ({ 14 }) لا ({ 12 13 }) الارتفاع ({ 7 }) البالغ ({ 9 }) التضخم ({ 6 }) حدوث ({ 5 }) امكانية ({ }) مجرد ({ 3 }) ف+ ({ }) , ({ 2 }) ذلك ({ 1 }) رغم ({ }) و+ (}) ه ({ 15 }) امرا ({ 10 }) محتملا ({ 16 11 }) , ({ 17 }) لذا ({ 18 }) يتعين ({ 21 }) علي ({ }) المرء ({ 19 }) ان ({ }) يتوخي ({ 20 }) الحذر ({ 22 }) حين ({ 23 }) يزعم ({ 24 }) ان+ ({ } { 25 }) م ({ 39 }) بدلا ({ 36 }) البعض ({ 38 }) زعم ({ 37 }) لقد ({ }) و+ ({ 35 }) . ({ 33 }) التضخم ({ 34 }) توقعات ({ }) الي ({ }) يرجع ({ 27 }) الذهب ({ 28 }) اسعار ({ 26 }) ارتفاع ({ }) ن ({ }) ان ({ 40 }) مسيرة ({ 42 45 }) الذهب ({ 41 }) الصاعدة ({ 44 }) الطولية ({ 43 }) كانت ({ 47 46 }) مدفوعة ({ 29 30 31 32 49 50 }) جزئيا ({ 48 }) ب+ ({ }) تطوير ({ 52 }) الادوات ({ 56 }) المالية ({ 55 }) الجديدة ({ 54 }) التي ({ 57 }) تعمل ({ 58 59 }) علي ({ 61 }) تيسير ({ 60 }) المتاجرة ({ 62 }) و+ ({ 63 }) المضاربة ({ 64 }) في ({ 65 }) الذهب ({ 66 }) . ({ 67 })

FIGURE 1.5: Exemple d'alignement en utilisant Giza++

Segmentation	Phrase
Brut	أعلن وزير الاتصالات والمعلومات المصري الدكتور أحمد نظيف أنه سيتم الانتهاء من صياغة تعريفة اتصالات جديدة قبل نهاية العام الجاري مؤك داً أن التعريفة الجديدة تستهدف إعادة التوازن وتبسيط تعريفة النداء الآلي داخل المحافظة الواحدة وبين المحافظات المختلفة .
D3	اعلن وزير ال+ اتصالات و+ ال+ معلومات ال+ مصري ال+ دكتور احمد نظيف ان +ه س+ يتم ال+ انتهاء من صياغة تعريفة اتصالات جديدة قبل نهاية ال+ عام ال+ جاري مؤكدا ان ال+ تعريفة ال+ جديدة تستهدف اعادة ال+ توازن و+ تبسيط تعريفة ال+ نداء ال+ الي داخل ال+ محافظة ال+ واحدة و+ بين ال+ محافظات ال+ مختلفة .
ATB	اعلن وزير الاتصالات و+ المعلومات المصري الدكتور احمد نظيف ان +ه س+ يتم الانتهاء من صياغة تعريفة اتصالات جديدة قبل نهاية العام الجاري مؤكدا ان التعريفة الجديدة تستهدف اعادة التوازنو+ تبسيط تعريفة النداء الالي داخل المحافظة الواحدة و+ بين المحافظات المختلفة .

TABLE 1.1: Exemple de segmentation d'une phrase en D3 et ATB avec MADAMIRA

Source	in the summer of 2005 , a picture that people have long been looking forward to started emerging with frequency in various major hong kong media .
Hypothèse de TA	و+ في صيف عام 2005 ، كانت الصورة التي يتطلع الناس الي +ها منذ فترة طويلة ل+ الظهور ب+ وتيرة في مختلف وسائط الاعلام الرئيسية في هونغ كونغ .
Fast-Align	0-1 1-0 2-2 3-2 4-3 4-4 5 -5 7-7 8-8 9-10 10-12 10-13 11-14 11-15 12-13 13-9 14-9 15-16 16-17 18-18 19-19 20-20 21-21 22-24 23-26 24-26 24-27 25-22 25-23 26-28

TABLE 1.2: Exemple d'alignement en utilisant Fast-Align

	Données
Source: Mot	romanian ambassador in cairo : mubarak 's visit aims at bolstering economic cooperation & coordinating political stands
Source: Mot+Sens	romanian|romanian%3:01:00:: ambassador|ambassador%1:18:00:: in|in cairo|cairo%1:15:00:: :|: mubarak|mubarak%1:18:00:: 's|'s visit|visit%1:04:02:: aims|aim%2:33:00:: at|at bolstering|bolstering economic|economic%3:01:01:: cooperation|cooperation%1:04:01:: &|& coordinating|coordinating political|political%3:00:00:: stands|stand%1:06:00::
Source: Mot+Sens + Lemme+POS	romanian|romanian%3:01:00::|romanian|n ambassador|ambassador%1:18:00::|ambassador|n in|in|in|n cairo|cairo%1:15:00::|cairo|n :|:|:|px mubarak|mubarak%1:18:00::|mubarak|n 's|'s|'s|n visit|visit%1:04:02::|visit|n aims|aim%2:33:00::|aim|v at|at|at|n bolstering|bolstering|bolstering|n economic|economic%3:01:01::|economic|n cooperation|cooperation%1:04:01::|cooperation|n &|&|&|n coordinating|coordinating|coordinating|n political|political%3:00:00::|political|n stands|stand%1:06:00::|stand|n
Référence	سفير رومانيا بالقاهرة : زيارة الرئيس مبارك تستهدف تدعيم التعاون الاقتصادي وتنسيق المواقف السياسية

Système	Bleu	Hypothèse
		Système de base
Mot→Mot	**60.72**	السفير الروماني بالقاهرة : زيارة مبارك لالطيران يهدف الى دعم التعاون الاقتصادي لالتنسيق بين مصر ورومانيا
		Systèmes factorisés
Mot+Sens→Mot	**69.18**	السفير الروماني في القاهرة : زيارة مبارك تهدف الي دعم التعاون الاقتصادي وتنسيق التنسيق السياسي
Mot+Sens +Lemme+POS →Mot	**69.79**	السفير الروماني بالقاهرة : زيارة مبارك تهدف الي تعزيز التعاون الاقتصادي وتنسيق مواقف سياسية
Mot→Mot+Sens	**62.07**	السفير|ambassador%1:18:00:: الروماني|romanian%3:01:00:: في|في القاهرة|cairo%1:15:00:: :|: مبارك|visit%1:04:01:: |mubarak%1:18:00:: زيارة|visit%1:14:00:: ل+|لتعزيز|تعزيز التعاون|cooperation%1:04:01:: الاقتصادي|economic%3:00:00:efficient:00 الروسي|الروسي بين|بين التنسيق|التنسيق السياسي|political%3:01:01::
Mot+Sens→Mot+Sens	**73.4**	السفير|ambassador%1:18:00:: الروماني|romanian%3:01:00:: في|في القاهرة|cairo%1:15:00:: :|: زيارة|visit%1:14:00:: مبارك|mubarak%1:18:00:: يهدف|يهدف الي|الي تعزيز|تعزيز التعاون|cooperation%1:04:01:: الاقتصادي|economic%3:00:00:efficient:00 و+|وتنسيق|تنسيق المواقف|stand%1:09:00:: السياسية|political%3:00:00::

FIGURE 1.6: Exemple d'amélioration de la traduction automatique (EN-AR)

Source: Mot	profitability of american companies ' branches operating in egypt higher than its rates in us
Source: Mot+Sens	profitability|**profitability%1:07:00::** of|**of** american|**american%3:01:01::** companies|**company%1:14:01::** '|' branches|**branch%1:14:00::** operating|**operating** in|**in** egypt|**egypt%1:15:00::** higher|**higher%3:00:02:high:02** than|**than** its|**its** rates|**rate%1:21:00::** in|**in** us|**us**
Source: Mot+Sens + Lemme+POS	profitability|**profitability%1:07:00::**|**profitability**|**n** of|**of**|**of**|**n** american|**american%3:01:01::**|**american**|**n** companies|**company%1:14:01::**|**company**|**n** '|'|'|**n** branches|**branch%1:14:00::**|**branch**|**n** operating|**operating**|**operating**|**n** in|**in**|**in**|**n** egypt|**egypt%1:15:00::**|**egypt**|**n** higher|**higher%3:00:02:high:02**|**higher**|**n** than|**than**|**than**|**n** its|**its**|**it**|**n** rates|**rate%1:21:00::**|**rate**|**n** in|**in**|**in**|**n** us|**us**|**us**|**n**
Référence	ربحية فروع الشركات الامريكية العاملة بمصر اعلي من معدلاتها داخل الولايات المتحدة

Système	Bleu	Hypothèse
		Système de base
Mot→Mot	**46.61**	ربحية الشركات الامريكية في مصر اكثر من معدلاتها في امريكا
		Systèmes factorisés
Mot+Sens→Mot	**52.42**	ربحية الشركات الامريكية العاملة في مصر اعلى معدلاتها في مصر
Mot+Sens +Lemme+POS →Mot	**77.44**	ربحية فروع الشركات الامريكية العاملة في مصر اعلي من اسعارها في الولايات المتحدة
Mot→Mot+Sens	**68.3**	ربحية|**ربحية** الشركات|**::company%1:14:01** الامريكية|**::american%3:01:01** التي|**التي** تعمل|**تعمل** في|**في** مصر|**مصر** ب+|**بنسبة** نسبة اكبر|**اكبر** من|**من** معدلات|**::rate%1:21:00ها**|+ها في|**في** الولايات|**الولايات** المتحدة|**المتحدة**
Mot+Sens→Mot+Sens	**54.24**	ربحية|**::profitability%1:07:00** الشركات|**::company%1:14:01** الامريكية|**::american%3:01:01** العاملة|**العاملة** في|**في** مصر|**::egypt%1:15:00** اعلي|**اعلي** من|**من** اسعار|**::rate%1:21:00** الفائدة|**::interest%1:21:00** في|**في** امريكا|**امريكا**

FIGURE 1.7: Exemple d'amélioration de la traduction automatique (EN-AR)

	Données
Source: Mot	she added that china had not so far participated in any multinational forces .
Source: Mot+Sens	she|she added|added that|that china|china%1:15:00:: had|had not|not%4:02:00:: so|so%4:02:04:: far|far%4:02:02:: participated|participated in|in any|any multinational|multinational%3:00:00:international:00 forces|force%1:14:01:: .|.
Source: Mot+Sens + Lemme+POS	she|she|she|n added|added|added|n that|that|that|n china|china%1:15:00::|china|n had|had|have|n not|not%4:02:00::|not|n so|so%4:02:04::|so|n far|far%4:02:02::|far|n participated|participated|participated|n in|in|in|n any|any|any|n multinational|multinational%3:00:00:international:00|multinational|n forces|force%1:14:01::|force|n .|.|.|px
Référence	واضافت ان الصين لم تشارك حتي الان في اية قوات متعدده الجنسيات .

Système	Bleu	Hypothèse
		Système de base
Mot→Mot	**75.12**	واضافت ان الصين لم حتي الان في اي قوي متعدده الجنسيات .
		Systèmes factorisés
Mot+Sens→Mot	**81.7**	واضافت ان الصين لم تعد حتي الان في اي قوات متعدده الجنسيات .
Mot+Sens +Lemme+POS →Mot	**90.34**	واضافت ان الصين لم تشارك حتي الان في اي قوات متعدده الجنسيات .
Mot→Mot+Sens	**90.34**	و+|واضافت|اضافت ان|ان الصين|لم|لم تشارك|تشارك حتي|حتي الان|الان ::china%1:15:00 في|في اي|اي قوات|قوات متعدده|متعدده ::force%1:14:01 الجنسيات|multinational%3:00:00:international:00 .|.
Mot+Sens→Mot+Sens	**90.34**	و+|واضافت|اضافت ان|ان الصين|لم|لم تشارك|تشارك حتي|حتي الان|الان ::china%1:15:00 في|في اي|اي قوات|قوات متعدده|متعدده ::force%1:14:01 الجنسيات|multinational%3:00:00:international:00 .|.

FIGURE 1.8: Exemple d'amélioration de la traduction automatique (EN-AR)

	Données
Source: Mot	تعد محادثات ابوجا خطوه تمهيدية ل+ المحادثات الرئيسية المقرر عقد +ها في نفس المدينة الاسبوع القادم و+ التي يتوقع ان يحضر +ها الرئيس جوزيف كابيلا .
Source: Mot+Sens	تعد|تعد محادثات|محادثات ابوجا|preparatory%3:00:00:preceding:00 خطوه|خطوة تمهيدية|تمهيدية ل+|ل+ المحادثات|المحادثات::step%1:04:02:: الرئيسية|الرئيسية المقرر|المقرر عقد|عقد +ها|+ها في|في main%3:00:00:important:00 نفس|نفس المدينة|المدينة::city%1:15:01 الاسبوع|الاسبوع::week%1:28:02 القادم|القادم::same%3:00:02 و+|و+ التي|التي يتوقع|يتوقع ان|ان يحضر|يحضر +ها|+ها الرئيس|الرئيس::president%1:18:04 جوزيف|جوزيف كابيلا|كابيلا .|. ::joseph%1:18:01
Source: Mot+Sens + Lemme+POS	تعد|تعد|عاد|v محادثات|محادثات|مُحادَثَة|n ابوجا|أبُوجا|preparatory%3:00:00:preceding:00|p خطوه|خَطْوَة|n::step%1:04:02:: تمهيدية|تمهيدية|تَمْهِيديّ|n ل+|ل+|ل+|p المحادثات|المحادثات|مُحادَثة|n الرئيسية|الرئيسية|رَئِيسِيّ|n المقرر|المقرر|مُقَرّر|n عقد|عقد|عَقْد|n +ها|+ها|n في|في|فِي|p نفس|نفس|نَفْس|n::same%3:00:02 المدينة|المدينة|مَدِينَة|n::city%1:15:01 الاسبوع|الاسبوع|أسْبُوع|n::week%1:28:02 القادم|القادم|قادِم|n و+|و+|وَ+|p التي|التي|الّذِي|n يتوقع|يتوقع|تَوَقَّع|v ان|ان|أنْ|p يحضر|يحضر|أحْضَر|v +ها|+ها|+ها|n الرئيس|الرئيس|رَئِيس|n جوزيف|جوزيف|جُوزِيف|p::joseph%1:18:01 ::president%1:18:04 كابيلا|كابيلا|كابِيلا|px .|.|.|p
Référence	the abuja discussions are considered a preparatory step towards the main discussions to be held in the same city next week , which are expected to be attended by president joseph kabila .

Système	Bleu	Hypothèse
		Système de base
Mot→Mot	**28.31**	rives 's talks are a preliminary step for the main talks to be held in the same city next week .
		Systèmes factorisés
Mot+Sens→Mot	**61.07**	preparations are a preliminary step to main posts due to be held in the same city next week , which is expected by president joseph kabila .
Mot+Sens +Lemme+POS →Mot	**74.05**	mofaz 's discussions are considered a preliminary step to be held in the same city next week , which is expected to be attended by president joseph kabila .
Mot→Mot+Sens	**69.7**	the|the talks|talk%1:10:00:: are|are unanimous|unanimous%3:00:00:accordant:00 a|a preliminary|preliminary%3:00:00:exploratory:00 step|step%1:04:02:: for|for the|the main|main%3:00:00:important:00 talks|talk%1:10:00:: to|to be|be%2:42:06:: held|held in|in the|the same|same%3:00:02:: city|city%1:15:00:: next|next%3:00:00:succeeding:00 week|week%1:28:02:: ,|, which|which is|is expected|expected to|to be|be%2:42:06:: attended|attended by|by president|president%1:18:04:: joseph|joseph%1:18:02:: kabila|kabila .|.
Mot+Sens→Mot+Sens	**65.17**	a|a preparatory|preparatory step|step%1:04:02:: forward|forward%4:02:00:: to|to a|a main|main%3:00:00:important:00 step|step%1:04:02:: will|will be|be%2:42:06:: held|held in|in the|the same|same%3:00:02:: city|city%1:15:00:: next|next week|week%1:28:02:: ,|, which|which is|is expected|expected to|to be|be%2:42:06:: attended|attended by|by president|president%1:18:04:: joseph|joseph%1:18:02:: kabila|kabila .|.

FIGURE 1.9: Exemple d'amélioration de la traduction automatique (AR-EN)

	Données
Source: Mot	البربر اعلنوا ب+ مقاطعة +هم ان +هم غير راضين عن مبادرات الرئيس عبد العزيز بوتفليقة ل+ حل مشاكل +هم .
Source: Mot+Sens	البربر|berber%1:18:00:: اعلنوا|اعلنوا ب+|ب+ مقاطعة|مقاطعة +هم|+هم ان|ان +هم|+هم غير|غير راضين|راضين عن|عن مبادرات|initiative%1:07:00:: الرئيس|president%1:18:04:: عبد|عبد العزيز|العزيز بوتفليقة|بوتفليقة ل+|ل+ حل|solve%2:31:00:: مشاكل|problem%1:26:00:: +هم|+هم .|.
Source: Mot+Sens + Lemme+POS	البربر|berber%1:18:00:: |بَرْبَرِيّ|p اعلنوا|اعلنوا|أَعْلَنَ|v ب+|ب+|بِ+|p مقاطعة|مقاطعة|مُقاطَعَة|n +هم|+هم|+هُم|n ان|ان|أَنَّ|p +هم|+هم|+هُم|n غير|غير|غَيْر|p راضين|راضين|راضٍ|n عن|عن|عَن|p مبادرات|initiative%1:07:00::|مُبادَرَة|n الرئيس|president%1:18:04::|رَئِيس|n عبد|عبد|عَبْد|p العزيز|العزيز|عَزِيز|p بوتفليقة|بوتفليقة|بُوتَفلِيقَة|p ل+|ل+|لِ+|p حل|solve%2:31:00::|حَل|n مشاكل|problem%1:26:00::|مُشْكِلَة|n +هم|+هم|+هُم|px .|.|. |n
Référence	the berbers announced with their boycott that they are dissatisfied with the initiatives of president abd al azeez butafleeka to solve their problems .

Système	Bleu	Hypothèse
		Système de base
Mot→Mot	**4.92**	berber berber announces in their boycott
		Systèmes factorisés
Mot+Sens→Mot	**66.14**	the berber declared that they are dissatisfied with the initiatives of president bouteflika to solve their problems .
Mot+Sens +Lemme+POS →Mot	**56.43**	the berber declared that they are satisfied with president bouteflika 's initiatives to solve their problems .
Mot→Mot+Sens	**68.45**	the|the berber|berber%1:18:00:: condemned|condemned in|in their|their boycott|boycott that|that they|they are|are not|not%4:02:00:: satisfied|satisfied with|with president|president%1:18:04:: bouteflika|bouteflika 's|'s initiatives|initiative%1:07:00:: to|to solve|solve%2:31:00:: their|their problems|problem%1:26:00:: .|.
Mot+Sens→Mot+Sens	**52.55**	the|the berber|berber%1:18:00:: declared|declared that|that they|they were|were satisfied|satisfied by|by president|president%1:18:04:: bouteflika|bouteflika 's|'s initiatives|initiative%1:07:00:: to|to solve|solve%2:31:00:: their|their problems|problem%1:26:00:: .|.

FIGURE 1.10: Exemple d'amélioration de la traduction automatique (AR-EN)

Bibliographie

L. Abouenour, K. Bouzoubaa, and P. Rosso. On the evaluation and improvement of Arabic wordnet coverage and usability. *Language Resources and Evaluation*, 47(3) : 891–917, 2013.

S. Abualhaija and K.-H. Zimmermann. D-bees : A novel method inspired by bee colony optimization for solving word sense disambiguation. *Swarm and Evolutionary Computation*, pages –, 2016.

H. M. Al-Serhan, R. Al Shalabi, and G. Kannan. New approach for extracting arabic roots. 2003.

A. Almahairi, K. Cho, N. Habash, and A. Courville. First result on arabic neural machine translation. *arXiv preprint arXiv :1606.02680*, 2016.

F. Alotaiby, S. Foda, and I. Alkharashi. Clitics in arabic language : a statistical study. In *Proceedings of the 24th Pacific Asia Conference on Language, Information and Computation*, 2010.

A. Amr. Syntax-based statistical machine translation models @ONLINE, 2008.

M. Apidianaki. Translation-oriented word sense induction based on parallel corpora. In *Language Resources and Evaluation (LREC)*, 2008.

D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv :1409.0473*, 2015.

L. Bentivogli, P. Forner, and E. Pianta. Evaluating cross-language annotation transfer in the multisemcor corpus. In *COLING 2004 : Proceedings of the 20th International Conference on Computational Linguistics*, 2004.

W. Black, S. Elkateb, H. Rodriguez, M. Alkhalifa, P. Vossen, A. Pease, and C. Fellbaum. Introducing the arabic wordnet project. In *Proceedings of the third international WordNet conference*, pages 295–300. Citeseer, 2006.

W. J. Black and S. ElKateb. A prototype english-arabic dictionary based on wordnet. In *Proceedings of 2nd Global WordNet Conference, GWC2004, Czech Republic*, pages 67–74, 2004.

F. Bond and R. Foster. Linking and extending an open multilingual wordnet. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, volume 1, pages 1352–1362, 2013.

H. Bouamor, H. Alshikhabobakr, B. Mohit, and K. Oflazer. A human judgement corpus and a metric for arabic mt evaluation. In *EMNLP*, pages 207–213, 2014.

P. Brown, S. D. Pietra, V. D. Pietra, and R. Mercer. Word-sense disambiguation using statistical methods. In *Proceedings of the 29th annual meeting on Association for Computational Linguistics*, pages 264–270. Association for Computational Linguistics, 1991.

L. Burnard. *The British National Corpus*. 1998.

A. Bérard. Neural machine translation architectures and applications @ONLINE, 2018.

J. F. Cai, W. S. Lee, and Y. W. Teh. Nus-ml : Improving word sense disambiguation using topic features. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, SemEval '07, pages 249–252, Stroudsburg, PA, USA, 2007. Association for Computational Linguistics. URL http://dl.acm.org.gate6.inist.fr/citation.cfm?id=1621474.1621527.

M. Carpuat and D. Wu. Improving statistical machine translation using word sense disambiguation. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, 2007.

Y. S. Chan, H. T. Ng, and Z. Zhong. Nus-pt : exploiting parallel texts for word sense disambiguation in the english all-words tasks. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 253–256. Association for Computational Linguistics, 2007.

F. Christiane, B. William, E. Sabri, M. Antonia, P. Adam, R. Horacio, and V. Piek. Constructing arabic wordnet in parallel with an ontology, 2005.

J. Chung, K. Cho, and Y. Bengio. A character-level decoder without explicit segmentation for neural machine translation. *arXiv preprint arXiv :1603.06147*, 2016.

J. Cowie, J. Guthrie, and L. Guthrie. Lexical disambiguation using simulated annealing. In *COLING 1992*, volume 1, pages 359–365, Nantes, France, août 1992.

F. Debili, H. Achour, and E. Souissi. La langue arabe et l'ordinateur de l'étiquetage gramatical à la voyellation automatique. *Correspondances : bulletin de l'IRMC*, (71) :10–26, 2002.

S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6), 1990. URL http://citeseer.nj.nec.com/deerwester90indexing.html.

M. Diab. Second generation amira tools for arabic processing : Fast and robust tokenization, pos tagging, and base phrase chunking. In *2nd International Conference on Arabic Language Resources and Tools*, 2009.

M. T. Diab. An unsupervised approach for bootstrapping arabic sense tagging. In *Proceedings of the Workshop on Computational Approaches to Arabic Script-based Languages*, pages 43–50. Association for Computational Linguistics, 2004.

J. Dichy, A. Braham, S. Ghazali, and M. Hassoun. La base de connaissances linguistiques diinar. 1 (dictionnaire informatisé de l'arabe, version 1). In *Proceedings of the International Symposium on The Processing of Arabic, Tunis (La Manouba University)*, pages 18–20, 2002.

C. Dyer, V. Chahuneau, and N. A. Smith. A simple, fast, and effective reparameterization of ibm model 2. Association for Computational Linguistics, 2013.

M. S. Eid, A. B. Al-Said, N. M. Wanas, M. A. Rashwan, and N. H. Hegazy. Comparative study of rocchio classifier applied to supervised wsd using arabic lexical samples. In *Proceedings of the tenth conference of language engeneering (SEOLEC'2010), Cairo, Egypt*, 2010.

A. Eisele and Y. Chen. Multiun : A multilingual corpus from united nation documents. In *LREC*, 2010.

S. Elkateb, W. Black, P. Vossen, D. Farwell, H. Rodríguez, A. Pease, and M. Alkhalifa. Arabic wordnet and the challenges of arabic. In *Proceedings of Arabic NLP/MT Conference, London, UK*. Citeseer, 2006.

W. B. H. R. M. A. P. V. A. P. Elkateb, Sabri and C. Fellbaum. Building a wordnet for Arabic. In *In Proceedings of The fifth international conference on Language Resources and Evaluation (LREC 2006)*, 2006.

J. L. Elman. Finding structure in time. *Cognitive science*, 14(2) :179–211, 1990.

S. Elmougy, H. Taher, and H. Noaman. Naïve bayes classifier for arabic word sense disambiguation. In *proceeding of the 6th International Conference on Informatics and Systems*, pages 16–21. Citeseer, 2008.

M. Federico, N. Bertoldi, and M. Cettolo. Irstlm : an open source toolkit for handling large scale language models. In *Ninth Annual Conference of the International Speech Communication Association*, 2008.

W. N. Francis and H. Kučera. A standard corpus of present-day edited american english, for use with digital computers (brown). Technical report, Brown University, Providence, Rhode Island, 1964.

A. Gelbukh, G. Sidorov, and S. Y. Han. Evolutionary approach to natural language wsd through global coherence optimization. *WSEAS Transactions on Communications*, 2 (1) :11–19, 2003.

S. A. Ghaffar and M. W. Fakhr. English to arabic statistical machine translation system improvements using preprocessing and arabic morphology analysis. *Recent Researches in Mathematical Methods in Electrical Engineering and Computer Science*, pages 50–54, 2011.

A. Graves, A.-r. Mohamed, and G. Hinton. Speech recognition with deep recurrent neural networks. In *Acoustics, speech and signal processing (icassp), 2013 ieee international conference on*, pages 6645–6649. IEEE, 2013.

N. Habash and O. Rambow. Arabic tokenization, part-of-speech tagging and morphological disambiguation in one fell swoop. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 573–580. Association for Computational Linguistics, 2005.

N. Habash and F. Sadat. Arabic preprocessing schemes for statistical machine translation. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume : Short Papers*, pages 49–52. Association for Computational Linguistics, 2006.

N. Habash, O. Rambow, and R. Roth. Mada+ tokan : A toolkit for arabic tokenization, diacritization, morphological disambiguation, pos tagging, stemming and lemmatization. In *Proceedings of the 2nd international conference on Arabic language resources and tools (MEDAR), Cairo, Egypt*, pages 102–109, 2009.

N. Habash, R. Roth, O. Rambow, R. Eskander, and N. Tomeh. Morphological analysis and disambiguation for dialectal arabic. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, pages 426–432, 2013.

B. Habert, C. Fabre, and F. Issac. *DE L'ECRIT AU NUMERIQUE. Constituer, normaliser et exploiter les corpus électroniques*. Number ISBN : 2-225-82953-5. ELSEVIER MASSON, 1998.

L. S. Hadla, T. M. Hailat, and M. N. Al-Kabi. Evaluating arabic to english machine translation. *Editorial Preface*, 5(11), 2014.

Z. S. Harris. Distributional structure. *Word*, 10(2-3) :146–162, 1954.

S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9 (8) :1735–1780, 1997.

I. Iacobacci, M. T. Pilehvar, and R. Navigli. Embeddings for word sense disambiguation : An evaluation study. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, pages 897–907, Berlin, Germany, August 2016. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/P16-1085.

N. Ide and C. Macleod. The american national corpus : A standardized resource of american english. In *Proceedings of Corpus Linguistics 2001*, volume 3, 2001.

F. Jelinek, R. L. Mercer, L. R. Bahl, and J. K. Baker. Perplexity - measure of the difficulty of speech recognition tasks. *The Journal of the Acoustical Society of America*, 62(S1) :S63–S63, 1977.

C. R. Johnson, C. J. Fillmore, M. R. Petruck, C. F. Baker, M. Ellsworth, J. Ruppenhofer, and E. J. Wood. Framenet : Theory and practice, 2002.

K. S. Jones. Towards better nlp system evaluation. In *Proceedings of the workshop on Human Language Technology*, pages 102–107. Association for Computational Linguistics, 1994.

E. Kang. Long short-term memory (lstm) : Concept @ONLINE, 2017.

S. Khoja. Apt : Arabic part-of-speech tagger. In *Proceedings of the Student Workshop at NAACL*, pages 20–25, 2001.

S. Khoja. *APT : An automatic arabic part-of-speech tagger*. PhD thesis, Lancaster University, 2003.

S. Kim, M. Jeong, J. Lee, and G. G. Lee. A cross-lingual annotation projection approach for relation detection. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 564–571. Association for Computational Linguistics, 2010.

D. P. Kingma and J. Ba. Adam : A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014a. URL http://arxiv.org/abs/1412.6980.

D. P. Kingma and J. Ba. Adam : A method for stochastic optimization. *arXiv preprint arXiv :1412.6980*, 2014b.

K. Kipper, H. T. Dang, M. Palmer, et al. Class-based construction of a verb lexicon. *AAAI/IAAI*, 691 :696, 2000.

G. Klein, Y. Kim, Y. Deng, J. Senellart, and A. M. Rush. OpenNMT : Open-source toolkit for neural machine translation. In *Proc. ACL*, 2017. doi : 10.18653/v1/P17-4012. URL https://doi.org/10.18653/v1/P17-4012.

P. Koehn, F. J. Och, and D. Marcu. Statistical phrase-based translation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 48–54. Association for Computational Linguistics, 2003.

P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, et al. Moses : Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions*, pages 177–180. Association for Computational Linguistics, 2007.

R. Kuhn and R. De Mori. A cache-based natural language model for speech recognition. *IEEE transactions on pattern analysis and machine intelligence*, 12(6) :570–583, 1990.

R. Lau, R. Rosenfeld, and S. Roukos. Trigger-based language models : A maximum entropy approach. In *Acoustics, Speech, and Signal Processing, 1993. ICASSP-93., 1993 IEEE International Conference on*, volume 2, pages 45–48. IEEE, 1993.

M. Lesk. Automatic sense disambiguation using mrd : how to tell a pine cone from an ice cream cone. In *Proceedings of SIGDOC '86*, pages 24–26, New York, NY, USA, 1986. ACM. ISBN 0-89791-224-1.

M.-T. Luong and C. D. Manning. Achieving open vocabulary neural machine translation with hybrid word-character models. *arXiv preprint arXiv :1604.00788*, 2016.

M. Maamouri, A. Bies, T. Buckwalter, and W. Mekki. The penn arabic treebank : Building a large-scale annotated arabic corpus. In *NEMLAR conference on Arabic language resources and tools*, volume 27, pages 466–467. Cairo, 2004.

M. P. Marcus, M. A. Marcinkiewicz, and B. Santorini. Building a large annotated corpus of english : The penn treebank. *Computational linguistics*, 19(2) :313–330, 1993.

W. S. McCulloch and W. Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4) :115–133, 1943.

R. McDonald, J. Nivre, Y. Quirmbach-Brundage, Y. Goldberg, D. Das, K. Ganchev, K. Hall, S. Petrov, H. Zhang, O. Täckström, et al. Universal dependency annotation

for multilingual parsing. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2 : Short Papers)*, volume 2, pages 92–97, 2013.

L. Medsker and L. C. Jain. *Recurrent neural networks : design and applications*. CRC press, 1999.

L. Merhben, A. Zouaghi, and M. Zrigui. Lexical disambiguation of arabic language : An experimental study. *Polibits*, (46) :49–54, 2012.

R. Mihalcea and T. Chklovski. *Building sense tagged corpora with volunteer contributions over the Web*, pages 357–402. John Benjamin Publishing Compagny, 2003.

T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc.

T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, 2013a.

T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013b.

G. A. Miller. Wordnet : a lexical database for english. *Communications of the ACM*, 38 (11) :39–41, 1995.

M. Nasiruddin, A. Tchechmedjiev, H. Blanchon, and D. Schwab. Création rapide et efficace d'un système de désambiguïsation lexicale pour une langue peu dotée. In *TALN 2015-22ème Conférence sur le Traitement Automatique des Langues Naturelles,, Caen, France*, 2015.

R. Navigli and S. P. Ponzetto. BabelNet : The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193 :217–250, 2012.

R. Navigli, K. C. Litkowski, and O. Hargraves. Semeval-2007 task 07 : Coarse-grained english all-words task. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 30–35. Association for Computational Linguistics, 2007.

H. T. Ng and H. B. Lee. Integrating multiple knowledge sources to disambiguate word sense : An exemplar-based approach. In *Proceedings of the 34th annual meeting on Association for Computational Linguistics*, pages 40–47. Association for Computational Linguistics, 1996.

Q.-P. Nguyen, A.-D. Vo, J.-C. Shin, and C.-Y. Ock. Effect of word sense disambiguation on neural machine translation : A case study in korean. *IEEE Access*, 6 :38512–38523, 2018.

A. Novischi, M. Srikanth, and A. Bennett. Lcc-wsd : System description for english coarse grained all words task at semeval 2007. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, SemEval '07, pages 223–226, Stroudsburg, PA, USA, 2007. Association for Computational Linguistics. URL http://dl.acm.org/citation.cfm?id=1621474.1621521.

F. J. Och and H. Ney. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1) :19–51, 2003.

C. Olah. Neural networks, types, and functional programming @ONLINE, 2015.

S. Padó and M. Lapata. Cross-lingual annotation projection for semantic roles. *Journal of Artificial Intelligence Research*, 36 :307–340, 2009.

M. Palmer, O. Babko-Malaya, A. Bies, M. T. Diab, M. Maamouri, A. Mansouri, and W. Zaghouani. A pilot arabic propbank. In *LREC*, 2008.

K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. Bleu : a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics, 2002.

A. Pasha, M. Al-Badrashiny, M. T. Diab, A. El Kholy, R. Eskander, N. Habash, M. Pooleery, O. Rambow, and R. Roth. Madamira : A fast, comprehensive tool for morphological analysis and disambiguation of arabic. In *LREC*, volume 14, pages 1094–1101, 2014.

A. Raganato, C. Delli Bovi, and R. Navigli. Neural sequence learning models for word sense disambiguation. In *Proceedings of the 2017 Conference on Empirical Methods*

in Natural Language Processing, pages 1167–1178. Association for Computational Linguistics, 2017. URL http://www.aclweb.org/anthology/D17-1121.

P. Resnik. Semantic similarity in a taxonomy : An information-based measure and its application to problems of ambiguity in natural language. *CoRR*, abs/1105.5444, 2011. URL http://arxiv.org/abs/1105.5444.

RFI. La langue française dans le monde @ONLINE, 2009.

F. Rosenblatt. *The perceptron, a perceiving and recognizing automaton Project Para*. Cornell Aeronautical Laboratory, 1957.

N. Schneider, B. Mohit, K. Oflazer, and N. A. Smith. Coarse lexical semantic annotation with supersenses : An arabic case study. 2012.

D. Schwab. Cours master mosig. 2017. URL http://lig-membres.imag.fr/blanchon/SitesEns/NLSP/.

D. Schwab, J. Goulian, and N. Guillaume. Désambiguïsation lexicale par propagation de mesures sémantiques locales par algorithmes à colonies de fourmis. In *Traitement Automatique des Langues Naturelles (TALN)*, Montpellier, France, 2011.

D. Schwab, J. Goulian, A. Tchechmedjiev, and H. Blanchon. Ant Colony Algorithm for the Unsupervised Word Sense Disambiguation of Texts : Comparison and Evaluation. In *Proceedings of the 25th International Conference on Computational Linguistics (COLING 2012)*, Mumbai (India), dec 2012.

D. Schwab, J. Goulian, and A. Tchechmedjiev. Worst-case complexity and empirical evaluation of artificial intelligence methods for unsupervised word sense disambiguation. *International Journal of Web Engineering and Technology*, 8(2) :124–153, 2013.

R. Sennrich, B. Haddow, and A. Birch. Neural machine translation of rare words with subword units. *arXiv preprint arXiv :1508.07909*, 2015.

L. Shi and R. Mihalcea. Putting pieces together : Combining framenet, verbnet and wordnet for robust semantic parsing. In *International conference on intelligent text processing and computational linguistics*, pages 100–111. Springer, 2005.

N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout : A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15(1) :1929–1958, Jan. 2014. ISSN 1532-4435. URL http://dl.acm.org/citation.cfm?id=2627435.2670313.

I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112, 2014.

K. Taghipour and H. T. Ng. One million sense-tagged instances for word sense disambiguation and induction. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning*, pages 338–344, Beijing, China, July 2015. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/K15-1037.

J. Tiedemann, Ž. Agić, and J. Nivre. Treebank translation for cross-lingual parser induction. In *Eighteenth Conference on Computational Natural Language Learning (CoNLL 2014)*, 2014.

L. van der Plas and M. Apidianaki. Cross-lingual word sense disambiguation for predicate labelling of french. *Proceedings of TALN 2014 (Volume 1 : Long Papers)*, 1 : 46–55, 2014.

J. Véronis. Cartographie lexicale pour la recherche d'information. *Actes de TALN 2003*, pages 265–274, 2003.

A. Verschaere. L'alphabet arabe : les 28 lettres @ONLINE, 2016.

L. Vial, A. Tchechmedjiev, and D. Schwab. Extension lexicale de définitions grâce à des corpus annotés en sens. In *23ème Conférence sur le Traitement Automatique des Langues Naturelles*, Paris, France, July 2016. URL https://hal.archives-ouvertes.fr/hal-01332850.

L. Vial, B. Lecouteux, and D. Schwab. Uniformisation de corpus anglais annotés en sens. In *24ème Conférence sur le Traitement Automatique des Langues Naturelles*, Orléans, France, June 2017. URL https://hal.archives-ouvertes.fr/hal-01599578.

L. Vial, B. Lecouteux, and D. Schwab. Approche supervisée à base de cellules lstm bidirectionnelles pour la désambiguïsation lexicale. In *25e conférence sur le Traitement Automatique des Langues Naturelles (TALN)*, 2018a.

L. Vial, B. Lecouteux, and D. Schwab. UFSAC : Unification of Sense Annotated Corpora and Tools. In *Language Resources and Evaluation Conference (LREC)*, Miyazaki, Japan, May 2018b. URL https://hal.archives-ouvertes.fr/hal-01718237.

L. Vial, B. Lecouteux, and D. Schwab. Approche supervisée à base de cellules LSTM bidirectionnelles pour la désambiguïsation lexicale. In *25e conférence sur le Traitement Automatique des Langues Naturelles*, Rennes, France, May 2018c. URL https://hal.archives-ouvertes.fr/hal-01781183.

P. Vossen, A. Görög, F. Laan, M. Van Gompel, R. Izquierdo-Bevia, and A. Van Den Bosch. Dutchsemco : building a semantically annotated corpus for dutch. In *Electronic lexicography in the 21st century : New Applications for New Users : Proceedings of eLex 2011, Bled, 10-12 November 2011*, pages 286–296, 2011.

R. Weischedel, M. Palmer, M. Marcus, E. Hovy, S. Pradhan, L. Ramshaw, N. Xue, A. Taylor, J. Kaufman, M. Franchini, M. El-Bachouti, R. Belvin, and A. Houston. Ontonotes release 5.0. *LDC2013T19. Web Download. Philadelphia : Linguistic Data Consortium*, 2015.

D. Yarowsky, G. Ngai, and R. Wicentowski. Inducing multilingual text analysis tools via robust projection across aligned corpora. In *Proceedings of the first international conference on Human language technology research*, pages 1–8. Association for Computational Linguistics, 2001.

D. Yuan, J. Richardson, R. Doherty, C. Evans, and E. Altendorf. Semi-supervised word sense disambiguation with neural models. In *COLING 2016*, 2016.

W. Zaghouani, M. Diab, A. Mansouri, S. Pradhan, and M. Palmer. The revised arabic propbank. In *Proceedings of the fourth linguistic annotation workshop*, pages 222–226. Association for Computational Linguistics, 2010.

O. Zennaki, N. Semmar, and L. Besacier. Utilisation des réseaux de neurones récurrents pour la projection interlingue d'étiquettes morpho-syntaxiques à partir d'un corpus parallèle. In *TALN 2015*, 2015.

A. Zouaghi, L. Merhbene, and M. Zrigui. Word sense disambiguation for arabic language using the variants of the lesk algorithm. *WORLDCOMP*, 11 :561–567, 2011.