Architectures neuronales multilingues pour le traitement automatique des langues naturelles

Adrien Bardet

DOCTORAT BRETAGNE LOIRE / MATHSTIC

Le Mans Université

THESE DE DOCTORAT DE

LE MANS UNIVERSITE

ECOLE DOCTORALE N° 601
Mathématiques et Sciences et Technologies de l'Information et de la Communication
Spécialité : *Informatique*

Par

Adrien BARDET

Architectures neuronales multilingues pour le traitement automatique des langues naturelles

Thèse présentée et soutenue à Le Mans IC2 et visioconférence, le 22/02/2021
Unité de recherche : LIUM
Thèse N° : 2021LEMA1002

Rapporteurs avant soutenance :

Claire Gardent	Professeur des Universités, LORIA, Vandoeuvre-lès-Nancy
Alexandre Allauzen	Professeur des Universités, LAMSADE, Paris

Composition du Jury :

Président :	Emmanuel Morin	Professeur des Universités, LS2N, Nantes
Examinateurs :	Claire Gardent	Professeur des Universités, LORIA, Vandoeuvre-lès-Nancy
	Alexandre Allauzen	Professeur des Universités, LAMSADE, Paris
	Nicolas Dugue	Maître de conférence, Le Mans Université
Dir. de thèse :	Sylvain Meignier	Professeur des Universités, Le Mans Université
Co-dir. de thèse :	Loïc Barrault	Senior Lecturer, University of Sheffield

Remerciements

Je tiens à remercier mon encadrant Loïc Barrault pour son suivi et ses conseils pendant l'ensemble de mon doctorat. Cette thèse est le fruit de plus de quatre années de collaboration à ses côtés.

Je voudrais aussi remercier mon autre encadrant Fethi Bougares pour ses conseils et son soutien ainsi que mon directeur de thèse, Sylvain Meignier, pour ses conseils justes et ses relectures avisées.

J'adresse tous mes remerciements à Claire Gardent et Alexandre Allauzen pour avoir accepté d'être rapporteurs pour cette thèse.

J'exprime ma gratitude à Emmanuel Morin et Nicolas Dugue qui ont bien voulu être examinateurs.

Je remercie mes collègues doctorants de l'équipe de traduction, Walid, Mercedes et Ozan pour leurs conseils, j'ai beaucoup appris à leurs côtés.

Je voudrais remercier mes collègues du LIUM et du département informatique, présents ou anciens, pour ces années de partage à vos côtés qui furent un soutien important.

Je voudrais remercier Etienne et Grégor pour le support technique et leurs conseils tout au long de la thèse ainsi qu'Anne-Cécile, Mélanie et Dominique qui ont su rendre simples les différentes formalités liées au doctorat.

Une mention spéciale pour Jane, Emmanuelle, Mélanie, Mercedes, Nicolas et Antoine qui sont vite devenus des amis en plus d'être collègues.

Je souhaite remercier ma famille, mes parents Nathalie et Laurent ainsi que ma sœur Lucie, pour leur soutien et leur écoute au fil de ces années.

Enfin, je souhaite remercier un groupe que j'ai rencontré, d'abord liés par le doctorat puis par l'amitié, je pense à Loredana, Juliette, Pauline, Moisés, Elodie, Clément, Alexandre, Dario, Lorenzo, Stanislas, Quentin et Tetiana pour leur soutien, leurs conseils et tous ces moments passés ensemble.

Résumé

La traduction des langues est devenue un besoin essentiel pour la communication entre humains dans un monde où les possibilités de communication s'élargissent. La traduction automatique est une réponse à l'évolution de ce besoin. Plus récemment, la traduction automatique neuronale s'est imposée avec les grandes performances des systèmes neuronaux qui ouvrent une nouvelle aire de l'apprentissage automatique. Les systèmes neuronaux exploitent de grandes quantités de données pour apprendre à réaliser une tâche automatiquement. Dans le cadre de la traduction automatique, les quantités de données parfois importantes et nécessaires pour apprendre des systèmes performants ne sont pas toujours disponibles pour toutes les langues. L'utilisation de systèmes multilingues est une solution pour répondre à ce problème. Les systèmes de traduction automatique multilingues permettent de traduire plusieurs langues au sein d'un même système. Ils permettent aux langues disposant de peu de données d'être apprises aux côtés de langues disposant de plus de données, améliorant ainsi les performances du système de traduction. Cette thèse se concentre sur des approches de traduction automatique multilingue en vue d'améliorer les performances pour les langues disposant de peu de données. J'ai travaillé sur plusieurs approches de traduction multilingue reposant sur différentes techniques de transfert entre les langues. La première emploie un système unique pour traduire plusieurs langues et ainsi cherche à améliorer les performances de ces langues ; c'est une approche de transfert multilingue. La seconde utilise un premier système traduisant des langues qui est utilisé, dans un second temps, pour apprendre à traduire de nouvelles langues disposant de peu de données. Le transfert séquentiel présenté dans cette approche permet d'améliorer les performances de traduction des langues disposant de peu de données en se basant sur des connaissances déjà assimilées par le système. La dernière approche proposée cherche à combiner les deux précédentes. Le transfert séquentiel multilingue utilise un système de traduction multilingue comme base d'apprentissage pour un système traduisant des langues disposant de peu de données. Cette approche permet de combiner différentes spécificités des langues qui se sont avérées pertinentes pour améliorer les performances de traduction des langues disposant de peu de données. Les différentes approches proposées ainsi que des analyses complémentaires ont révélé l'impact des critères pertinents pour le transfert. Elles montrent aussi l'importance, parfois négligée, de l'équilibre des langues au sein d'approches multilingues.

Abstract

The translation of languages has become an essential need for communication between humans in a world where the possibilities of communication are expanding. Machine translation is a response to this evolving need. More recently, neural machine translation has come to the fore with the great performance of neural systems, opening up a new area of machine learning. Neural systems use large amounts of data to learn how to perform a task automatically. In the context of machine translation, the sometimes large amounts of data needed to learn efficient systems are not always available for all languages. The use of multilingual systems is one solution to this problem. Multilingual machine translation systems make it possible to translate several languages within the same system. They allow languages with little data to be learned alongside languages with more data, thus improving the performance of the translation system. This thesis focuses on multilingual machine translation approaches to improve performance for languages with limited data. I have worked on several multilingual translation approaches based on different transfer techniques between languages. The first uses a single system to translate several languages and thus seeks to improve the performance of these languages ; it is a multilingual transfer approach. The second uses a first system translating languages which is used in a second stage to learn how to translate new languages with little data. The sequential transfer presented in this approach makes it possible to improve the translation performance of languages with little data, based on knowledge already assimilated by the system. The last proposed approach seeks to combine the two previous ones. Multilingual sequential transfer uses a multilingual translation system as a learning base for a system translating languages with little data. This approach makes it possible to combine different language specificities that have been shown to be relevant for improving the translation performance of languages with little data. The different approaches proposed, as well as additional analyses, have revealed the impact of the relevant criteria for transfer. They also show the importance, sometimes neglected, of the balance of languages within multilingual approaches.

TABLE DES MATIÈRES

LISTE DES TABLEAUX

TABLE DES FIGURES

INTRODUCTION

La langue est le support principal de communication des humains, mais la grande variété de langues utilisées à travers le monde est un obstacle au partage global des informations. Il est alors nécessaire de recourir à la traduction comme pont entre ces multiples langues. L'essor d'internet, à lui seul, met à disposition de grandes quantités d'informations. La traduction de ces informations permettrait un accès dans la langue maternelle de chacun.

L'anglais est prédominant sur internet (figure 1). Il représente plus de 50% du contenu, le reste est réparti à travers les autres langues avec, en tête, le russe, le japonais et l'allemand. Les autres langues, plusieurs centaines, représentent les 50 autres pourcents.

La traduction humaine est considérée comme la meilleure solution pour obtenir une qualité suffisante. Par exemple, la traduction de documents officiels doit être parfaite. Dans ce cas précis, la traduction automatique n'est pas adaptée et la traduction humaine ne peut plus répondre à la forte augmentation des demandes. La traduction assistée par ordinateur est un procédé où les traducteurs humains corrigent les traductions proposées par la machine. Cette collaboration entre le correcteur et le système de traduction permet de produire rapidement un plus grand nombre de traductions de qualité élevée. Mais cette solution hybride reste insuffisante face aux besoins, la traduction automatique reste la solution privilégié.

La traduction automatique, faite par des machines, fait partie du domaine du traitement des langues naturelles (NLP - Natural Language Processing). Elle utilise les méthodes d'apprentissage automatique pour entraîner les systèmes de traduction. L'apprentissage automatique, qu'il soit statistique ou neuronale, nécessite de grandes quantités de données disponibles dans de nombreuses langues. Cependant, toutes les langues et tous les dialectes ne disposent pas de ressources suffisantes

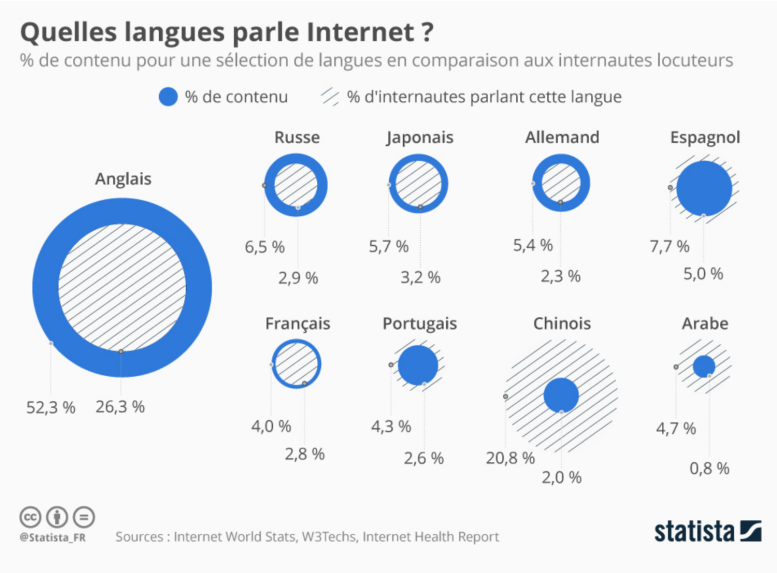

FIGURE 1 – Schéma de répartition des langues du contenu d'internet et de leurs internautes locuteurs. Ce schéma est extrait du site statista.com.

pour apprendre des systèmes performants (figure 1). De plus en plus, les systèmes de traduction reposent sur des méthodes génériques qui dépendent peu des langues qu'ils traduisent, et qui nécessitent de moins en moins de recourir à des spécialistes de la langue. L'automatisation de la traduction, et plus globalement du traitement des langues naturelles, passe aujourd'hui par les systèmes neuronaux. Ces systèmes s'inspirent du cerveau humain, ou plus précisément de l'interaction entre les neurones, et apprennent à représenter dans leurs «cerveaux» artificiels les concepts des langues. Un entraînement correctement paramétré permet, avec les données adéquates, de réaliser des systèmes de traduction automatique performants et rapides. Les approches utilisant les systèmes neuronaux définissent un domaine à part qu'est la traduction automatique neuronale (NMT - Neural Machine Translation).

Cette thèse s'inscrit dans le cadre du projet Chistera *M2CR*[1] pour «Multimodal Multilingual Continuous Representations» que l'on peut traduire en français par : Représentations Continues Multilingues et Multimodales. Le projet M2CR vise à combiner plusieurs modalités de la communication humaine dans une architecture unique, basée sur des réseaux neuronaux. Les approches développées cherchent à exploiter des représentations vectorielles d'images, de textes et d'audio en plusieurs langues. Le postulat de départ est que la sémantique est répartie à travers les différentes modalités. Le but de la traduction est de traduire un sens, quelque soit la modalité le véhiculant. Représenter ces modalités dans un espace commun pourrait

1. https ://projets-lium.univ-lemans.fr/m2cr/

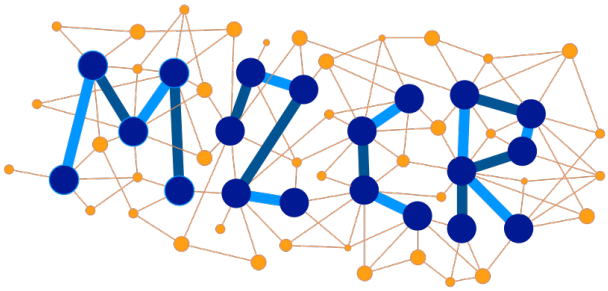

FIGURE 2 – Logo du projet M2CR

permettre une meilleure compréhension. Les multiples langues peuvent être vues comme de multiples modalités car elle sont chacune, à leur façon, une représentation différente d'une même idée. Les tâches applicatives du projet sont la compréhension de la parole, la traduction automatique, la description multilingue d'image.

Dans ce projet, ma thèse s'inscrit dans le cadre de la traduction automatique pour la représentation textuelle multilingue. Nous cherchons à faire le lien entre les langues et analyser les liens entre elles.

Les humains ont la faculté d'exploiter des connaissances antérieures pour faciliter l'apprentissage d'une nouvelle langue. Les phrases dans différentes langues ont toutes pour but de transmettre une information. Ce postulat nous conduit à penser qu'au sein d'un système multilingue, les représentations obtenues à partir de plusieurs langues pourraient faciliter l'apprentissage d'une nouvelle paire de langues, notamment pour les paires où moins de données sont disponibles.

Entrainer des systèmes de traduction automatique performants est un processus lourd et coûteux en ressources (temps, puissance de calcul, énergie). Réutiliser des systèmes existants est une possibilité. Des techniques de transfert permettent d'adapter un système existant pour traduire de nouvelles langues dans le cadre de la traduction automatique. C'est ce que nous appelons l'apprentissage par transfert. Les faibles quantités de données disponibles pour entrainer des systèmes de traduction automatique sont un frein à l'obtention de bonnes performances. La réutilisation de systèmes existants permet de compenser les faibles quantités de données en exploitant celles précédemment apprises par le système.

L'étude de l'état de l'art autour du domaine de l'apprentissage par transfert soulève plusieurs questions. Nous cherchons à apporter des éléments de réponse à celles-ci dans le cadre de la traduction automatique.

Pour améliorer les performances de langues où peu de données sont disponibles, il faut se poser la question de **ce qui est pertinent pour le transfert** (données ou composants des architectures automatiques par exemple). Il faut aussi s'intéresser à

comment transférer ces informations d'une langue à une autre dans un système automatique. La sélection des informations ainsi que les techniques employées pour le transfert sont importantes pour s'assurer de la performance du système qui en résulte. On peut aussi soulever la question de **savoir si le transfert est pertinent** pour les langues sélectionnées. Nous verrons que le transfert peut être négatif dans certains cadres.

L'objectif de mes travaux est une analyse du transfert autour de plusieurs critères importants pour sa performance. Nous verrons notamment que les quantités de données et les proximités entre les langues ont un impact significatif sur le transfert. Nous chercherons à expliquer comment ces critères impactent le transfert à travers les performances obtenues et nous ferons une analyse de l'évolution des composants des systèmes de traduction automatique. Nous verrons dans les contributions de cette thèse que les approches proposées ont fourni des réponses à ces différentes questions.

Ce document est composé de six chapitres. Un premier chapitre pose le cadre d'étude de la thèse. Nous allons présenter l'état de l'art de la traduction automatique neuronale dans un second chapitre. Le second chapitre pose les fondements des techniques utilisées en traduction automatique. Nous présenterons les principales techniques d'apprentissage automatique employées dans des systèmes de traduction, à base de systèmes neuronaux. Nous verrons également plusieurs spécificités de la traduction automatique tels que les pré-traitements de données.

Le troisième chapitre présentera l'état de l'art des approches d'apprentissage par transfert. Ces approches permettent d'améliorer les performances des systèmes dans le cadre de la traduction automatique grâce à d'autres langues.

Les contributions de cette thèse sont regroupées en trois chapitres. Le quatrième chapitre présente mes travaux sur l'apprentissage par transfert multilingue. Il propose notamment une approche qui permet de spécialiser certaines parties de l'architecture neuronale aux langues qui sont traduites pour améliorer leurs performances.

Le cinquième chapitre porte sur le transfert séquentiel. Cette forme de transfert réalisé en plusieurs étapes entre plusieurs systèmes de traduction est particulièrement efficace lorsque peu de données sont disponibles pour réaliser un système de traduction automatique. Nous proposerons une analyse des réseaux de neurones employés pour mieux comprendre ce transfert et l'impact qu'il a directement sur les réseaux de neurones. Un dernier chapitre présentera une approche de transfert séquentiel multilingue. Cette approche cherche à combiner le transfert séquentiel et le transfert multilingue en utilisant un système multilingue comme source d'un transfert pour un second système de traduction. Nous verrons que ces expériences

ont soulevé plusieurs questions sur l'équilibre des langues au sein des systèmes multilingues pour lesquelles nous proposerons une réponse.

Première partie

État de l'art

CHAPITRE 1

CADRE GÉNÉRAL

Dans ce premier chapitre nous allons introduire le cadre général de la thèse avec une présentation de la traduction et ses différentes composantes. Nous discuterons des principes de traduction, des ressources nécessaires à la traduction automatique ainsi que des métriques utilisées pour l'évaluer. Enfin, nous ferons un rapide historique de la traduction automatique et présenterons plusieurs axes d'études récents en traduction automatique.

1.1 La traduction

La traduction automatique est la traduction d'un texte dans une langue (dite langue source) vers une autre langue (dite langue cible) réalisée par une machine. La traduction automatique est à différencier de la traduction assistée par ordinateur, qui elle, implique l'intervention humaine lors du processus de traduction. Elle peut aussi être dite interactive avec une coopération entre la machine et son utilisateur.

La figure 1.1 est un exemple d'un système de traduction traduisant du français vers l'anglais.

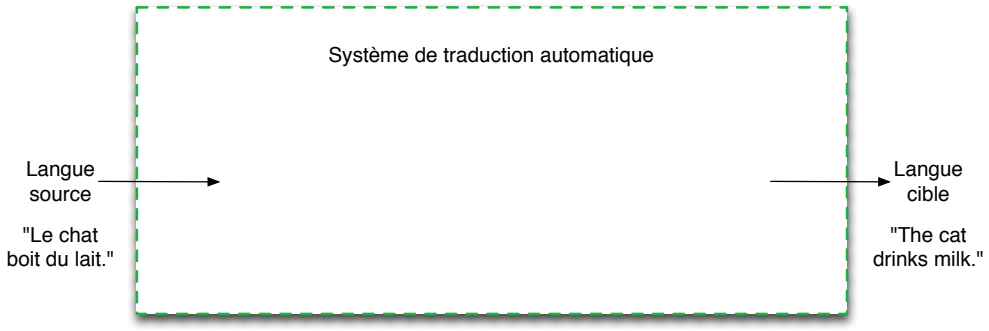

FIGURE 1.1 – Exemple de système de traduction automatique français-anglais.

Dans le cadre de la traduction automatique, il n'y a pas d'intervention humaine. Un ou plusieurs algorithmes composent le système de traduction qui traduit d'une langue source s vers une langue cible c.

$$c^* = \underset{c}{\operatorname{argmax}}\, P(c|s) \tag{1.1}$$

$$c^* = \underset{c}{\operatorname{argmax}}\, P(s|c)P(c) \tag{1.2}$$

L'objectif d'un système de traduction (équation 1.1) est de trouver la phrase cible c^* qui maximise la probabilité $P(c|s)$ où c représente la phrase cible et s la phrase source. Le théorème de Bayes nous donne l'équation 1.2. Il reste à maximiser $P(s|c)$ correspondant à la probabilité de la phrase source s sachant la phrase cible c qui sera grande si s et c sont des traductions possibles. La probabilité $P(c)$ est grande si la suite de mots est bien formée, elle est obtenue par un modèle de langage.

1.2 Ressources en traduction automatique

Les systèmes de traduction automatique utilisent un vocabulaire. Il comprend les mots que le système pourra prendre en compte et utiliser pour composer une traduction. La taille des vocabulaires est un critère important, notamment pour les systèmes neuronaux, car elle conditionne le nombre de paramètres et donc la complexité finale du système. Il est nécessaire de trouver le juste équilibre qui, avec le plus petit nombre de mots, donne les meilleures traductions. Une des premières étapes est la normalisation des mots.

Pour l'apprentissage de systèmes de traduction automatique, nous avons besoin de corpus de données. Ils peuvent prendre plusieurs formes. Généralement, ils sont bilingues, composés d'une langue source et d'une langue cible. Les phrases de ces deux corpus sont alignées afin qu'une phrase dans un corpus corresponde à sa traduction dans l'autre. De cette façon, le système peut apprendre à générer, à partir d'une phrase source, la phrase cible qui est sa traduction. Il est aussi possible d'exploiter des données monolingues, avec des techniques pour améliorer un système classique ou en essayant directement de traduire sans données parallèles (Lample et al., 2018; Sennrich et al., 2016a; Gülçehre et al., 2015).

Les quantités de ces données sont un facteur clé pour l'apprentissage de systèmes automatiques. Ils sont le plus souvent performants lorsqu'ils sont appris sur de grands corpus.

Un système de traduction automatique classique traduit une seule paire de langues. Pour traduire N paires de langues, il faut donc développer N systèmes. Pour s'affranchir de cette limite, la solution la plus explorée est l'utilisation de systèmes de traduction automatique multilingues. Les systèmes multilingues ont pour vocation de traduire plusieurs paires de langues et ainsi limiter le nombre de systèmes nécessaires lorsque plusieurs paires de langues sont prises en compte.

1.3 Évaluation automatique des traductions

Pour entrainer un système de traduction automatique, il faut optimiser ses paramètres. Pour cela, le système doit être évalué régulièrement. La meilleure évaluation possible reste celle effectuée par des traducteurs humains. Cependant, cette approche est très coûteuse, que ce soit en temps ou en argent, et donc difficilement applicable à l'échelle, lorsque beaucoup de systèmes de traduction sont réalisés et où plusieurs paires de langues sont impliquées.

Pour contourner ce problème, l'évaluation des systèmes de traduction automatique est elle aussi réalisée automatiquement. De nombreuses métriques d'évaluation tels que le BLEU (Papineni et al., 2002), le METEOR (Banerjee and Lavie, 2005) ou encore SIMILE (Wieting et al., 2019) se veulent toujours être les plus corrélées possibles à l'évaluation humaine. Les métriques employées en traduction automatique sont un domaine d'étude actif (Madnani et al., 2012; Munkova et al., 2020).

L'automatisation du processus d'évaluation a plusieurs avantages. Il permet d'évaluer fréquemment le système au cours de son développement. Les métriques automatiques reposent le plus souvent sur des comparaisons entre les mots de traduction proposés et les mots de la référence correspondant à la traduction faite par un ou plusieurs annotateurs humains. L'avantage de cette approche est que

les métriques sont utilisables sans ressources ou connaissances propres aux langues évaluées.

La métrique la plus utilisée est le BLEU (Papineni et al., 2002). C'est une moyenne géométrique des précisions n-grams communs entre l'hypothèse et une ou plusieurs références. Les résultats en BLEU sont compris entre 0 et 1. Plus le score est élevé, meilleure est la traduction.

L'utilisation de références multiples permet une évaluation fine de la traduction candidate qui prend en compte les différentes façons de traduire un texte. Cependant, il est rare d'avoir plusieurs références.

Le calcul du BLEU est donné dans les équations 1.3 et 1.4.

$$BLEU = BP \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right) \tag{1.3}$$

ou N représente l'ordre des n-gram pris en compte, w_n représente les poids des n-gram dont le total vaut 1, p_n représente les précisions n-gram modifiées. Cette métrique est calculée par une moyenne géométrique des précisions n-grams multipliées par une pénalité de brièveté (BP). Les précisions n-grams sont modifiées car elles prennent en compte le nombre d'occurrences du n-gram dans la référence pour éviter qu'il ne soit compté plusieurs fois et augmente le BLEU artificiellement.

Le BLEU utilise une pénalité de briéveté (BP) qui pénalise les traductions trop courtes. Elle est calculée de la façon suivante :

$$BP = \begin{cases} 1 & \text{si c} > r \\ \exp\left(1 - r/c\right) & \text{si c} \leq r \end{cases} \tag{1.4}$$

c représente la longueur de traduction candidate, r la longueur de la traduction de référence.

Le score BLEU s'exprime le plus souvent en pourcentage.

1.4 Rapide historique de la traduction automatique

Les premiers travaux sur l'automatisation de la traduction débutent dans les années 50 avec les travaux de Warren Weaver (Hutchins, 2001). Ils s'intéressent à des approches de traduction à base de règles qui utilisent des informations linguistiques pour créer une liste de règles de traduction.

Un fort regain d'intérêt est apparu dans les années 90 avec l'arrivée des systèmes statistiques (Brown et al., 1990, 1993). L'apprentissage de modèles statistiques pour la traduction a été rendu possible en utilisant des corpus monolingues et bilingues

de plusieurs millions de phrases.

Les années 2013-2014 et l'accès à de toujours plus grands corpus et au développement du calcul scientifique hautement parallèle sur carte GPU [1] ont permis aux approches utilisant des réseaux de neurones de révéler leurs potentiels (Sutskever et al., 2014; Bahdanau et al., 2015). Lecun et al. (1998); Hihi and Bengio (1996) avaient déjà proposé la théorie autour de ces approches à la fin des années 90. Cependant, il aura fallu de nombreuses années pour quelles s'imposent.

1.5 La traduction automatique aujourd'hui

Historiquement, les recherches en traduction automatique se sont focalisées sur la traduction d'une paire de langues, composée d'une langue source traduite vers une langue cible. Cependant, il est aussi possible de traduire plusieurs paires de langues avec un unique système de traduction. Nous parlerons de traduction multilingue. Traduire une paire de langues avec des racine communes (français - italien) est plus facile que de traduire des paires de langues sans parenté (français - basque) ou avec des systèmes d'écriture différents (français - chinois).

La traduction automatique ne se résume pas à de la traduction de textes. Il est possible de traduire différentes modalités telles que l'audio ou encore l'image. Le projet M2CR est un bon exemple de la capacité qu'ont les méthodes automatiques à s'adapter à différentes modalités. Lorsque plusieurs sources de natures différentes sont disponibles, nous parlons de traduction multimodale (Caglayan, 2019).

1. GPU : Graphics Processing Unit

CHAPITRE 2

TRADUCTION AUTOMATIQUE NEURONALE

Les systèmes neuronaux sont fondés sur les réseaux de neurones dont les fondements remontent jusqu'aux années 50 (Wang et al., 2017).

L'évolution des systèmes neuronaux et de leurs performances joue un rôle important depuis quelques années pour la traduction automatique. En effet, ils sont devenus l'approche la plus efficace pour obtenir de bonnes performances de traduction comme en attestent les campagnes d'évaluation WMT18 (Bojar et al., 2018) et WMT19 (Barrault et al., 2019).

Ce chapitre introduit la traduction automatique neuronale et ses différentes avancées. Les pré-traitements utilisés sont aussi présentés en fin de chapitre.

2.1 Réseaux de neurones

Les systèmes neuronaux sont directement inspirés du cerveau humain. Comme lui, ils sont composés de neurones qui communiquent et apprennent. Un neurone artificiel est une unité de calcul. Son potentiel d'activation est défini comme la somme pondérée de ses entrées. Sa sortie est calculée suivant cette somme pondérée et sa fonction d'activation. Une fonction d'activation est une fonction mathématique appliquée en sortie d'un neurone artificiel. Lorsqu'un seuil de stimulation est atteint, elle entraîne une réponse du neurone. Les neurones sont regroupés en couches qui forment une nouvelle unité de calcul logique. De multiples couches interconnectées forment un réseau de neurones multicouches (multilayer).

Les systèmes neuronaux regroupent l'ensemble des techniques d'apprentissage automatique utilisant des neurones. Dans cette section, nous présentons les principales approches neuronales utilisées en traduction automatique.

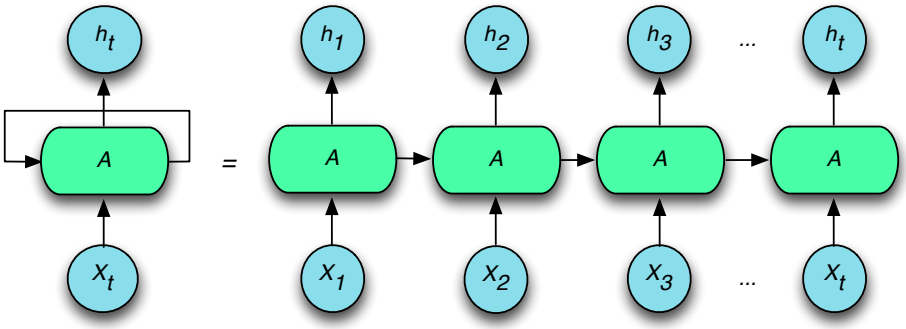

FIGURE 2.1 – Description de la récurrence dans un réseau récurrent

2.1.1 Réseaux neuronaux récurrents

L'apparition des réseaux neuronaux récurrents (RNN) (Rumelhart et al., 1986) a été l'un des premiers pas vers l'avènement des systèmes neuronaux de traduction automatique.

Les réseaux récurrents utilisent la rétropropagation du gradient pour apprendre les poids du réseau. Le gradient est un vecteur composé de valeurs proportionnelles à l'erreur commise lors de l'apprentissage d'une donnée d'un corpus. Le gradient traverse ensuite le réseau en sens inverse par l'algorithme de rétropropagation. L'algorithme rétropropage le gradient de l'erreur et met à jour les poids du modèle afin de minimiser l'erreur du modèle à la prochaine itération, c'est une procédure d'optimisation des poids.

$$W_{t+1} = W_t - \lambda(\frac{\partial E}{\partial W_t}) \tag{2.1}$$

La formule 2.1 présente le fonctionnement de la mise à jour des poids. W_{t+1} représente le nouveau poids après la mise à jour, W_t l'ancien poids. Le taux d'apprentissage (*learning rate*) est représenté par λ et l'erreur par E.

Une cellule de réseaux neuronaux récurrents est composée d'un état caché et d'une sortie optionnelle. La figure 2.1 présente la récurrence d'un RNN comme un enchaînement de cellules récurrentes dont l'état évolue. Un RNN classique est composé d'une entrée X_t, d'une sortie h_t et d'un réseau de neurones A qui utilise l'information provenant de l'état précédent dans une boucle.

La figure 2.2 décrit la composition d'une unité de réseau récurrent. Son fonctionnement est le suivant : l'état précédent h_{t-1} et l'entrée X_t sont passés à une fonction d'activation non linéaire tangente hyperbolique ($tanh$) qui détermine la valeur de sortie de la cellule. À l'instant t, l'état courant est stocké dans h_t. Étant

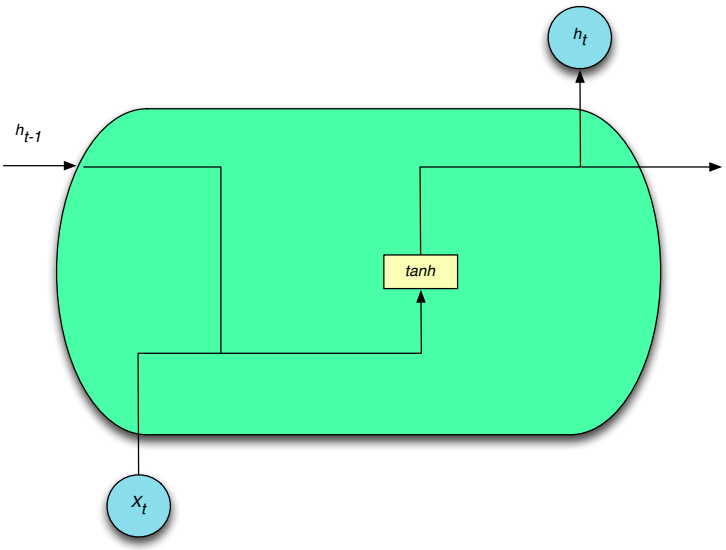

FIGURE 2.2 – Cellule d'unité de réseau récurrent

récurrent, ce processus est répété à chaque nouvelle valeur d'entrée. Il permet de prendre en compte l'état précédent qui contient, par récurrence, les autres états précédents pour calculer l'état actuel. Cependant, lors de l'apprentissage, le passage répété du gradient dans l'unité tend à accumuler une valeur qui peut devenir toujours plus grande ou plus petite jusqu'à ne plus faire fonctionner le réseau. En effet, l'augmentation rapide des valeurs des gradients pendant la rétropropagation peut rendre l'apprentissage instable jusqu'à entraîner un arrêt de l'apprentissage si ce nombre dépasse la capacité de représentation interne des nombres. C'est ce que nous appelons *l'explosion du gradient*. Elle est prévenue en définissant une valeur maximale que le gradient peut atteindre. De la même manière, de petites valeurs sont aussi problématiques, elles mènent à la disparition du gradient et à l'arrêt de l'apprentissage. Ces phénomènes ont poussé à la conception de nouvelles solutions pour répondre à ces problèmes.

2.1.2 Long Short-Term Memory

L'utilisation d'unités récurrentes à portes permet de répondre au problème de disparition du gradient.

Les *Long Short-Term Memory* (LSTM) ou cellule à Mémoire Longue à Court-Terme, proposées par Hochreiter and Schmidhuber (1997), offrent une solution au problème du gradient qui s'évanouit/disparait. Les LSTM introduisent des portes, elles sont une alternative aux neurones classiques.

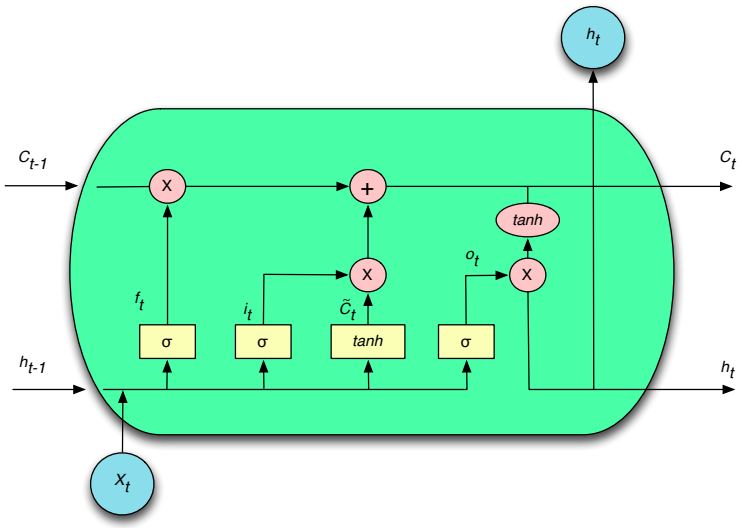

FIGURE 2.3 – Cellule LSTM (Long Short Term Memory)

En effet, lorsque des données passent par la cellule, elle sont proposées sous forme d'une activation candidate. Cette activation dépend de l'entrée courante X_t, ainsi que de la mémoire de l'état précèdent de la cellule h_{t-1}. De cette façon, la cellule prend en compte les contextes précédents et décide si les nouvelles données sont pertinentes. Cela permet à la cellule de se concentrer sur les informations importantes, de garder en mémoire les contextes intéressants et ainsi de filtrer les informations moins utiles.

L'équation 2.2 et la figure 2.3 présentent le fonctionnement d'une cellule LSTM.

$$\begin{aligned} f_t &= \sigma\left(W_f \cdot [h_{t-1}, X_t] + b_f\right) \\ i_t &= \sigma\left(W_i \cdot [h_{t-1}, X_t] + b_i\right) \\ o_t &= \sigma\left(W_o\,[h_{t-1}, X_t] + b_o\right) \\ \tilde{C}_t &= \tanh\left(W_C \cdot [h_{t-1}, X_t] + b_C\right) \\ C_t &= f_t * C_{t-1} + i_t * \tilde{C}_t \\ h_t &= o_t * \tanh\left(C_t\right) \end{aligned} \tag{2.2}$$

L'état de la cellule C_t repose sur la multiplication de la porte d'oubli f_t et la mémoire précédente C_{t-1} ainsi que la multiplication de la porte d'entrée i_t par l'activation candidate $\tilde{C}_t$. L'évolution de l'état caché h_t dépend de la porte de sortie o_t multiplié par l'activation de C_t. Les portes d'oubli f_t, d'entrée i_t et de sortie o_t sont calculées par activation sigmoïde suivant les poids des neurones W, l'entrée de la cellule X, la valeur de la couche précédente h_{t-1} et le biais b. L'état candidat $\tilde{C}_t$ est

défini de façon similaire mais avec une fonction d'activation tangente hyperbolique (tanh).

Les LSTM et de façon plus générale, les unités récurrentes à portes permettent de se focaliser sur les parties pertinentes de la séquence pour les prochaines sorties générées, en oubliant, ou en conservant certaines parties de celle-ci.

2.1.3 Gated Recurrent Unit

Les *Gated Recurrent Unit* (GRU) soit Unité Récurrentes à Portes (Cho et al., 2014), sont une alternative aux LSTM. Elles sont plus simples, composées de moins de portes. L'équation 2.3 et la figure 2.4 présentent le fonctionnement d'une cellule GRU.

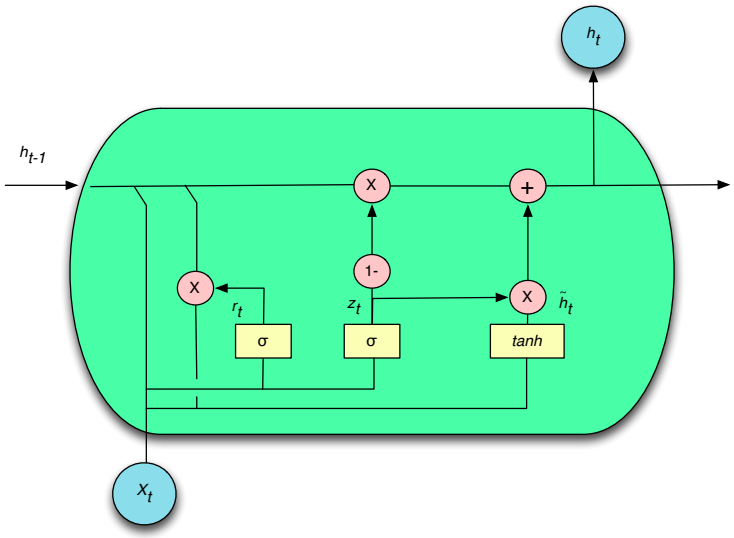

FIGURE 2.4 – Cellule GRU (Gated Recurrent Unit)

$$
\begin{aligned}
z_t &= \sigma\left(W_z \cdot [h_{t-1}, X_t]\right) \\
r_t &= \sigma\left(W_r \cdot [h_{t-1}, X_t]\right) \\
\tilde{h}_t &= \tanh\left(W_h \cdot [r_t * h_{t-1}, X_t]\right) \\
h_t &= (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t
\end{aligned}
\tag{2.3}
$$

La porte de mise à jour (*update gate*) z_t et la porte de remise à zero (*reset gate*) r_t sont calculées par activation sigmoïde suivant les poids des neurones W, l'entrée de la cellule X, la valeur de la couche précédente h_{t-1} et le biais b. L'état candidat $\tilde{h}_t$ fonctionne de façon similaire utilisant cette fois une fonction d'activation tangente hyperbolique (tanh). Cependant, il prend aussi en compte la porte de remise à zéro.

L'état de la cellule h_t est obtenu avec la multiplication de la porte de mise à jour z_t et l'état de la couche précédente h_{t-1} ainsi que la multiplication de la porte de mise à jour z_t et l'état candidat $\tilde{h}_t$.

En traduction automatique les GRU obtiennent des résultats similaires aux LSTM. Cependant, leur simplicité requiert moins de calculs et offre donc de meilleures performances en temps de calcul.

2.1.4 Encodeur / Décodeur

Les systèmes de traduction automatique neuronaux actuels sont fondés sur une architecture encodeur/décodeur (Sutskever et al., 2014). L'utilisation d'approche de type encodeur/décodeur utilise deux réseaux neuronaux récurrents. Le premier est l'encodeur qui transforme son entrée en une représentation plus ou moins complexe (vecteur, matrice, etc.). Cette représentation est utilisée par le second réseau qu'est le décodeur pour générer la sortie. Dans le cadre de la traduction automatique, une phrase en langue source est donnée à l'encodeur. La représentation qui en résulte est utilisée par le décodeur pour générer une phrase dans la langue cible.

Les systèmes de bout en bout (Sutskever et al., 2014)(*end to end* en anglais) permettent de traduire une phrase en prenant en compte la phrase source dans sa totalité.

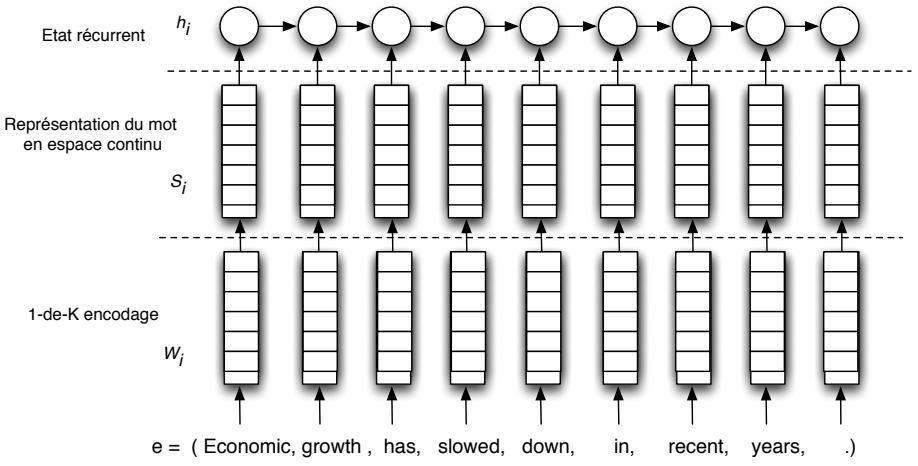

FIGURE 2.5 – Architecture encodeur / décodeur : l'encodeur

Dans la figure 2.5, la phrase e en anglais est la phrase que nous souhaitons encoder dans notre système de traduction. L'état de l'encodeur, à chaque étape, est calculé pour chaque mot i de la phrase source. Ces mots sont traités les uns après les autres. Le mot est représenté par une entrée du vocabulaire qui correspond à

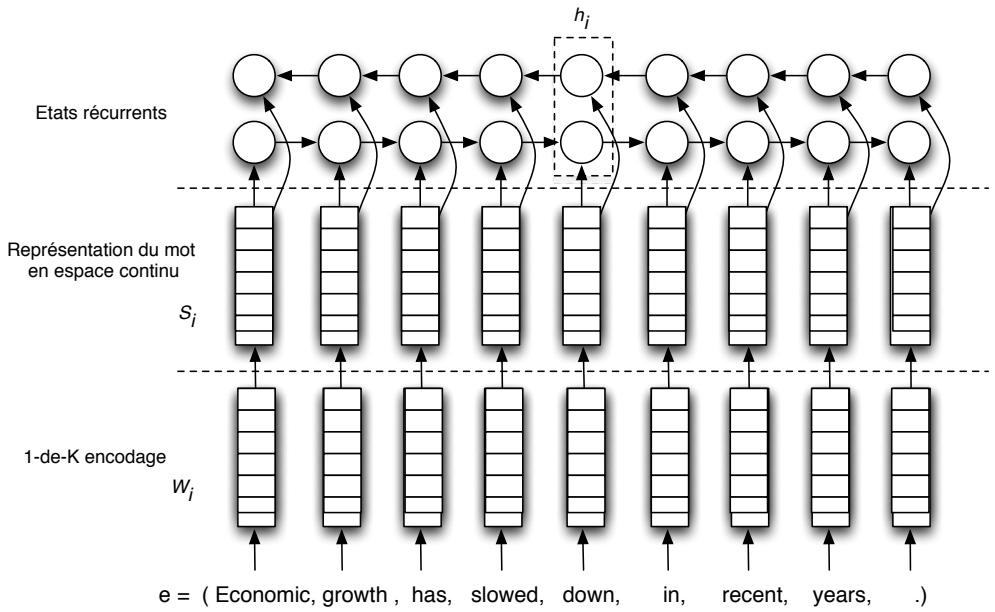

FIGURE 2.6 – Architecture encodeur / décodeur : l'encodeur bi-directionnel

l'encodage 1-de-K (W_i) où K est la taille de notre vocabulaire. Ensuite, le vecteur W_i est projeté dans un espace continu (*embeddings*/plongement de mot) noté S_i qui va représenter notre mot. Cette représentation S_i est fournie à la cellule récurrente qui va mettre à jour son état interne/caché h_i de l'encodeur. L'état va évoluer pour chaque mot qui lui est fourni, et la récurrence permet au dernier état d'être un vecteur qui représente la phrase source complète.

Dans la figure 2.6, notre exemple utilise un encodeur bidirectionnel. L'encodage est donc réalisé dans les deux sens de la phrase. h_i est maintenant une annotation composée des deux états récurrents. Pour chaque mot, une représentation de celui-ci est obtenue en tenant compte des mots qui le précèdent (pour le RNN lisant la phrase de gauche à droite) et qui le suivent (pour le RNN lisant la phrase de droite à gauche). Chaque représentation de mot est en fait une représentation de la phrase entière, mais avec un focus sur le mot lui-même car c'est la dernière entrée utilisée pour mettre à jour les 2 RNNs. L'encodage bidirectionnel a l'avantage, par rapport à un encodage unidirectionnel, de prendre en compte le contexte futur en plus du contexte passé du mot.

Le décodeur est également un RNN mais il traite des informations différentes. Il utilise comme entrée la représentation de la phrase source générée par l'encodeur. En prenant en compte les informations du mot précédant, le décodeur s'assure de formuler une suite de mots la plus probable dans la langue cible (figure 2.7). Le mot

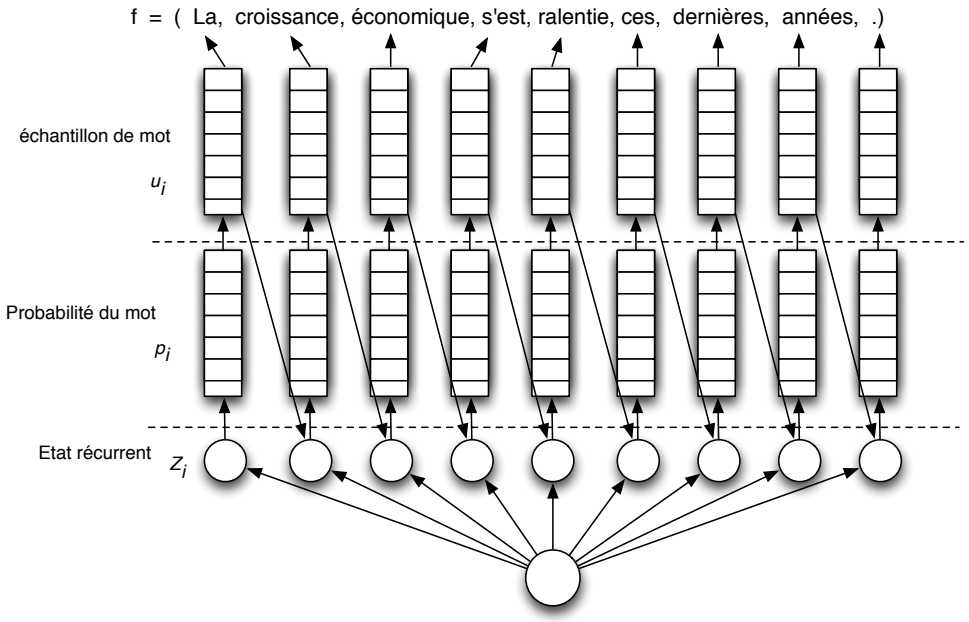

FIGURE 2.7 – Architecture encodeur / décodeur : le décodeur

le plus probable est obtenu en appliquant la fonction softmax qui permet d'obtenir une distribution de probabilités sur l'ensemble des mots du vocabulaire P_i. u_i est un vecteur 1-de-K où K est la taille de notre vocabulaire. L'état interne du décodeur Z_i évolue à chaque mot décodé.

On obtient au final, en sortie la phrase cible en français, traduction de la phrase source fournie à l'encodeur.

2.1.5 Mécanisme d'attention

La modélisation de la phrase repose sur la puissance des RNNs grâce à la récurrence. Cependant, cela n'est pas toujours performant pour capturer les dépendances à longue distance entre les mots. Bahdanau et al. (2015) présentent un mécanisme d'attention qui permet d'assigner des poids à chaque mot de la phrase source pour traduire le mot courant de la phrase cible.

L'architecture présentée dans la figure 2.8 est celle d'un mécanisme d'attention utilisant un encodeur bidirectionnel tel que présenté dans la section 2.1.4. Cet encodeur fournit des annotations de la phrase source (X_1,X_2,X_3, . . . , X_T dans la figure). La particularité des systèmes avec attention est la présence d'un troisième réseau de neurones appelé mécanisme d'attention qui est un réseau à propagation avant (*feed-forward*).

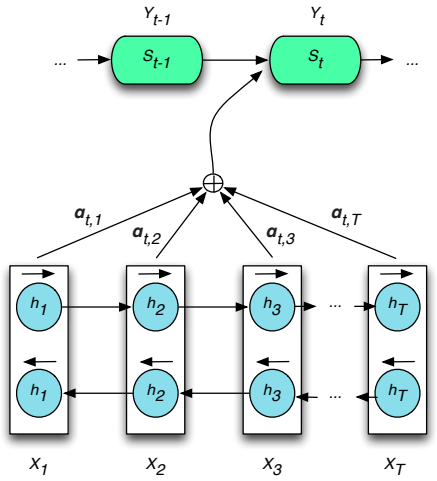

FIGURE 2.8 – Mécanisme d'attention avec encodeur bidirectionnel qui génère le mot cible Y_t en assignant des poids $\alpha_{t,T}$ aux annotations de la phrase source (X_1, X_2, X_3, . . . , X_T)

Au lieu de simplement prendre la représentation de la phrase, nous avons maintenant des poids $\alpha_{t,T}$ sur les différentes annotations de la phrase source. Le contexte est calculé à chaque étape du décodage en portant attention aux mots de la phrase source qui sont importants pour générer le mot suivant. Le mécanisme d'attention permet de mieux prendre en compte les dépendances à longues distances des mots dans la phrase source. Pour cela, il donne à ces mots pertinents pour la traduction des poids plus élevés qu'aux autres.

2.1.6 Transformers

Les systèmes *transformers* (Vaswani et al., 2017) obtiennent les meilleures performances lors des évaluations de 2017 à 2020 en traduction automatique. Les *transformers* utilisent des couches d'auto-attention (*self-attention*) multiples (*multi-head*) qui sont intégrées à l'encodeur et au décodeur. De cette façon, il se différencie du mécanisme d'attention classique qui est un réseau de neurones à part. L'auto-attention permet de calculer des poids d'attention sur l'ensemble de la phrase source sans utiliser d'architecture récurrente.

La figure 2.9 [1] présente l'architecture *transformers* décrite dans Vaswani et al. (2017). Dans cette figure, la colonne de gauche représente l'encodeur et celle de droite le décodeur. Pour chaque bloc de l'encodeur du *transformers*, les mots de la phrase source passent par une couche d'attention multiple (*multi-head* attention).

1. Figures 2.9 et 2.10 provenant de Vaswani et al. (2017)

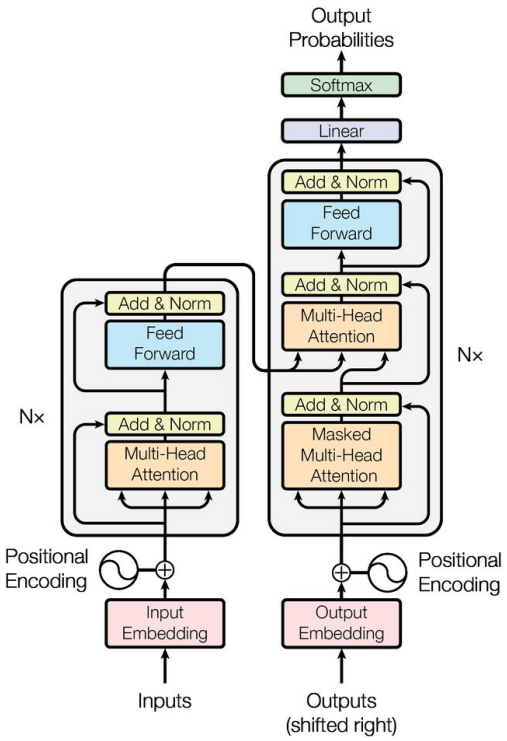

FIGURE 2.9 – Architecture réseau *transformers*

Pour chaque phrase en entrée, 3 vecteurs sont créés ; la requête, la clé et la valeur. Les clés et valeurs sont de dimensions égales à la taille de la séquence d'entrée. Elles sont les états cachés de l'encodeur. Dans le décodeur, la sortie précédente forme la requête. La sortie du décodeur est obtenue grâce à l'attention utilisant ces vecteurs.

La figure 2.10 présente le calcul de l'attention multi-têtes. Les 3 vecteurs, valeur (V), clé (K) et requête (Q) sont utilisés par le calcul du produit scalaire d'attention. Ce calcul est répété h fois avec h correspondant au nombre de têtes d'attention. Les têtes d'attention ont chacune des initialisations aléatoires donc différentes. Elles obtiennent des résultats du calcul des 3 vecteurs différents qui sont complémentaires. Ensemble, elles forment une représentation plus robuste car elles se sont focalisées sur des parties de phrases différentes.

Ainsi, l'exemple 2.11, présenté dans l'article, montre le maintien du contexte à travers plusieurs mots retrouvés plus loin dans la phrase. On voit notamment que l'attention est répartie entre plusieurs mots à travers la phrase, et que les différentes têtes d'attention (en différentes couleurs) se concentrent sur différents mots de la phrase.

Le décodeur fonctionne de façon similaire : il forme une représentation des mots générés aux étapes d'avant. Il récupère ensuite la sortie de l'encodeur avant de

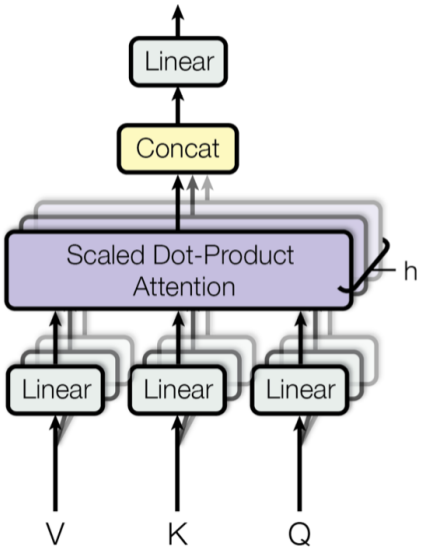

FIGURE 2.10 – Attention multi-têtes

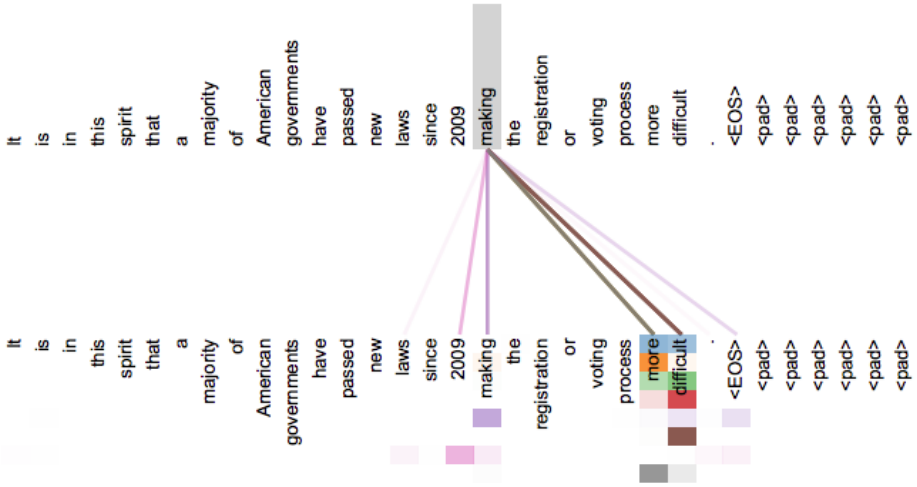

FIGURE 2.11 – Exemple d'attention suivant les dépendances à longue distance où les différentes couleurs représentent les différentes têtes de *multi-head* attention

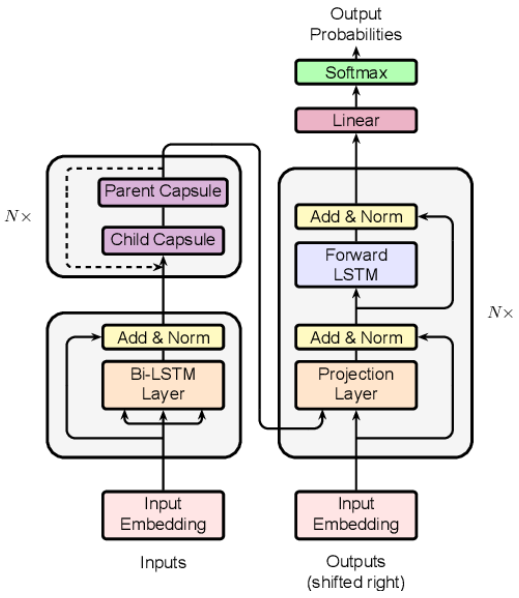

FIGURE 2.12 – Architecture réseau capsule pour la traduction automatique

repasser à nouveau par une couche d'attention multi-têtes. Enfin, le *softmax* produit la phrase cible la plus probable.

2.1.7 Réseaux de neurones capsule

L'approche proposée par Wang (2019) reprend le réseau de neurones basé sur le principe de capsule présenté par Sabour et al. (2017) pour la traduction automatique. L'idée est d'utiliser un mécanisme d'agrégation pour représenter la phrase source à l'aide d'une matrice de taille fixe. Ensuite, un réseau de LSTM décode la phrase à partir de cette représentation pour obtenir la phrase en langue cible. Cette approche, qui dérive des réseaux convolutionnels, supprime certaines connections et divise les couches en petits groupes de neurones. Elle vient répondre à une limite des réseaux convolutionnels qui répartissaient des concepts et des sous-concepts sans maintenir de cohérence en termes de position dans les objets de plus haut niveau. Par exemple, en traitement de l'image, un visage qui était décomposé dans les couches suivantes en œil, bouche, nez n'assurait pas la position de ces derniers dans le visage représenté précédemment.

L'architecture du réseau présentée dans la figure 2.12 [2] est une adaptation de l'architecture *transformers*. Ils intègrent deux couches de capsule, une parent et une enfant qui à travers plusieurs itérations apprennent à réduire la taille de l'information à encoder, la taille de la capsule parent étant plus petite que celle de l'enfant.

2. Schéma d'architecture provenant de Wang (2019)

Les auteurs montrent qu'ils obtiennent des performances similaires aux approches de type *transformers*.

2.1.8 Discussion sur les architectures neuronales

Nous avons décrit les évolutions des techniques d'apprentissage automatique autour des réseaux de neurones jusqu'à des architectures complexes telles que les *transformers* et les réseaux de neurones capsule. Les systèmes ont évolué ainsi que les unités les composant, passant de cellules récurrentes à des unités récurrentes à portes plus sophistiquées (GRU/LSTM). Les mécanismes d'attention ont servi de base pour des techniques plus complexes d'auto-attention multiple. Les systèmes utilisés en 2020 en traduction automatique sont les approches à base de *transformers*. Ils surpassent les performances des systèmes avec mécanisme d'attention classique. Les réseaux de neurones capsule ne sont pas représentés dans les récentes campagnes d'évaluation (Barrault et al., 2019) dominées par les *transformers*.

2.1.9 Initialisation

L'initialisation des poids d'un système neuronal est une des premières étapes avant le début de l'apprentissage. Les poids sont initialisés aléatoirement en utilisant une certaine distribution. La façon dont ils sont initialisés a un impact sur les performances finales du système et sur la vitesse à laquelle le système va converger. L'objectif est de faciliter l'activation du neurone pour qu'un petit changement de poids implique un grand changement de la valeur de sortie. Plusieurs algorithmes d'initialisation de ces poids existent (Glorot and Bengio, 2010; He et al., 2015).

L'initialisation Xavier (Glorot and Bengio, 2010) cherche à déterminer comment initialiser les poids en fonction du nombre de connexions en entrée et en sortie du neurone.

$$W \sim U\left[-\frac{\sqrt{6}}{\sqrt{n_j + n_{j+1}}}, \frac{\sqrt{6}}{\sqrt{n_j + n_{j+1}}}\right] \tag{2.4}$$

La formule 2.4 de l'algorithme présentée dans Glorot and Bengio (2010) détermine automatiquement l'initialisation des poids des couches W basée sur une distribution aléatoire uniforme U suivant le nombre de connexions entrantes (n_j) et sortantes (n_{j+1}).

L'initialisation présentée dans He et al. (2015) dérive de celle de Xavier en prenant en compte la propagation de l'information à travers de nombreuses couches et gère mieux les non-linéarités. Ils montrent dans un exemple, avec 22 couches et 30 couches, que là où l'initialisation Xavier ne permet pas au système de converger,

leur approche obtient de meilleurs résultats.

2.2 Vocabulaire, normalisation et longueur des phrases

Cette section décrit les différentes étapes de pré-traitements qui sont appliquées aux corpus en vue de leur utilisation pour l'apprentissage de systèmes automatiques. L'attention sera portée sur les corpus textuels pour la traduction automatique.

La taille de vocabulaire pour les systèmes de traduction automatique doit rester raisonnable avec moins d'une centaine de milliers de mots en moyenne et ainsi limiter le nombre de paramètres du système de traduction.

Plus celui-ci est grand, plus le calcul est coûteux. Cela est dû à la fonction softmax appliquée sur la couche de sortie du système pour calculer une distribution de probabilités sur l'ensemble des mots du vocabulaire. Plus ces valeurs sont nombreuses, plus le calcul est important.

La taille du vocabulaire a aussi un impact sur les dimensions des plongements de mots, étant donné qu'ils sont définis par N dimensions multipliées par le nombre d'entrées du vocabulaire. De plus, une architecture plus grande peut être problématique d'un point de vue matériel car elle nécessitera des machines avec plus de mémoire.

Des travaux ont été proposés pour conserver de grands vocabulaires tout en maintenant le temps de calcul raisonnable. Par exemple, lors de l'entrainement, Jean et al. (2015) proposent d'effectuer le softmax sur un sous-ensemble du vocabulaire correspondant à la liste des mots présents dans les phrases cibles du batch. Cela permet d'éviter un softmax coûteux sur l'ensemble du vocabulaire pour chaque mot à traduire lors de l'apprentissage.

2.2.1 Normalisation

Il existe plusieurs techniques pour réduire la taille du vocabulaire. La principale est la normalisation des corpus. La normalisation est une somme de règles de segmentation et de transformation des phrases. Une fois celle-ci définie, elle est appliquée aux corpus. Elle permet de formater et d'orthographier les mots de la même façon, sur l'ensemble du corpus.

Le formatage et l'orthographe des mots font apparaitre parfois un même mot sous de multiples représentations dans les corpus. Ils sont parfois orthographiés de façons différentes, avec ou sans accent, avec ou sans tiret pour les mots composés, avec ou sans apostrophe et avec ou sans espace etc. Chaque écriture possible devient une entrée de notre vocabulaire si laissée telle quelle. La normalisation permet

d'uniformiser leurs représentations sur l'ensemble du corpus et donc de réduire le nombre d'entrées du vocabulaire, qui sont appelées token.

Ex 2.2.1 *Exemple : « J'ai un problème de normalisation. »*
Phrase normalisée : « J' ai un problème de normalisation . »

Dans l'exemple 2.2.1, la normalisation de «j'ai» est sujette à débat. La plupart du temps, on choisi de normaliser en séparant l'apostrophe du mot qui la suit mais c'est une convention. Le point est séparé du dernier mot de la phrase afin d'éviter d'avoir une entrée du vocabulaire pour « normalisation » et « normalisation. ».

2.2.2 Pré-traitements en traduction automatique

Un autre pré-traitement communément utilisé est le filtrage des phrases trop longues ou trop courtes. En effet, pour l'apprentissage d'un système automatique de traduction, il est important de voir les mots dans leur contexte. La traduction d'une phrase composée de seulement quelques mots s'avère complexe. Les mots rares vus dans des contextes spécifiques seront difficiles à traduire avec seulement quelques mots les entourant. Les phrases trop courtes sont retirées des corpus. Le seuil est généralement fixé entre 3 et 5 mots minimum pour que la phrase soit conservée.

À l'inverse, les phrases trop longues sont aussi ignorées. Historiquement, avec les réseaux récurrents, les phrases trop longues étaient filtrées car le réseau et sa récurrence s'avéraient peu efficaces pour conserver un contexte sur un nombre de mots conséquents et donc n'arrivaient pas à modéliser des dépendances à longues distances dans la phrase. Le filtrage des phrases longues était adapté pour éviter ce cas de figure. Aujourd'hui, ces dépendances avec les mécanismes d'attention sont mieux gérées. Cependant, nous conservons ces filtrages. Bien que de mieux en mieux traités avec les nouvelles architectures, les vecteurs représentant des phrases longues consomment plus de mémoire. De très longues phrases, de plusieurs centaines de mots, peuvent impacter l'apprentissage négativement, du simple ralentissement jusqu'à l'arrêt complet pour défaut de mémoire.

2.2.3 Mot inconnu, caractères et sous-mots

La traduction de mots complets peut s'avérer complexe lorsque des représentations peu robustes de ces mots sont apprises par les systèmes automatiques. Ce manque de robustesse est le plus souvent lié à un manque de données ou de diversité de ces données. Ces mots mal appris polluent le système, car ils seront rarement

utilisés à bon escient. Le système aura des difficultés à les utiliser car ils n'apparaissent que dans peu de contextes différents. L'approche pour gérer ce phénomène est de remplacer les mots peu fréquents par le symbole «unk» représentant un mot inconnu. Le système se concentre alors sur les mots qu'il connait le mieux et évite de remplir le vocabulaire de mots peu utilisés.

Il est même possible de se passer de mots avec les systèmes fondés sur les caractères (Lee et al., 2017). Historiquement, les systèmes de traduction automatique fonctionnent avec des mots normalisés comme entrées du vocabulaire. Les expériences réalisées par Lee et al. (2016) remplacent les mots par des suites de caractères, et prédisent aussi en sortie des suites de caractères. Ce niveau de représentation est un défi pour le système qui doit composer des séquences de caractères représentant des mots, mais aussi gérer les espaces entre les mots et composer des phrases correctes. En revanche, la réduction de la taille du vocabulaire est drastique dans ces systèmes. Ils ne comprennent plus qu'une liste de caractères en minuscules et majuscules ainsi que la ponctuation et quelques caractères spéciaux. Les approches utilisant des représentations au niveau du caractère ont été d'abord expérimentées pour la tâche de modélisation du langage (Kim et al., 2016).

Une solution hybride existe entre les représentations de mots et celle de caractères : les sous-mots. Les approches à base de sous-mots conçoivent des vocabulaires optimisés comprenant généralement des mots, des caractères et des sous-mots. Les sous-mots présentent de nombreux avantages que nous présentons à travers deux des algorithmes les plus répandus que sont BPE et SPM.

2.2.3.1 Byte Pair Encoding

L'utilisation de sous-mots (Sennrich et al., 2016b) a offert une alternative pour la prise en compte des mots peu fréquents dans les systèmes de traduction. Le principe consiste à représenter les mots fréquents avec moins de symboles et les mots moins fréquents avec plus de symboles.

Les auteurs partent de la constatation suivante : les mots fréquents ont des représentations robustes lors de l'apprentissage alors que les mots peu fréquents obtiennent de moins bonnes représentations. Pour remédier à cela, l'algorithme tend à représenter ces mots peu fréquents en les décomposant en sous-mots qui occurrent plus souvent. Ils ont donc une meilleure représentation. Ainsi dans le vocabulaire, les mots peu fréquents apparaissent, après l'application de l'algorithme, sous la forme d'une suite de plus petits sous-mots.

L'algorithme, inspiré d'un algorithme de classification hiérarchique ascendant, fonctionne de la façon suivante : à l'initialisation, il décompose tout les mots du

corpus en caractères. Le vocabulaire est alors composé de l'alphabet de la langue et des différentes ponctuations. L'algorithme cherche le bigramme le plus fréquent. Puis il concatène ces deux symboles pour en créer un nouveau avant de l'ajouter au vocabulaire. L'algorithme répète cette opération d'assemblage jusqu'à atteindre le nombre maximum d'opérations d'assemblage défini en paramètre. Au cours des itérations, des sous-mots de plus en plus longs sont construits jusqu'à l'obtention, régulièrement, de mots complets. En fin d'algorithme, on obtient un vocabulaire de taille égale au nombre de concaténations et de symboles de départ. L'algorithme BPE permet de limiter la taille des vocabulaires tout en offrant une meilleure couverture du corpus par rapport à l'utilisation classique de mots complets.

Ex 2.2.2 *Exemple d'application de BPE :*
Phrase originale : un homme sur une tyrolienne pénètre dans l'eau .
Phrase avec BPE : un homme sur une ty@@ ro@@ li@@ enne p@@ én@@ è@@ tre dans l' eau .

Dans l'exemple 2.2.2, une suite de deux arobases représente un découpage en sous-mots. Les différents mots des corpus après normalisation (section 2.2) seront segmentés en fonction du vocabulaire construit par l'algorithme BPE. Les mots «tyrolienne» et «pénètre» sont découpés en sous-mots. La phrase est décomposée en mots et sous-mots séparés par des espaces.

Les mots ne sont recomposés qu'après la phase de décodage pour l'évaluation en assemblant les sous-mots en mots en enlevant les arobases.

2.2.3.2 SentencePiece SPM

Un autre algorithme a été développé par Kudo and Richardson (2018). Celui-ci permet, selon un nombre prédéfini, d'optimiser une taille de vocabulaire. L'idée est de réaliser des découpages en sous-mots afin d'obtenir la plus grande couverture du corpus possible avec un vocabulaire de taille donnée. L'algorithme calcule un vocabulaire optimal pour couvrir les données qui sera appliqué comme précédemment aux différents corpus. L'avantage de cet algorithme est qu'il prend en compte les espaces comme un caractère à part entière dans les mots et les sous-mots. Contrairement à l'algorithme BPE, il n'est plus nécessaire de normaliser les corpus au préalable.

Ex 2.2.3 *Voici un exemple d'application de SPM :*
Phrase originale : J'ai vu une fille avec un télescope.
Phrase avec SPM : _J'ai _vu _une _fille _avec _un _ té le s c o pe .

On peut voir dans l'exemple 2.2.3 les caractères spécifiques à SPM pour représenter un espace avec certains token(__). La plupart des mots restent intacts à l'exception du mot « télescope ». Dans cet exemple, il y a une entrée dans le vocabulaire pour chaque mot complet avec l'espace représenté, par exemple « __J'ai » est une entrée du vocabulaire. Le mot « télescope » est découpé en plusieurs sous-mots et caractères. Ce mot a dû apparaitre rarement dans le corpus et l'algorithme à décidé de le découper en sous-mots. De cette façon, l'algorithme assure une couverture maximale du corpus en préservant la taille de vocabulaire. L'approche est à l'opposé de celle de BPE qui assemble des sous-mots pour former des sous-mots et des mots fréquents. SPM cherche à découper les mots peu fréquents pour que les sous-mots les composant soient fréquents. SPM s'apparente à un algorithme de classification hiérarchique descendant.

Les approches à base de sous-mots permettent la génération de mots qui n'apparaissaient pas dans le corpus d'entraînement. Un mot tel que « télescopique » qui n'aurait, par exemple, pas été vu dans le corpus d'entraînement, pourrait être généré en traduction grâce à l'utilisation de sous-mots. Les systèmes à base de sous-mots représentent, d'une certaine façon, une approche hybride où des mots, des sous-mots et des caractères cohabitent dans le vocabulaire. Des recherches ont été effectuées sur ce que capturent ces différentes approches de manière séparée ou combinée (Durrani et al., 2019). En traduction automatique, les auteurs montrent que les systèmes à base de sous-mots sont les plus performants.

2.2.4 Discussion sur les vocabulaires et les pré-traitements

Qu'est-ce qu'un mot pour les systèmes automatiques ? En traduction, c'est une entrée d'un vocabulaire qui est propre à un corpus de données. Les entrées d'un vocabulaire peuvent être un mot, un sous-mot, un caractère, un chiffre, etc. Cependant, dans les systèmes de traduction automatique, les entrées du vocabulaires sont ensuite remplacées par un nombre entier qui les représente. La transformation en entier des entrées du vocabulaire explique mieux pourquoi l'utilisation de sous-mots fait sens. Pour le système, une phrase est une suite de nombres. Durant l'apprentissage, peu importe que cette suite représente des mots ou des sous-mots, le système cherche à apprendre à partir de cette suite de nombres pour prédire la meilleure séquence étant donnée l'entrée.

Les architectures de traduction automatique sont le plus souvent des adaptations des approches les plus performantes développées dans le domaine de recherche de l'apprentissage automatique, un domaine en plein essor avec le succès des réseaux de neurones.

CHAPITRE 3

APPRENTISSAGE PAR TRANSFERT

Par essence, les systèmes neuronaux ont besoin d'une grande quantité de données pour être appris. Mais, dans le cas où peu de données sont disponibles, les systèmes fournissent rarement de bonnes performances.

Comment qualifier une quantité de données ? La frontière est floue et varie selon les tâches d'apprentissage automatique. Dans le cas de la traduction automatique, la frontière est difficile à situer. Le cas le plus extrême est celui des paires de langues où nous ne disposons pas de données parallèles. Alors, il est impossible d'apprendre directement un système de traduction automatique classique. Généralement, pour beaucoup de paires de langues nous disposons de quelques milliers de phrases parallèles, jusqu'à parfois plusieurs millions. Ces quantités ne permettent pas toujours d'apprendre un système de traduction automatique performant.

L'apprentissage par transfert est une méthode souvent employée lorsque peu de données sont disponibles. Le concept de l'apprentissage par transfert repose sur l'idée qu'un modèle déjà appris peut être utilisé pour initialiser un modèle à la place des algorithmes classiques (Glorot and Bengio, 2010; He et al., 2015). Les poids d'un réseau de neurones précédemment appris sont alors utilisés pour initialiser le nouveau modèle. Le transfert des poids vers un nouveau modèle permet d'utiliser une base robuste reposant sur des connaissances antérieures plutôt qu'une initialisation aléatoire (tel que présenté dans la section 2.1.9).

Dans ce chapitre, les principes liés au transfert seront présentés dans la première section. La seconde section traitera du transfert séquentiel couramment utilisé en apprentissage automatique. Une troisième section s'intéressera au transfert multilingue inhérent à l'apprentissage d'un système de traduction automatique multilingue. Enfin, une dernière section fera un bilan sur le transfert.

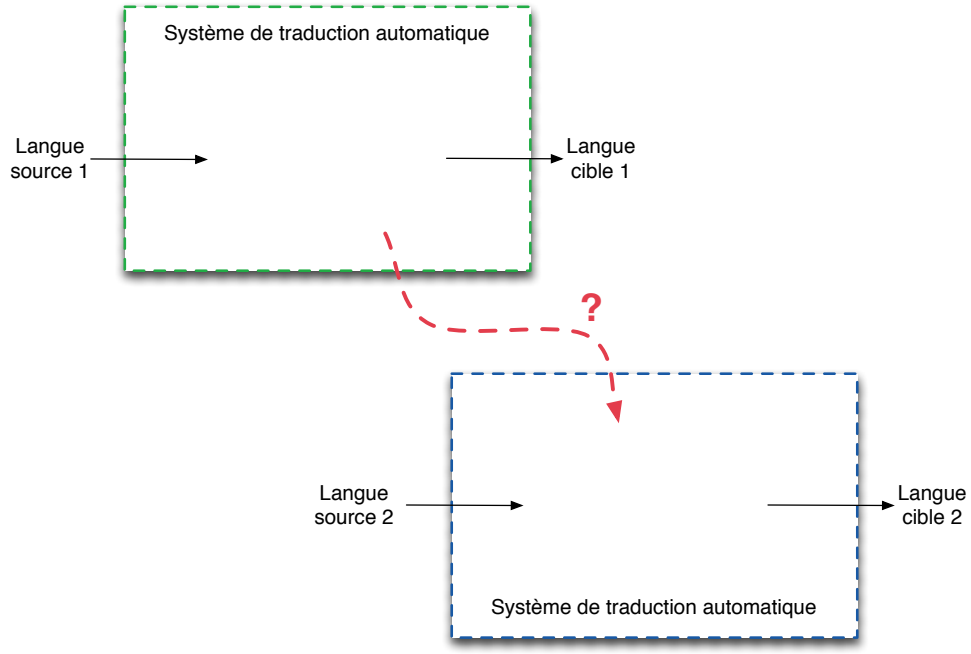

FIGURE 3.1 – Exemple de recherche d'un transfert entre deux systèmes de traduction. Quelle information doit être transférée et comment ?

3.1 Principes du transfert

Les premiers travaux sur la réutilisation d'informations d'un réseau de neurones pour aider l'apprentissage d'un second réseau datent du début des années 90 avec les travaux de Pratt et al. (1991). Plutôt que d'ajouter de nouvelles données d'apprentissage, ils cherchent à transférer de l'information d'un réseau vers un autre. La figure 3.1 illustre cette recherche d'un transfert entre deux systèmes de traduction qui pose plusieurs questions.

La proposition de Pratt et al. (1991) est un transfert des poids d'un perceptron multicouche vers un nouveau perceptron de même forme. Cette méthode est utilisée pour apprendre un classifieur adapté à une nouvelle tâche. Repris dans Pratt (1993), les auteurs montrent que le transfert permet d'apprendre plus rapidement un réseau de neurones qu'avec une initialisation aléatoire.

Les architectures de réseaux de neurones profonds, composées de nombreuses couches, capturent à différents niveaux de profondeurs des représentations des don-

nées différentes. Les représentations intermédiaires sont intéressantes car elles ne sont pas spécialisées sur une tâche ou sur des données (Bengio et al., 2011). Les couches supérieures du réseau (les plus éloignées de la couche d'embedding) capturent des concepts plus généraux (Bengio, 2012; Conneau et al., 2017). Les spécificités de ces couches sont intéressantes pour le transfert. Le transfert a été particulièrement exploré en traitement de l'image (Zhou et al., 2019; Yosinski et al., 2014) avec par exemple de la reconnaissance d'objet. Ces modèles sont composés d'un grand nombre de couches capturant chacune une information différente : les premières agissent comme de simples détecteurs de contours, et plus on avance vers des couches profondes, plus celles-ci capturent des concepts sémantiques complexes (Zhou et al., 2019).

Les termes système « parent » et système « enfant » ont été introduits par Zoph et al. (2016) pour distinguer les deux systèmes utilisés lors du transfert. Le système parent correspond au système dont on réutilise l'apprentissage pour le système à apprendre, nommé quant à lui système enfant. Le système parent est le système qui servira de base au système enfant qui l'utilise comme point de départ pour son apprentissage.

Le principe du transfert est largement utilisé en traitement automatique de la langue. Les plongements de mots sont souvent extraits pour être utilisés comme base pour un système enfant. Par exemple, l'approche utilisée par ELMo (Peters et al., 2018) est d'apprendre des plongements de mots à l'aide d'un réseau pour la modélisation du langage pour être réutilisés pour des tâches de classification de texte, de traduction automatique ou encore de reconnaissance d'entités nommées. Plus récemment, une grande partie des travaux s'intéresse à la réutilisation de modèle de langue fondé sur les *transformers* (BERT, (Devlin et al., 2019)) afin de le spécialiser pour une autre tâche (Rogers et al., 2020).

Si le principe du transfert est simple, il peut être réalisé de nombreuses façons. Pan and Yang (2010) synthétisent sous forme de trois questions les éléments à considérer avant de mettre en œuvre une méthode d'apprentissage par transfert.

1. **« Que transférer ? »**
   Les modèles neuronaux considérés dans cette thèse sont composés de nombreuses matrices de poids qui sont apprises durant l'entraînement (dans l'encodeur et le décodeur et le ou les mécanismes d'attention). Il faut donc se poser la question de savoir quels sont les paramètres pertinents à transférer du parent vers l'enfant.

2. **« Comment transférer ? »**
   Comment réutiliser l'information acquise par le parent et comment la trans-

mettre à l'enfant ? En traduction automatique, nous verrons que l'initialisation d'un système est un moment propice à l'utilisation de poids précédemment appris.

3. **«Quand transférer ? »**
   Le transfert n'est pas toujours bénéfique. Selon le cadre, on observe parfois un transfert négatif (Wang et al., 2019b). On appelle transfert négatif le fait qu'un système enfant offre des performances moindres qu'un système entraîné sur les mêmes données sans transfert. Il est donc important d'identifier les conditions nécessaires à un bon transfert.

Deux approches de transfert sont largement exploitées en traduction automatique : le transfert séquentiel et le transfert multilingue. Ces deux sujets sont présentés dans les deux prochaines sections.

3.2 Transfert séquentiel

Le transfert dit « séquentiel » est l'approche classique de transfert. Elle consiste à réutiliser toute ou partie des poids d'un modèle parent entraîné sur une tâche parent afin d'initialiser les poids du modèle enfant. L'approche est souvent utilisée en traduction automatique lorsque peu de données sont disponibles pour apprendre une paire de langues. Le principe consiste à réutiliser les poids d'un modèle parent appris sur de grandes quantités de données. L'effet escompté est que les représentations apprises sur la paire de langues parent permettent d'améliorer les performances sur la paire de langues enfant.

La figure 3.2 illustre ce principe : un premier système de traduction automatique est une source de transfert pour un autre système de traduction. L'implication de langues communes aux différents systèmes est un moyen simple de favoriser le transfert entre eux.

Dans le cadre du transfert séquentiel, les questions du transfert « que transférer ? » et « comment transférer ? » en traduction automatique ne sont pas différentes du cas général. En revanche, la question « quand transférer ? » fait particulièrement débat avec plusieurs critères qui entrent en considération tels que la proximité des langues employées et l'importance des quantités de données disponibles.

Nous allons voir dans la prochaine section que plusieurs techniques sont possibles pour le transfert séquentiel en traduction automatique.

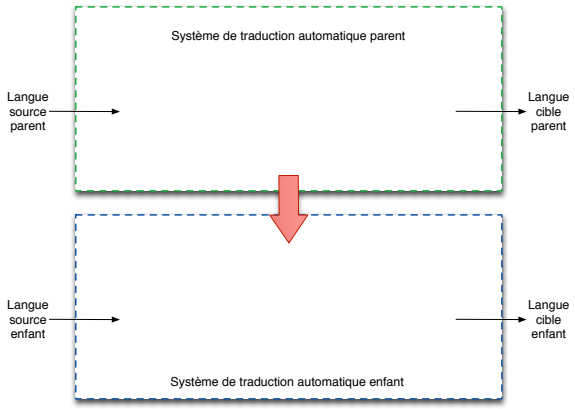

FIGURE 3.2 – Exemple de Transfert Séquentiel en Traduction Automatique Neuronale

3.2.1 Techniques de transfert séquentiel

L'utilisation d'approche de transfert séquentiel en traduction automatique correspond le plus souvent à réutiliser des poids du système parent comme initialisation du système enfant (Zoph et al., 2016; Kocmi and Bojar, 2018; Dabre et al., 2017).

Zoph et al. (2016) font partie des premiers à utiliser une approche de transfert séquentiel en traduction automatique neuronale. L'architecture proposée est composée d'un encodeur et d'un décodeur avec mécanisme d'attention pour une paire de langues enfant espagnol-anglais utilisant 2.5M de phrases parallèles. Pour les systèmes parents, ils comparent un parent français-anglais et un parent allemand-anglais disposant tout deux de 53M de phrases parallèles comme corpus d'entraînement. Ils utilisent l'ensemble des poids appris par le parent pour l'apprentissage de l'enfant. Cependant, des modifications sont apportées sur le processus d'apprentissage pour obtenir une amélioration des performances sur le système enfant. Les plongements de mots (embeddings) sont conservés pour le côté source et cible. La langue cible étant la même, ces embeddings vont améliorer les performances du décodeur. En revanche, côté source, les embeddings sont conservés alors que la langue change, ce qui s'avère probablement plus intéressant qu'une initialisation aléatoire. Ils proposent de geler des parties du réseau enfant de plus en plus grandes pour réduire la quantité de poids à réapprendre et montrent une amélioration des performances alors qu'une grande partie du réseau est gelé à l'exception des embeddings. Ils concluent que pour des paires de langues proches, geler une partie des poids est bénéfique pour le transfert. Leurs expériences montrent une amélioration liée au transfert de plus de 15 points BLEU en passant de 16.4 sans transfert à 29.8

en utilisant le parent allemand-anglais et 31.0 avec le parent français-anglais. La conclusion est que le meilleur transfert pour l'espagnol provient du français qui est plus proche que l'allemand. Cela soulève la question de l'importance de la proximité des langues pour le transfert.

Kocmi and Bojar (2018) présentent une approche de transfert nommée **transfert trivial**. Le concept est de ne pas faire évoluer l'architecture lors du passage du parent à l'enfant. L'architecture du modèle ne change pas et les poids du réseau de neurones sont transmis directement à l'enfant. En terme d'apprentissage, cela se traduit simplement par un apprentissage complet du système parent classique. Ensuite, les poids du modèle parent sont utilisés pour initialiser les poids du système enfant. Le système parent et enfant ont la même architecture et les mêmes dimensions. Kocmi and Bojar (2018) utilisent un vocabulaire partagé comprenant les différentes langues employées. L'approche de Kocmi and Bojar (2018) contraint à ne pas faire évoluer l'architecture lors du passage du parent à l'enfant. Pour cela, les vocabulaires ne doivent pas changer entre les deux systèmes. Ainsi, la réalisation du vocabulaire doit être anticipée car il comprend les mots qui seront dans le système parent et dans le système enfant qui sera appris ensuite. Le vocabulaire comprend donc plusieurs langues. C'est sur les pré-traitements que l'approche diffère d'un système classique.

Les auteurs présentent une architecture à base de *transformers* pour la traduction automatique. Ils utilisent de nombreuses paires de langues incluant le finnois, l'estonien, le russe et le slovaque pour tester différents transferts entre différentes langues. L'anglais est toujours utilisé soit en langue source ou en langue cible. Leurs expériences montrent un transfert positif dans presque toutes les paires de langues, même pour des systèmes enfants qui obtenaient déjà de bonnes performances sans transfert. Les auteurs montrent qu'un transfert significatif se produit également lorsque les langues des systèmes parents et enfants ne partagent par le même alphabet (tel que le russe et l'estonien) ce qui réduit fortement le partage entre les langues. Leur déduction est que la proximité des langues n'est pas le critère le plus important et que les quantités de données des systèmes parents est un critère primordial pour un transfert performant.

Thompson et al. (2018) montrent que geler certaines parties d'un système neuronal avant de continuer l'apprentissage a un faible impact sur les performances finales, tout en accélérant le processus. Ils montrent également que de geler tout, sauf un composant du réseau de neurones (par exemple l'encodeur, le décodeur ou encore les embeddings source ou cible), offre des performances similaires à un réseau complet sur lequel l'apprentissage est poursuivi. Les auteurs observent que l'adaptation d'un seul composant lors d'une continuation d'apprentissage offre de

bonnes performances, obtenant des résultats similaires à un apprentissage complet.

Nguyen and Chiang (2017) utilisent la translittération pour associer des plongements de mots avec des mots de sens proche, mais d'orthographes différentes. Leur approche vise à maximiser le potentiel de transfert entre parent et enfant. Cependant, ils se placent dans un cadre où les systèmes parents envisagés sont eux aussi peu dotés, et montrent que le fait de geler les plongements de mots peut mener à de moins bonnes performances chez l'enfant.

Nous avons pu voir à travers ces différentes approches qu'il y a plusieurs techniques de transfert possibles avec des avantages et inconvénients différents. L'utilisation de sous-mots, appris entre les différentes langues utilisées, semble être un levier efficace pour favoriser le transfert entre le parent et l'enfant. Il a été montré que geler certains poids du réseau enfant peut avoir un impact minime sur ses performances tout en accélérant l'apprentissage. Cette technique s'est aussi révélée performante pour un transfert impliquant des langues proches.

3.2.2 Débat sur les critères importants du transfert

Dabre et al. (2017) proposent une étude sur les langues utilisées pour le transfert. Les auteurs comparent plusieurs langues pour un système parent et s'intéresse, plus particulièrement, au lien entre la proximité des langues utilisées et les performances du transfert. En effet, leurs expériences montrent qu'il y a un lien entre la proximité des langues et les performances du transfert. Pour cela, ils s'intéressent à 6 groupes de langues différentes regroupant un total de 16 langues avec par exemple des langues européennes telles quel le français, l'allemand et le luxembourgeois mais aussi des langues indo-aryennes telles que le l'hindi ou le pendjabi. Suite à leurs expériences, ils concluent qu'une forte proximité des langues mène à un meilleur transfert.

Cependant, les expériences de Kocmi and Bojar (2018) montrent que la proximité n'est pas primordiale pour un bon transfert et ils obtiennent même de meilleurs résultats avec des paires très éloignées et de bons résultats avec des paires qui ne partagent pas le même vocabulaire. Ceci contredit les conclusions de Dabre et al. (2017) et paraît plus éloigné de l'intuition générale décrite précédemment.

Kocmi and Bojar (2019) affirment que les langues utilisées pour le transfert ne sont pas particulièrement importantes. Selon eux, le critère le plus important est la taille du corpus d'entraînement du système parent. Les auteurs vont plus loin, en proposant d'utiliser l'apprentissage par transfert pour l'apprentissage de chaque système de traduction à l'avenir. L'approche qu'ils décrivent est d'utiliser un système précédemment appris comme système parent pour l'apprentissage de

chaque système de traduction. De cette façon, ils montrent des résultats supérieurs à ceux obtenus avec une initialisation aléatoire. Le vocabulaire du système enfant pouvant être très différent de celui du parent employé, ils proposent un algorithme pour faire évoluer le vocabulaire dynamiquement. L'algorithme remplace les entrées du vocabulaire provenant du système parent au cours de l'apprentissage du système enfant. Les nouvelles entrées sont des mots récurrents dans le corpus du système enfant. Cela permet de conserver les entrées du vocabulaire provenant du parent qui sont utiles pour l'enfant, et de remplacer celles qui ne le sont pas par des mots propres au corpus du système enfant. D'un point de vue quantitatif, cette approche ne semble pas toujours être la meilleure car ses performances ne surpassent pas, en moyenne, celles de leurs travaux précédents (Kocmi and Bojar, 2018). En revanche, d'un point de vue vitesse d'apprentissage et donc de temps de convergence du modèle, cette nouvelle approche est bien plus rapide.

Ce débat sur le transfert à travers les critères de proximité des langues et des quantités de données disponibles pour les paires de langues fera l'objet d'une étude dans le chapitre 5 de contributions.

3.3 Transfert multilingue

En traduction automatique un système unique peut être utilisé pour apprendre plusieurs paires de langues, c'est ce que nous appelons un système multilingue. Le transfert multilingue tel que nous le présentons est le transfert inhérent à l'apprentissage d'un système multilingue. Les approches multilingues visent à mutualiser plusieurs parties d'un réseau de neurones, il en résulte un transfert entre les langues au sein même du système.

Pour le transfert multilingue, les trois questions du transfert prennent un sens différent ici avec l'utilisation d'une architecture unique et un partage direct entre les paires de langues quelque soient les quantités de données dont elles disposent.

La question « que transférer ? » pour une approche de système de traduction multilingue a un sens particulier. En effet, le transfert au sein d'un système multilingue passe par la mise en commun des différents éléments de l'architecture pour les différentes langues. Ce sont donc tous les paramètres qui sont partagés/transférés (à l'exception des embeddings de mots qui n'apparaissent que dans une langue). Le transfert multilingue se distingue du transfert séquentiel par ce partage global et surtout l'aspect « simultané » de l'apprentissage. La principale différence étant que pour le modèle multilingue, l'apprentissage se fait simultanément sur l'ensemble des langues considérées alors que pour le transfert séquentiel, on apprend d'abord le parent puis l'enfant. En transfert multilingue, le transfert se produit pendant

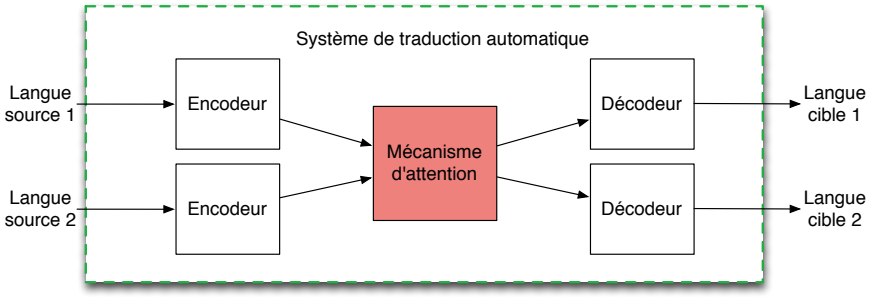

FIGURE 3.3 – Architecture à multiples encodeurs et décodeurs pour la traduction automatique multilingue

l'apprentissage des différentes paires de langues qui s'impactent les unes les autres.

La question « comment transférer ? » fait référence aux techniques de transfert multilingue. La section 3.3.1 présente les différentes approches possibles pour le transfert multilingue. Elles se basent principalement sur une adaptation de l'architecture.

Les travaux de recherche sur les approches multilingues utilisent généralement de nombreuses paires de langues et partent du principe que plus de langues sont disponibles, meilleur sera le système d'un point de vue performances générales. La question « quand transférer ? » est rarement évoquée car les travaux tendent à s'intéresser seulement au résultat du transfert sans rechercher les causes de celui-ci.

3.3.1 Techniques de transfert multilingue

L'intégralité du système de traduction porte le transfert entre les langues. C'est donc sur la conception de l'architecture et des pré-traitements que les techniques de transfert multilingues vont s'orienter.

Firat et al. (2016) proposent une approche de traduction automatique multilingue. L'architecture proposée comporte un encodeur pour chaque langue source et un décodeur pour chaque langue cible. Cela permet au système de voir son nombre de paramètres évoluer linéairement en fonction du nombre de paires de langues qui sont utilisées. Néanmoins, cela nécessite l'utilisation d'un mécanisme d'attention partagé entre les différentes paires de langues. La figure 3.3 illustre cette approche où une architecture de traduction multilingue est composée d'un encodeur et d'un décodeur par langue utilisée dans le système. L'utilisation d'encodeurs et de décodeurs séparés permet de rester proche du fonctionnement de systèmes mono-paires

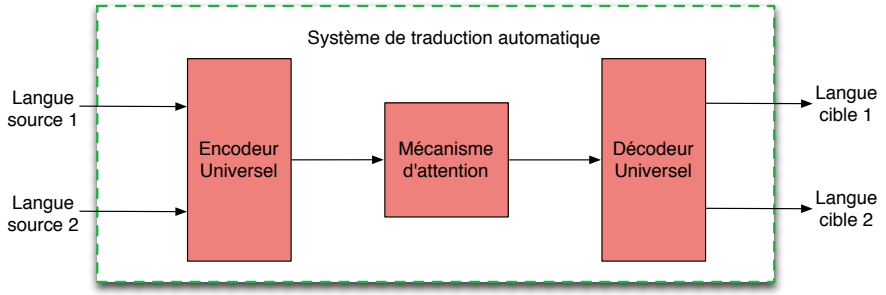

FIGURE 3.4 – Architecture universelle pour la traduction automatique multilingue

classiques. Seul le mécanisme d'attention est partagé (en rouge), il porte le transfert entre les langues dans cette configuration.

Une autre approche envisagée est celle d'encodeur et décodeur universels (Ha et al., 2016). Elle repose sur un encodeur et/ou un décodeur unique capables de traiter l'ensemble des langues considérées dans le système. La figure 3.4 présente une architecture de traduction automatique neuronale multilingue avec un encodeur et un décodeur universels.

Le système de traduction automatique multilingue proposé par Johnson et al. (2017) présente une architecture de très grande taille. Dans celle-ci, un encodeur universel composé de 8 couches encode les différentes langues d'entrée du système. Puis, le mécanisme d'attention et le décodeur lui aussi universel, composé de 8 couches récurrentes (les deux premières sont bidirectionelles), décodent vers les langues cibles. Leur approche permet de répartir les 8 couches de l'encodeur et du décodeur sur autant de machines différentes, accélérant la vitesse d'apprentissage et d'utilisation. L'utilisation d'encodeur et de décodeur universels pousse à un fort partage entre les langues car les vocabulaires sont multilingues. Ils présentent aussi une approche originale de traduction automatique de paires de langues où aucune donnée d'apprentissage n'est utilisée pour apprendre la paire traduite : ils nomment cette approche *zero shot learning* que nous traduisons par traduction entre des paires de langues non rencontrées durant l'apprentissage.

Cela est possible grâce à l'apprentissage d'un système multilingue. Ils font une étude sur les paires de langues n'ayant pas été entraînées conjointement. Ils essaient de traduire du portugais vers l'espagnol alors qu'ils n'ont pas de données pour cette paire de langues, elle n'a donc pas été vue lors de l'apprentissage. En revanche l'encodeur portugais a été entraîné par une paire de langues portugais-anglais et le

décodeur espagnol a été entraîné par une paire de langues anglais-espagnol. L'approche multilingue de Johnson et al. (2017) indique au système, lorsqu'on lui fournit une phrase source, la langue dans laquelle ont veut que la phrase soit traduite. Pour cela, ils choisissent d'utiliser un token spécifique en début de phrase qui conditionne le système à traduire dans une langue spécifique. Ils auraient pu faire le choix de forcer la langue cible en restreignant le softmax sur une sous-partie des unités qui correspondent à la langue de sortie désirée. Cependant, ils ont préféré utiliser un token particulier qui correspond à une langue spécifique pour que le système infère la langue cible désirée pour la phrase courante à traduire.

Ils présentent un exemple avec le token "<2es>" qui est ajouté en début de phrase source pour indiquer au système la langue de sortie attendue. Dans l'exemple, c'est donc l'espagnol qui est attendu et le décodeur a appris, d'une façon comparable à celle d'un modèle de langage, que les mots qui devront être utilisés pour la traduction en phrase cible devront être espagnols. Ce token peut être employé pour demander une traduction d'une phrase source vers une langue cible où la paire de langues n'a pas été apprise par le système. L'encodeur est entraîné à encoder cette langue source, et le décodeur est entraîné à décoder vers la langue cible, mais cette paire de langues n'a pas fait l'objet d'un apprentissage du système : aucune donnée parallèle d'apprentissage correspondant à cette paire n'a été utilisée. Cette approche est particulièrement intéressante, car très souvent il s'avère difficile de trouver des données pour toutes les paires de langues désirées. Cependant, pour ces paires de langues, où le système n'a pas vu de données parallèles, les performances sont mauvaises. En revanche, à l'aide d'une petite quantité de données parallèles ont obtient rapidement des résultats corrects.

3.3.2 Équilibrage des langues et transfert négatif

L'apprentissage de systèmes multilingues passe par la mise en commun dans un système unique de multiples paires de langues. Ces différentes paires de langues ont des quantités de données propres qui varient les unes des autres. Les approches multilingues explorées dans la communauté exploitent généralement de nombreuses paires de langues apprises. Les problèmes liés à l'équilibrage des langues et du transfert négatif sont rarement évoqués.

Les systèmes de traduction automatique neuronaux multilingues sont une solution pour l'apprentissage de paires de langues peu dotées. Le transfert inhérent à l'apprentissage de systèmes multilingues peut faire profiter, à une paire de langues peu dotée, l'apprentissage de paires de langues mieux dotées. En revanche, l'aspect simultané peut s'avérer problématique car avec des quantités de données différentes

entre les paires, le cycle d'apprentissage doit être adapté pour maintenir un équilibre entre les langues.

Aharoni et al. (2019) démontrent qu'il est possible de traduire plus de 100 paires de langues avec un système unique. L'apprentissage parallèle de toutes ces paires de langues a notamment un effet bénéfique sur les paires les moins dotées. En revanche, les performances de paires de langues les mieux dotées s'en voient diminuées. Des approches (Zaremoodi et al., 2018; Yang et al., 2019) essaient de contrer ce transfert négatif pour les paires mieux dotées avec notamment un agrandissement du système neuronal en termes de nombre de paramètres.

Des architectures toujours plus imposantes sont expérimentées. Elles sont principalement utiles pour supporter de grandes quantités de paires de langues apprises en parallèle, mais aussi pour exploiter au mieux les grandes quantités de données utilisables. La figure 3.5 montre les scores BLEU moyens obtenus avec leur approche [1].

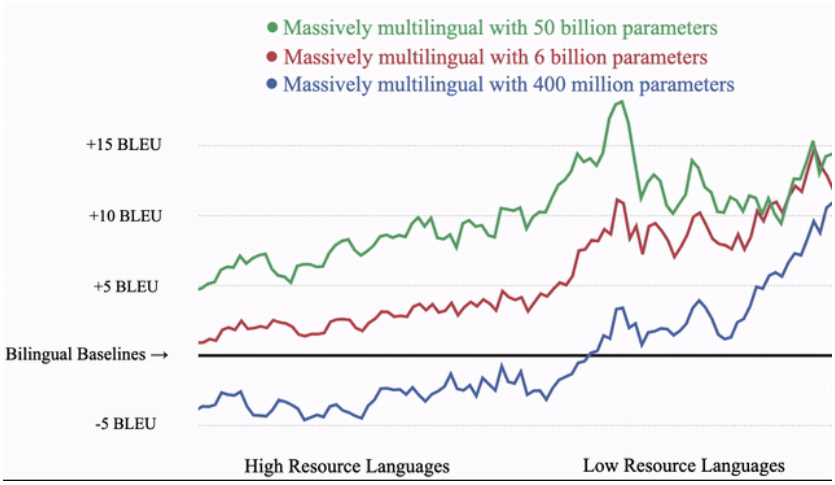

FIGURE 3.5 – Amélioration moyenne en BLEU en Traduction Automatique Neuronale Multilingue selon la taille de l'architecture. Figure extraite du blog de Google AI.

Ils obtiennent une amélioration moyenne de 5 points de BLEU sur les différentes paires de langues avec une architecture *transformers* de 128 couches avec plus de 6 milliards de paramètres.

Cela soulève plusieurs questions quant à l'équilibrage des langues. Nous nous intéresserons aux questions sur l'équilibrage des données des systèmes multilingues et au transfert négatif observé sur les paires mieux dotées dans nos contributions.

1. https ://ai.googleblog.com/2019/10/exploring-massively-multilingual.html

3.4 Bilan et discussions sur le transfert

Nous avons pu voir les différents concepts liés au transfert ainsi que certaines mises en applications. Les avantages du transfert sont clairs pour les systèmes neuronaux où les quantités de données jouent un rôle prédominant dans les performances finales des systèmes. Les quantités de données, lorsqu'elles sont insuffisantes, sont un obstacle qui peut être surmonté en exploitant des données d'une autre tâche et en transférant les connaissances des systèmes appris.

Dans ce chapitre nous nous sommes intéressés aux transferts séquentiel et multilingue en traduction automatique.

Le transfert séquentiel permet de réutiliser les poids d'un système parent comme base pour l'apprentissage d'un système enfant. Nous avons vu que cette approche est particulièrement employée et efficace lorsque peu de données sont disponibles pour la paire de langues enfant. Les techniques d'adaptation varient selon les approches, et peuvent s'appliquer sur certains poids seulement (embeddings/plongements) ou sur l'ensemble des poids du réseau parent. Qu'elles soient à base de poids du réseau gelées ou d'adaptation dynamique du vocabulaire, ces modifications sont propices à l'amélioration des performances des systèmes enfants.

Nous avons également vu plusieurs approches de transfert multilingue. Le transfert s'opère à travers l'apprentissage conjoint de multiples paires de langues au sein d'une même architecture. Les dimensions de ces architectures sont de plus en plus importantes avec les années jusqu'à traduire en 2020 plus d'une centaines de paires de langues simultanément. Nous avons vu que les questions du transfert ont un sens particulier avec ce genre d'approche. Les questions « que transférer ? » et « comment transférer ? » sont des sujets peu abordés dans la communauté qui se focalise sur les techniques de transfert. Les mots communs aux différentes langues sont un levier intéressant de partage au sein de ce type d'architecture.

De façon plus générale, le transfert en traduction automatique s'effectue par le partage de mots, et plus encore de sous-mots, entre les langues. L'analyse de ce partage est un axe d'étude intéressant qui sera abordé dans le chapitre 6 de contributions. Les sous-mots ont l'avantage d'être plus simples à retrouver dans plusieurs langues. Ils permettent également d'utiliser un algorithme d'apprentissage de sous-mots entre les différentes langues des systèmes parents et enfants. Cela permet d'obtenir des sous-mots optimisés pour représenter de multiples langues employées dans les systèmes et devrait favoriser encore plus le transfert.

Dans mes contributions, je m'intéresserai particulièrement au transfert en traduction automatique et à des approches combinant le transfert séquentiel et le transfert multilingue.

Deuxième partie

Contributions

CHAPITRE 4

TRANSFERT MULTILINGUE EN TRADUCTION AUTOMATIQUE

4.1 Introduction

Ce chapitre regroupe les contributions de cette thèse axées sur l'apprentissage par transfert multilingue. Nous nous intéressons particulièrement à l'apprentissage de systèmes de traduction automatique neuronaux pour des paires de langues peu dotées. Les approches neuronales actuelles, quoique très performantes, requièrent de grandes quantités de données d'entraînement. Cela pose évidemment un problème lorsque la ou les paires de langues mises en jeu ne sont pas dotées d'autant de données. Rendre les systèmes performants sur ces paires de langues peu dotées figure parmi les principaux challenges en traduction automatique neuronale (Koehn and Knowles, 2017).

Pour améliorer les performances de systèmes appris sur des paires de langues peu dotées nous proposons d'utiliser des approches d'apprentissage par transfert. Nous observons que le transfert multilingue tend à réduire les performances des paires de langues bien dotées. La principale problématique adressée est donc la suivante : comment faire en sorte que le transfert se fasse sans pénaliser les paires de langues bien dotées ?

Nous définissons le transfert multilingue comme le transfert induit par l'apprentissage de multiples langues dans un système unique. Nous nous plaçons uniquement dans des cadres où un système de traduction unique traduit plusieurs paires de langues. L'objectif d'un tel système est que les paires de langues impliquées qui possèdent une grande quantité de données vont permettre de fixer les paramètres de notre système et ainsi en faire bénéficier les paires de langues moins bien dotées.

Nous allons chercher à obtenir un système multilingue performant en testant plusieurs configurations différentes de systèmes notamment autour des pré-traitements des données. Nous verrons qu'un transfert négatif s'opère sur les paires de langues les mieux dotées et nous proposerons une approche pour réduire cette perte de performances.

Firat et al. (2016) sont à l'origine de l'une des premières publications de traduction automatique multilingue neuronale. Cet article et le toolkit associé sont disponibles en ligne et serviront de base à de nombreux travaux dont les nôtres. L'approche proposée par Firat et al. (2016) est celle d'un système multi-encodeurs et multi-décodeurs avec un mécanisme d'attention partagé pour la traduction automatique telle que décrite dans la section 3.3.1.

4.2 Architecture neuronale

L'architecture que nous utilisons est un encodeur bi-RNN (tel que présenté dans la section 2.1.4) avec mécanisme d'attention de type perceptron multi-couche à une couche cachée avec un décodeur de type GRU conditionnelle. Dans cette architecture un encodeur est propre à chaque langue source et un décodeur est propre à chaque langue cible. Nous utilisons une taille d'état caché des encodeurs et des décodeurs de 1000 et 620 dimensions pour les plongements de mots. La couche cachée du mécanisme d'attention est de dimension 1200. La dimension du mécanisme d'attention est particulièrement grande car celui-ci est partagé entre les différents encodeurs et décodeurs ; il a donc un rôle-clé car il sera utilisé par chacun d'entre eux. L'algorithme d'optimisation utilisé est Adam (Kingma and Ba, 2015). Le taux d'apprentissage utilisé est de 0.0002 et la taille de batch de 60.

Notre expérience multilingue suit l'architecture multi-encodeurs et multi-décodeurs proposée par Firat et al. (2016).

4.3 Choix des données

Nous avons utilisé les données proposées pour la campagne d'évaluation WMT2016.

Nous utilisons les données des paires de langues Allemand-Anglais (DE-EN) et de la paire de langues Anglais-Turc (EN-TR) car elles offrent un parfait exemple de différence de quantités de données. La paire de langues DE-EN est fortement dotée avec 40M de phrases parallèles et la paire de langues EN-TR est faiblement dotée (200k phrases après pré-traitements). Nous nous attendons à ce que cette faible quantité de données pour la paire EN-TR améliore les performances du système sur

cette paire grâce à un apprentissage multilingue aux côtés d'une autre paire mieux dotée.

Nous avons utilisé des pré-traitements tels que présentés dans la section 2.2. Tout d'abord, nous appliquons une normalisation avec les scripts présents dans le toolkit Moses (Koehn et al., 2007). Le script de tokenisation nous permet de séparer les mots en plusieurs unités, visant à réduire le vocabulaire final du système. On utilise ensuite une technique visant à découper les mots en unités sublexicales (sous-mots). Cela permet d'une part, de réduire drastiquement le vocabulaire et d'autre part, de ne modéliser que des unités qui apparaitront un nombre de fois assez conséquent dans le corpus, aboutissant à une meilleure modélisation. Ils seront réalisés avec l'algorithme BPE tel que présenté dans la section 2.2.3.1 avec 30000 opérations d'assemblage. Les phrases de moins de 3 unités et de plus de 100 unités seront retirées. Enfin, un vocabulaire est extrait du corpus résultant. Seuls les 30000 unités les plus fréquentes sont modélisées, les autres étant assignées à l'unité correspondant aux mots inconnus ("unk").

Paire de langues	DE-EN	EN-TR
nombre de phrases	5.8M	200k
nombre de tokens	170M / 164M	9.5M / 4.6M
taille moyenne phrases en tokens	29 / 28	47 / 23
taille du vocabulaire en mots	2M / 1M	76k /163k

Tableau 4.1 – Statistiques sur les corpus employés lors de l'apprentissage de nos systèmes multilingues, ici les tokens sont composés de sous-mots.

Le tableau 4.1 regroupe les statistiques des données que nous allons employer. La paire de langues DE-EN peut être utilisée dans le sens opposé (EN-DE), les statistiques sont donc les mêmes pour cette paire de langues. Pour la paire de langues EN-TR, j'ai choisi de maintenir les mêmes quantités de sous-mots comme paramètre du modèle BPE. Cependant, le vocabulaire étant de base très petit dans cette paire de langues, le nombre de sous-mots dans le corpus est faible, il reste principalement composé de mots complets.

4.4 Expériences multilingues

Afin d'évaluer les performances de nos systèmes multilingues, il nous faut une base de comparaison fondée sur des systèmes mono-paires classiques.

Deux ensembles de systèmes contrastifs sont réalisés. Le premier est à base de

Paire de langues	DE-EN	EN-DE	EN-TR
Système à base de mots	19.8	12.5	5.09
Système à base d'unités BPE	18.3	17.1	3.09

Tableau 4.2 – Résultats en %BLEU des systèmes contrastifs mono-paires avec les différentes paires de langues avec des systèmes à base de mots et de sous-mots.

mots et le second est, quant-à lui, réalisé avec des sous-mots (précisément les unités BPE décrites précédemment). Nous réalisons 3 modèles mono-paires distincts où chaque paire de langues est apprise séparément. Les scores obtenus sont décrits dans le tableau 4.2. On peut noter que le système DE-EN à base de mots obtient un score supérieur d'un point et demi au système à base d'unités BPE. Cet écart s'accroît fortement lorsque l'on inverse l'ordre de langues (EN-DE) où le système obtient un score inférieur de plus de 4 points. Ceci est dû au fait que l'allemand contient des mots composés qui ne sont pas gérés par les prétraitements classiques. Ainsi, de nombreux mots sont assignés à l'unité unk alors que les modèles à base de BPE pourront décomposer ces mots en unités plus petites qui seront mieux modélisées par le système. L'utilisation de sous-mots a donc un impact significatif comparé aux résultats des systèmes contrastifs.

Dans les deux cas, le résultat en BLEU pour la paire EN-TR est extrêmement faible. Il est très probable que les traductions n'aient pas de sens. Les meilleurs systèmes que nous avons entraînés sur cette paire ont obtenu un score aux alentours des 7 points de BLEU. Les meilleurs résultats de la campagne d'évaluation cette année-là obtiendront des scores %BLEU dépassant les 13 points (García-Martínez et al., 2017). Cette paire de langues est donc très difficile à apprendre. L'impact le plus négatif de l'utilisation des sous-mots est pour la paire EN-TR qui dispose d'une quantité de données trop faible dans son corpus pour apprendre des représentations de sous-mots robustes. Elle voit donc ses performances diminuer.

Cependant, nos expériences ne vont pas se focaliser sur les performances de la paire de langues EN-TR mais plutôt sur l'évolution des autres paires dans le contexte multilingue. Pour la suite, nos expériences ont montré que les systèmes à base d'unités BPE étaient les plus performants et donc, nous ne nous référerons plus qu'aux systèmes contrastifs les utilisant.

Notre première expérience multilingue utilisant l'architecture multi-encodeurs et multi-décodeurs est illustrée par la figure 4.1. Nous avons un encodeur par langue source et un décodeur par langue cible. Un seul encodeur EN est présent car nous avons fait le choix d'un encodeur unique pour les paires de langues EN-DE et EN-TR. Nous voulons, grâce à cet encodeur, favoriser le transfert multilingue entre les

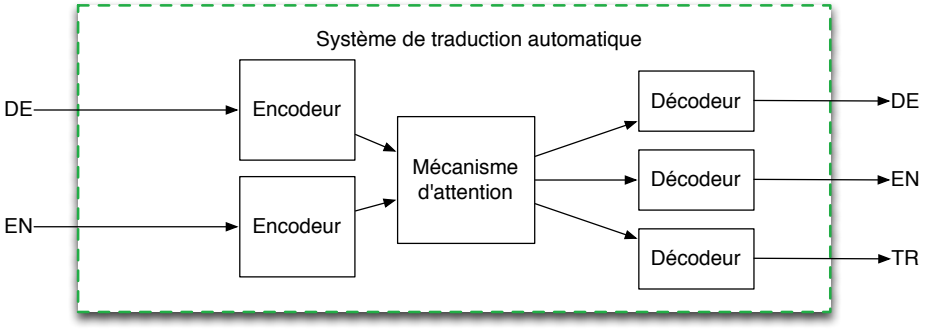

FIGURE 4.1 – Système multilingue neuronal avec ses différents encodeurs et décodeurs pour chaque paire de langues.

paires de langues, notamment à l'intention de la paire EN-TR qui est moins dotée.

Paire de langues	DE-EN	EN-DE	EN-TR
Système contrastifs mono-paire	18.3	17.1	3.09
Système multilingue	17.61	14.96	3.42

Tableau 4.3 – Résultats en %BLEU des systèmes contrastifs mono-paires avec les différentes paires de langues.

Le tableau 4.3 présente les résultats de cette expérience multilingue. Les scores des systèmes contrastifs sont rappelés pour comparaison. Cependant, nous comparons maintenant les scores de 3 systèmes distincts à un système unique multilingue pour les 3 résultats présentés. Nous pouvons voir, pour la paire de langues EN-TR, une amélioration qui n'est pas significative aux vues de la qualité du système contrastif. Avec ces résultats il est difficile de conclure sur l'impact du transfert multilingue.

En revanche, nous notons une baisse significative des résultats des deux autres paires mieux dotées. Une perte d'environ 0.5 points en DE-EN et de plus de 2 points en EN-DE.

À nouveau, cette baisse significative des paires mieux dotées est surprenante. Mon hypothèse est toujours liée à la possibilité que le transfert soit à l'avantage des paires moins dotées et au détriment des paires mieux dotées. Nous allons particulièrement nous intéresser à cela dans la section suivante.

4.5 Systèmes multilingues avec parties spécifiques

La baisse de performance pour les paires de langues mieux dotées est probablement liée à la cohabitation dans le système avec une paire de langues peu dotée. Celle-ci vient possiblement perturber les paramètres du modèle, notamment ceux de l'encodeur et ceux du mécanisme d'attention partagé.

Ainsi le pouvoir de modélisation du modèle pour les paires de langues bien dotées s'en voit réduit. Afin de restaurer la capacité du modèle à apprendre correctement les langues bien dotées, nous avons eu l'idée de réserver un espace spécifique pour chaque langue. Pour cela, nous avons alloué une partie de l'état caché qui sera spécifique pour chaque langue au sein de l'encodeur EN qui est partagé. Ainsi, l'encodeur est composé d'une partie commune, similaire à son fonctionnement originel mais aussi d'une partie supplémentaire qui s'active selon la paire de langues traduite. Notre hypothèse est que les parties spécifiques préservent les informations propres aux paires de langues et que la partie commune permet un transfert entre les différentes paires.

L'architecture avec partie spécifique est telle qu'illustrée par la figure 4.2. L'encodeur EN est modifié. Il possède 2 parties nommées "SPEC" qui sont activées suivant la paire couramment traduite. L'équation suivante représente le choix effectué par l'architecture suivant la langue à traduire.

$$\text{état caché } W_h = \begin{cases} \text{EN} +\!\!+ \text{ SPEC DE} & \text{si langue} = \text{DE} \\ \text{EN} +\!\!+ \text{ SPEC TR} & \text{si langue} = \text{TR} \end{cases} \tag{4.1}$$

Pour un batch donné, la langue de celui-ci conditionne quelle partie spécifique est concaténée ($+\!\!+$) à la partie globale (EN) de l'encodeur anglais. L'état caché de l'encodeur est toujours composé de sa partie globale concaténée à une partie spécifique.

Nous ajoutons une partie spécifique de 500 dimensions aux 1000 présentes de base dans l'état caché de l'encodeur. L'ajout de la partie spécifique mène à un espace plus grand. J'ai voulu évaluer l'impact qu'a cet agrandissement en testant aussi une réduction globale de l'espace de l'état caché. Pour cela, j'ai divisé les dimensions utilisées précédemment par deux. Avec 500 dimensions pour la partie cachée globale et 250 pour les parties spécifiques, nous pouvons mesurer cet impact.

Les résultats présentés dans le tableau 4.4 sont complétés par ceux de notre expérience d'encodeur à parties spécifiques. Les résultats n'évoluent pas pour la paire de langues DE-EN et EN-TR. En revanche, la paire de langues EN-DE qui enregistrait une perte significative de plus de 2 points par rapport au système contrastif, a maintenant une perte réduite à un point de BLEU.

Le dernier résultat est celui où la dimension de l'espace de l'état caché et celle

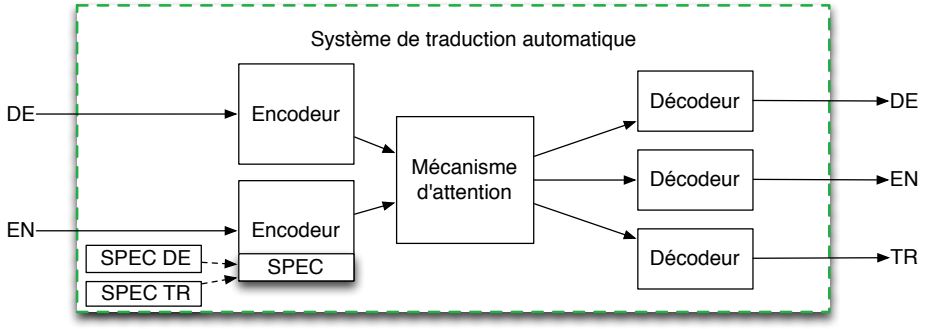

FIGURE 4.2 – Apprentissage multilingue neuronal avec ses différents encodeurs et décodeurs pour chaque paire de langues. L'encodeur EN possède des parties spécifiques pour les deux paires de langues employées.

Paire de langues	DE-EN	EN-DE	EN-TR
Systèmes contrastif mono-paire	18.3	17.1	3.09
Système multilingue	17.61	14.96	3.42
Système multilingue + spec	17.63	15.79	3.26
Système multilingue réduit + spec	17.11	15.71	2.34

Tableau 4.4 – Résultats en %BLEU des systèmes multilingues avec et sans encodeur à parties spécifiques.

des parties spécifiques sont réduites de moitié par rapport aux autres valeurs testées. Ce changement de dimensions ne montre pas d'impact majeur sur les performances dans notre cadre.

Notre choix d'encodeur EN unique peut avoir un impact sur les performances. À titre comparatif et afin d'évaluer l'impact de ce choix nous avons réalisé une expérience où la paire de langues DE-EN n'est pas présente dans notre système, et où nous avons un encodeur EN pour chacune des deux paires de langues restantes. Les résultats sont présentés dans le tableau 4.5.

On observe dans ce tableau une baisse d'un point de BLEU sur la paire EN-DE, et des changements non significatifs sur la paire EN-TR. Ce résultat montre que c'est bien le partage de l'encodeur entre les deux paires de langues qui est à la source de la baisse de performances dans la paire EN-DE. Cependant, les résultats EN-TR ne nous permettent aucune observation sur l'impact de ce choix sur la paire

Paire de langues	EN-DE	EN-TR
Systèmes contrastif mono-paire	17.1	3.09
Système multilingue enc EN séparé	16.08	3.3
Système multilingue enc EN séparé + spec	15.99	3.28
Système multilingue enc EN séparé réduit + spec	15.37	1.91

Tableau 4.5 – Résultats en %BLEU des systèmes multilingues avec encodeur (enc) EN séparé avec et sans parties spécifiques.

moins dotée. Les résultats ne sont pas impactés par la partie spécifique qui se trouve sans utilité avec un encodeur propre à chaque paire. Le dernier résultat, avec des dimensions réduites n'a pas d'impact important sur les paires mieux dotées. En revanche, on note une baisse d'un point BLEU pour la paire de langues EN-TR, il est cependant difficile d'estimer l'impact réel avec des valeurs aussi petites.

Ce résultat nous conforte dans l'idée que la partie spécifique est une technique pertinente pour préserver les résultats des paires mieux dotées et ainsi réduire le transfert négatif. Cependant, ces résultats ne nous permettent pas d'affirmer qu'il serait possible de maintenir complètement les performances des paires plus dotées, et donc de ne conserver que le transfert positif dans nos systèmes multilingues.

4.6 Conclusions sur le transfert multilingue

Dans ce chapitre, nous avons observé que dans un système multilingue servant à traduire des paires de langues disposant de quantités de données différentes, les performances variaient. Les paires de langues moins dotées profitent du transfert des paires mieux dotées, et voient leurs performances augmenter grâce à cette cohabitation. À l'inverse, les paires mieux dotées subissent un transfert négatif par cette cohabitation avec des paires moins dotées.

Nous avons proposé une approche de parties spécifiques dans l'encodeur de notre système de traduction. Cette partie spécifique a pour but de spécifier une partie de l'encodeur à la paire de langues couramment traduite. Son objectif est de réserver un espace aux différentes paires de langues pour qu'elles profitent de la cohabitation avec les autres paires, tout en maintenant une partie qui leur est propre. Nous avons vu que cet espace a permis aux paires de langues mieux dotées de récupérer une partie des performances perdues par la cohabitation au sein du système multilingue.

L'utilisation de partie spécifique est une réponse à la question « que transférer ? » (Pan and Yang, 2010) dans le cadre de l'apprentissage par transfert multilingue.

Cette question est peu étudiée en transfert multilingue car l'utilisation même de système multilingue implique le partage de l'architecture. Notre proposition de partie spécifique reflète notre volonté de contrôler le partage au sein du système multilingue pour ne conserver que ses aspects bénéfiques.

Nos expériences nous ont montré qu'il n'est pas simple d'obtenir une configuration idéale où plusieurs paires de langues sont apprises, et dont l'apprentissage profite les unes aux autres. Il semblerait que de grandes architectures arrivent à obtenir ces contextes multilingues plus performants (Johnson et al., 2017; Conneau et al., 2020). En revanche, avec des architectures de dimensions plus modestes, ces résultats semblent hors d'atteinte.

Nous avons montré que les parties spécifiques sont une solution possible à ce problème, cependant, nous n'avons pas pu complètement effacer le transfert négatif. Une extension de l'approche de spécialisation des poids que nous avons employée dans l'encodeur pourrait être appliquée à l'ensemble des poids du système pour mieux conserver les performances de l'ensemble des paires de langues. Cette hypothèse pourra faire l'objet de prochains travaux.

CHAPITRE 5

TRANSFERT SÉQUENTIEL EN TRADUCTION AUTOMATIQUE

5.1 Introduction

La seconde partie de nos expériences s'oriente sur les approches de traduction par transfert séquentiel. En traduction automatique, l'approche d'apprentissage par transfert séquentiel consiste à apprendre un système de traduction sur une paire de langues et ensuite de réutiliser ce modèle pour l'apprentissage d'une nouvelle paire. L'objectif est de capitaliser les connaissances acquises par le modèle « parent » et d'en faire bénéficier le modèle « enfant ».

Après s'être intéressé à l'état de l'art en transfert séquentiel (section 3.2), nous formulons plusieurs hypothèses auxquelles nous allons essayer de répondre avec nos expériences.

> **Première hypothèse : Le gain en performance du système enfant est dépendant de la quantité de données utilisée pour entraîner le modèle parent.**

Dans le cadre où nous allons nous placer, les systèmes parents sont de langues différentes et disposent de quantités de données différentes. Nous voulons vérifier si les systèmes parents que nous utilisons disposent d'assez de données pour offrir un transfert améliorant les performances des systèmes enfants.

> **Seconde hypothèse : La proximité des langues est un critère important pour un transfert performant.**

Les conclusions de Kocmi and Bojar (2018) ont montré que la proximité des langues n'est pas un critère primordial pour un transfert de qualité. À l'inverse, Dabre

et al. (2017) concluaient que la proximité des langues est un facteur important car ils obtiennent de meilleures performances avec un transfert provenant de langues proches.

Les deux premières hypothèses (supportées par les travaux récents de Dabre et Kocmi) nous amènent à la troisième hypothèse suivante :

> **Troisième hypothèse : Les facteurs de quantité de données et de proximité des langues sont interdépendants et leur impact varie selon leurs associations.**

La quantité de données du système parent et la proximité des langues seront les deux axes majeurs de notre étude sur l'impact du parent. Ainsi, nous supposons que, si peu de données sont disponibles, alors la proximité de la langue parent sera primordiale. À l'inverse, dans le cas où une grande quantité de données est disponible, la proximité de la langue sera un critère moins important. Nous allons vérifier l'impact de ces critères de façon isolée pour nos deux premières hypothèses. Nous cherchons maintenant à déterminer l'impact qu'ils ont l'un vis à vis de l'autre.

> **Quatrième hypothèse : La construction du vocabulaire est un levier pour améliorer le transfert.**

Nguyen and Chiang (2017) et Kocmi and Bojar (2018) ont montré que l'utilisation de sous-mots favorise le transfert entre le système parent et l'enfant. Étant donné que nous nous situons dans le cas où les paires de langues ne sont pas aussi bien fournies les unes que les autres, et que les algorithmes de découpage en sous-mots se basent sur les statistiques des corpus, cela pourrait désavantager les langues moins bien dotées.

5.2 Comment, quand et quoi transférer ?

Pour réaliser nos expériences nous nous somme basés sur le principe d'apprentissage par transfert trivial de Kocmi and Bojar (2018). L'avantage de cette approche est que l'ensemble des poids du système parent sont transférés à l'enfant. Il devrait en résulter un fort transfert entre les deux. Nous avons une architecture qui ne change pas entre l'apprentissage du système parent et du système enfant tel qu'illustré par la figure 5.1.

Seules les données d'apprentissage sont changées pour passer de l'apprentissage du parent à l'enfant. Ces contraintes nous imposent de réaliser des vocabulaires multilingues pour nos systèmes. Pour cela, nous regroupons les corpus sources du parent et de l'enfant pour la création des sous-mots, ainsi que celle du vocabulaire.

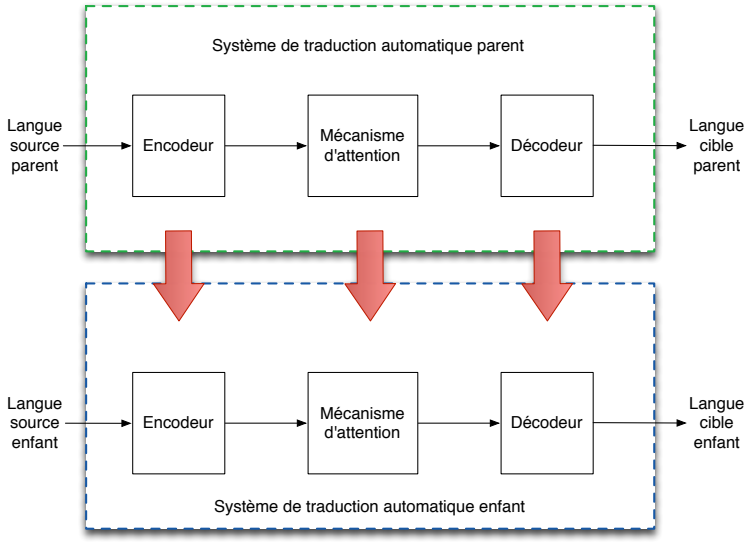

FIGURE 5.1 – Transfert séquentiel depuis un système parent vers un système enfant.

L'ensemble de la stratégie d'apprentissage doit être prévu, cela inclut aussi les prétraitements sur les différents corpus. Nous orienterons une partie de nos recherches sur ce phénomène et ses conséquences.

Pour les différents choix sur le transfert, nous revenons aux 3 questions à se poser dans le cadre de l'apprentissage par transfert présenté dans le chapitre 3.

À la première question « **que transférer ?** », le choix avait été fait dans des articles précédents de ne transférer que les *embeddings*, avec l'apprentissage par transfert trivial, c'est l'ensemble des paramètres que nous transférons. L'apprentissage par transfert trivial permet de réutiliser au mieux les connaissances du système parent en transférant l'intégralité des poids du système parent à l'enfant.

La seconde question est « **comment transférer ?** ». Pour répondre, nous nous en remettons au concept de l'apprentissage par transfert trivial. C'est l'apprentissage d'un parent complet menant à un modèle qui est réutilisé comme initialisation d'un système enfant. Nous allons chercher à partager le plus possible les langues et des tokens communs dans les vocabulaires des systèmes. Le choix avait été fait, dans plusieurs articles précédents (Zoph et al., 2016; Thompson et al., 2018), de geler les parties transférées. Avec l'apprentissage par transfert trivial, l'ensemble des paramètres sont transférés et réutilisés sans geler le moindre paramètre. Lors de l'apprentissage de l'enfant, celui-ci aura la possibilité de capitaliser sur ce qui a été acquis par le parent, mais aussi de remettre en cause les paramètres qui ne sont pas

adaptés à cette nouvelle langue. Dans cette configuration, seules les performances du modèle sur la paire de langues enfant nous importe. Contrairement à l'apprentissage multilingue, on ne souhaite pas conserver les performances des autres paires de langues.

Il pourrait être discutable de ne pas geler de poids dans le cadre d'un enfant très peu doté où l'enfant et son apprentissage pourraient mener à un transfert négatif en écrasant une partie de l'apprentissage du parent. Le risque est que l'apprentissage de l'enfant ne soit pas performant avec le peu de données dont il dispose. Une langue en commun entre le système parent et le système enfant, l'anglais en cible par exemple, pourrait souffrir de l'apprentissage d'un système enfant sur peu de données comparé à la base apportée par le système parent. Cette hypothèse n'a pas été étudiée dans nos expériences et pourrait faire l'objet de prochains travaux.

La dernière question, « **quand transférer ?** », interroge la pertinence du transfert. Dans notre configuration, la paire de langues estonien-anglais dont nous souhaitons améliorer les performances est peu dotée. Un transfert provenant d'une paire mieux dotée est donc une solution évidente. Cependant, nous proposerons une paire de langues proches mais disposant d'une quantité de données qui reste relativement faible pour les standards actuels en traduction automatique. Nous proposerons une paire de langues beaucoup mieux dotée mais d'une langue plus éloignée. Il est donc intéressant de comparer le transfert à partir de ces deux langues afin de mieux comprendre le lien existant entre quantité de données et proximité des langues, et leur impact sur les performances du système enfant.

La question « **quand transférer ?** » peut aussi être interprétée de façon plus littérale, elle peut faire référence à quel moment de l'apprentissage transférer ?

Cependant faut-il que le système parent soit optimisé pour la paire de langues parent, ou alors faut-il utiliser un modèle produit plus tôt dans l'apprentissage ? Un modèle parent complètement convergé pourrait être trop spécialisé pour la langue parent. Un système dont l'apprentissage a été interrompu avant la fin pourrait être plus générique et offrir un meilleur transfert pour un système enfant avec des langues différentes.

Kocmi and Bojar (2018) se sont intéressés à la question. Ils ont effectué un transfert séquentiel avec un parent pris à plusieurs temps de l'apprentissage jusqu'au modèle complètement appris (c'est à dire le plus optimisé). Ils observent et concluent que c'est avec le parent le plus entraîné qu'ils obtiennent les meilleures performances sur l'enfant. Donc je vais utiliser cette méthode dans mes travaux.

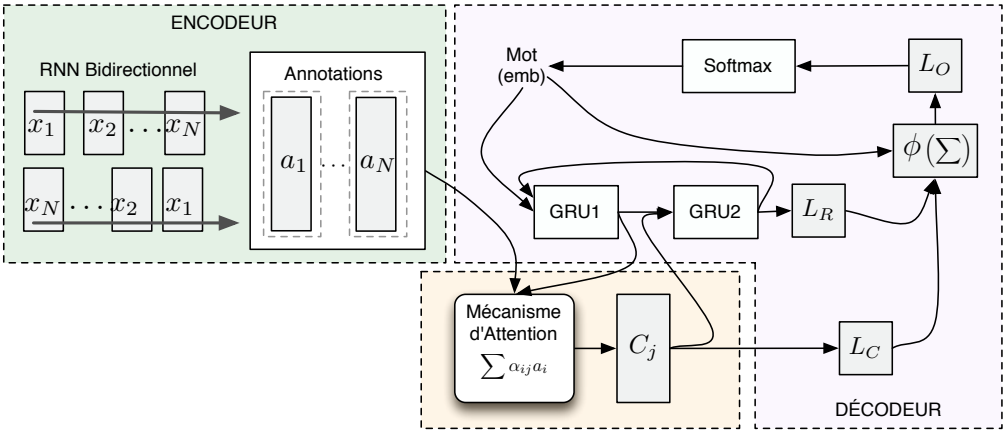

FIGURE 5.2 – Architecture neuronale Bi-GRU avec encodeur bidirectionnel et mécanisme d'attention

5.3 Architecture neuronale

Nous utilisons une architecture bout en bout de type encodeur/décodeur standard avec mécanisme d'attention en traduction automatique (Sutskever et al., 2014; Bahdanau et al., 2015). Nous allons utiliser le toolkit NMTPY dans sa dernière version en pytorch [1] (Caglayan et al., 2017). Ce toolkit réalisé au LIUM est un toolkit de traduction automatique neuronale. J'ai participé au développement de certaines fonctionnalités sur les versions précédentes.

L'architecture que j'utilise est composée d'un encodeur bi-directionnel et d'un décodeur à base d'unités récurrentes à portes, Bi-GRU (Cho et al., 2014) de taille 800. Cette architecture est présentée dans la figure 5.2. [2] Chaque phrase présentée en entrée est encodée en une annotation obtenue en concaténant les états internes de deux RNNs (l'un traitant la phrase dans un sens et l'autre dans l'autre sens). Chaque annotation représente la phrase, avec une attention particulière sur le mot courant. Le décodeur est composé de deux GRUs interagissant avec le mécanisme d'attention. La première GRU reçoit l'état précédent ainsi que l'embedding du dernier token généré en sortie. La seconde GRU reçoit la sortie de la première GRU ainsi que le vecteur de contexte fourni par le mécanisme d'attention. La couche de sortie du réseau reçoit la somme d'une tangente hyperbolique des *embeddings* du mot généré précédemment, ainsi que le vecteur de contexte et la sortie de la seconde GRU. Enfin, les probabilités de sorties de chaque token sur le vocabulaire cible sont calculées par une recherche par faisceau (beam search) qui permet de considérer

1. https ://github.com/lium-lst/nmtpytorch
2. Cette figure est extraite de García Martínez (2018)

plusieurs hypothèses de traduction partielles. La traduction retenue sera celle ayant la plus grande probabilité une fois le décodage de la phrase source effectué. Pour l'évaluation de nos expériences, nous nous reposons sur la métrique BLEU présentée dans la section 1.3.

Pour le reste de notre architecture ; les plongements lexicaux (*embeddings*) sont de taille 400. Nous appliquons un *dropout* (Srivastava et al., 2014) de 0.3 sur les *embeddings*, sur le contexte avant qu'il soit fourni au mécanisme d'attention et sur la sortie avant le *softmax*. Nous utilisons Adam (Kingma and Ba, 2015) pour optimiser les poids. Les poids sont initialisés d'après He et al. (2015). Le taux d'apprentissage est initialisé à 1.10e-4 et la taille d'un batch est de 32. Cette architecture est la seule configuration utilisée pour tous les systèmes présentés ultérieurement.

Le dimensionnement de l'architecture est un sujet intéressant avec l'apprentissage par transfert séquentiel. Nous devons définir des dimensions qui n'évolueront pas lors du passage du parent à l'enfant pour maintenir la simplicité du transfert trivial. Cependant, une taille de réseaux de neurones est optimale pour un système (LeCun et al., 1990) et nous nous retrouvons ici à définir une taille unique pour un système parent et un système enfant. Nous avons, par définition, un jeu de dimensions pour deux systèmes et un choix doit être fait. LeCun et al. (1990) montrent qu'un système neuronal obtient des performances optimales avec un dimensionnement de son architecture juste. Pour cela, l'architecture doit être assez grande pour permettre un apprentissage robuste sur les données présentées. Mais elle doit aussi être d'une taille raisonnable pour éviter que le système n'exploite un trop grand espace pour sur-apprendre, c'est-à-dire se spécialiser sur les données qui lui sont présentées et perdre en capacité de généralisation. Pour obtenir un apprentissage optimal, il faut donc une architecture bien dimensionnée.

Je remarque que cette réflexion sur le dimensionnement de l'architecture n'a pas fait l'objet de recherches poussées dans les articles précédemment cités sur l'apprentissage par transfert. Des expériences sur des systèmes classiques sans transfert nous donnent des dimensions optimales pour chaque système. Il nous faut maintenant fixer une dimension unique mais quels critères faut-il prendre en compte ?

Nous disposons de quantités de données différentes pour les deux systèmes. Les quantités de données sont un facteur important du choix des dimensions de l'architecture. En règle générale, plus on dispose de données et plus on peut se permettre d'utiliser une grande architecture sans que cela n'entraine de sur-apprentissage.

Il est aussi possible de partir des dimensions optimales pour le système parent car Kocmi and Bojar (2018) observaient qu'un système parent performant est primordial pour un transfert efficace. En effet, une architecture plus grande est sous optimale pour l'apprentissage de l'enfant, mais les principaux cas de sur-apprentissage

prennent place dans un cadre où le système n'a reçu aucun entraînement préalable, ce qui ne sera pas le cas ici. De plus, ces grandes dimensions permettront au système enfant d'apprendre sans modifier en profondeur l'apprentissage du parent pour les tokens qui lui sont propres, et qui n'ont pas été appris en amont. À l'inverse les tokens communs aux deux seront spécialisés lors de l'apprentissage de l'enfant. L'architecture retenue est celle qui offre les meilleures performances pour le système parent. Nous utiliserons le *dropout* qui est une technique visant a éviter le sur-apprentissage pour réduire l'impact de ce choix de grandes architectures.

5.4 Choix des données

L'origine de mes travaux avec cette approche est basée sur la campagne d'évaluation de traduction automatique WMT (Workshop on Machine Translation) de 2018. Lors de cette campagne, une dizaine de paires de langues sont proposées avec des données mises à disposition pour chacune d'elles. Les paires de langues estonien-anglais et finnois-anglais proposées par la campagne d'évaluation sont un très bon cadre d'application d'apprentissage par transfert car l'estonien et le finnois sont des langues proches partageant des racines communes de mots. Ainsi, on peut espérer qu'un transfert puisse grandement améliorer les performances pour l'estonien, dont les données sont disponibles en quantité relativement faibles.

En effet, la forte similarité entre ces deux langues proches est un facteur intéressant pour favoriser le transfert.

Nous utilisons les données présentées dans la campagne d'évaluation de traduction automatique WMT2018 (Bojar et al., 2018). Nous disposons de 2,5 millions de phrases parallèles pour la paire de langues estonien-anglais (ET-EN) ce qui correspond à 41M de tokens coté source et 52M de tokens coté cible. Nous avons choisi la paire finnois-anglais (FI-EN) avec comme langue source le finnois car cette langue est proche de l'estonien, ce sont toutes deux des langues finno-ougriennes. Nous disposons de 5 millions de phrases parallèles en finnois-anglais avec 78M de tokens coté source et 114M de tokens coté cible.

L'autre paire de langues que nous avons choisie est l'allemand vers l'anglais : l'allemand est une langue plus éloignée du finnois et de l'estonien. Pour cette affirmation nous faisons référence à la figure 5.3 [3]. Cette figure montre un arbre des grandes familles de langues indo-européennes et ouraliennes. On peut notamment voir un arbre à part, en bas à droite, qui regroupe les langues ouraliennes dont le finnois et l'estonien. Un grand arbre regroupe les langues européennes d'un côté, où on peut voir l'allemand qui est une langue germanique. De plus, le finnois et l'esto-

3. image provenant de https ://www.sltway.com/fr/le-corse-langue-ou-dialecte/

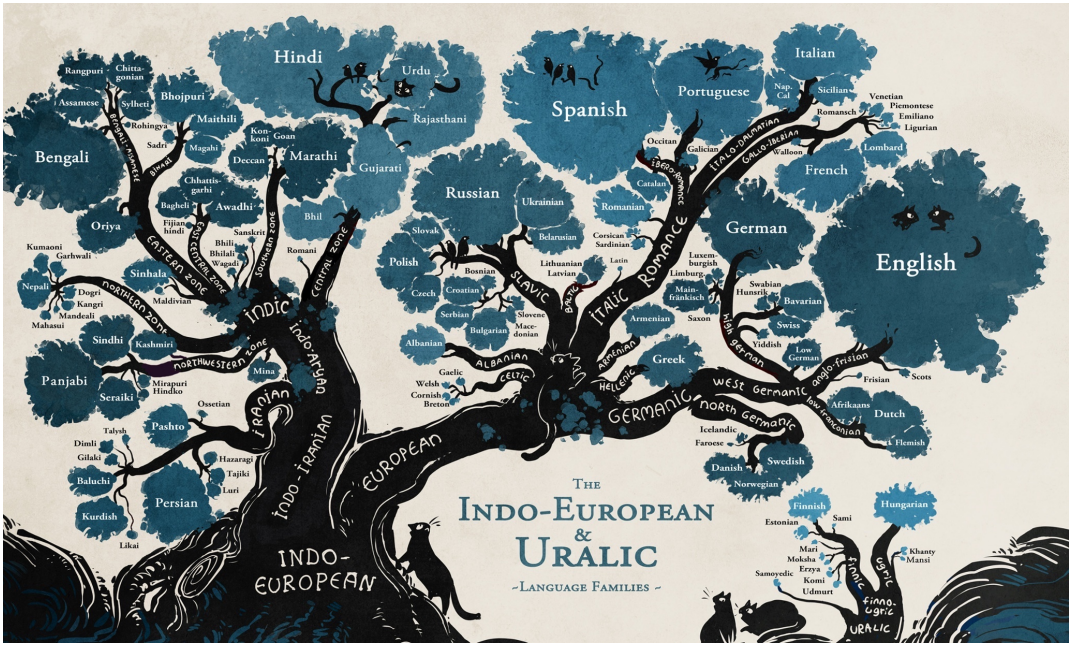

FIGURE 5.3 – Arbre de famille des langues

nien sont des langues dites agglutinantes, c'est-à-dire que les mots sont formés par assemblage de morphèmes. C'est une caractéristique intéressante car nos modèles utilisant des unités sublexicales (certes déterminées statistiquement), pourront probablement retrouver ces morphèmes et donc mieux modéliser la langue. La langue allemande quant-à elle, utilise des surcompositions, elle juxtapose déterminant et déterminé, parfois plusieurs fois, et crée de grands mots.

Nous avons donc à disposition, trois langues sources (allemand, estonien, finnois) dont deux sont proches : l'estonien (moins bien dotée avec 2.5M phrases) et le finnois (mieux dotée avec 5M de phrases), et une éloignée mais disponible en très grande quantité : l'allemand. Pour la paire allemand-anglais nous disposons de 40 millions de phrases parallèles, ce qui constitue un corpus de choix avec 570M de tokens coté source et 607M de tokens coté cible. Nous voulons découvrir si cette différence significative de quantités permettra à un système parent allemand-anglais de fournir un transfert au système enfant aussi efficace que pour le système parent finnois-anglais. Nous allons comparer ces différentes paires de langues pour évaluer l'impact de cette distance et de ces quantités de données.

Nous allons procéder, lors de nos expériences, à l'entraînement de systèmes ET-EN enfant utilisant les 2.5M de phrases parallèles disponibles. D'un premier abord, on pourrait se refuser à qualifier cela de « peu doté ». Nos premières expériences montreront que la performance des systèmes entraînés sur une telle quantité de

données n'est pas très élevée (voir section 5.5). Afin de se situer dans un cadre où l'aspect peu doté de la langue enfant est indiscutable, nous avons extrait 200k phrases parallèles des 2.5M disponibles de façon aléatoire. Ce petit corpus ET-EN nous servira de corpus enfant peu doté. Nous apprendrons des systèmes enfants sur ce corpus en parallèle du corpus ET-EN complet dans le but de comparer leurs comportements.

Afin de préparer les données, nous appliquons plusieurs phases de pré-traitement des corpus. Nous utilisons des unités sous-mots SPM (Kudo and Richardson, 2018). L'utilisation d'unités sublexicales permet un transfert plus important entre le parent et l'enfant (Nguyen and Chiang, 2017). Le phénomène est proportionnel à la quantité d'unités sous-mots en commun dans les deux langues mises en jeu. Nous apprenons des modèles de sous-mots séparés entre le côté source et le côté cible de notre architecture. Afin de ne pas bruiter notre système, nous retirons les phrases de moins de 3 sous-mots et de plus de 100 sous-mots. Finalement, les vocabulaires sont extraits en ne conservant que les unités apparaissant au moins 5 fois dans le corpus d'entraînement, et nous regroupons les autres en une unité modélisant les mots inconnus : <unk>.

5.5 Systèmes de base et analyse de l'impact des modèles de sous-mots

Les résultats des systèmes sont calculés sur les corpus de développement de la tâche de traduction de *news* de la campagne d'évaluation WMT2018.

Nous cherchons à répondre à notre première hypothèse sur le transfert positif selon laquelle un parent disposant de grandes quantités de données améliorera les performances d'un enfant disposant de peu de données. Pour cela, nous devons évaluer les performances d'un système de traduction sur la paire de langues enfant sans transfert qui nous servira de base de comparaison. Nous réalisons des systèmes de traduction automatique classiques (c'est à dire sans transfert) avec la paire de langues ET-EN afin d'évaluer ses performances avec plusieurs configurations telles que présentées par la figure 5.4.

Les résultats de base des systèmes ET-EN sont présentés dans le tableau 5.1. Nous comparons ici plusieurs modèles SPM utilisant différentes tailles de modèles sous-mots. Nous voulons évaluer quelle quantité de sous-mots mène aux meilleures performances pour la paire ET-EN.

La première colonne de résultats est celle sur l'ensemble du corpus ET-EN ; c'est-à-dire des 2.5M de phrases parallèles dont nous disposons. La seconde colonne

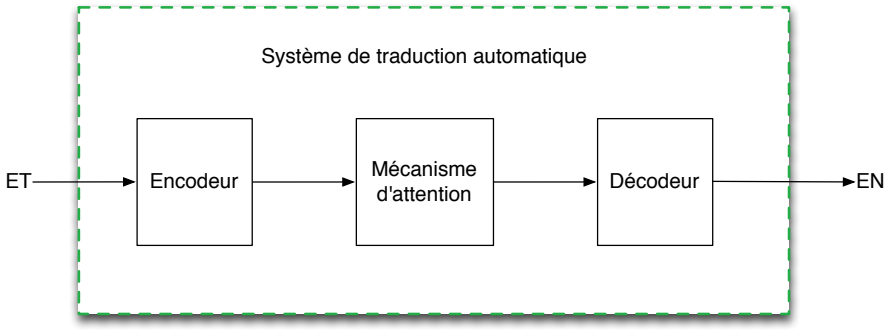

FIGURE 5.4 – Architecture encodeur/décodeur avec mécanisme d'attention des systèmes de comparaison Estonien-Anglais sans transfert.

présente des résultats sur 200k phrases parallèles qui sont notre cadre sous-doté simulé que nous avons présenté dans la sous-section 5.4.

Les résultats montrent que c'est avec 8000 et 16000 sous-mots que nos systèmes sont les plus performants. On observe plus particulièrement qu'il n'y a pas de différences de résultats entre 8000 unités SPM et 16000 unités SPM. En revanche, on observe une chute des performances avec 32000 unités SPM.

Quantité d'unités SPM	ET-EN 2.5M	ET-EN 200k
8000	14.12	10.69
16000	14.17	10.70
32000	13.6	10.10

Tableau 5.1 – Résultats en %BLEU pour la paire de langues ET-EN sans apprentissage par transfert avec des vocabulaires comprenant seulement des sous-mots provenant des corpus d'entraînement ET côté source et EN côté cible.

Pour l'apprentissage par transfert trivial, nous conservons le vocabulaire entre les systèmes parents et enfants. Nous conservons donc aussi les modèles SPM de sous-mots appris.

La quantité de sous-mots que nous choisissons pour ces modèles a un impact sur la qualité finale des systèmes. Nous avons voulu quantifier l'impact de ces vocabulaires multilingues sur les performances des systèmes. Pour cela, nous avons pris les modèles SPM créés pour notre apprentissage par transfert, qui comprennent les sous-mots parents et enfants, et avons appris un système ET-EN classique comme

Nb unités SPM	DE+ET 2.5M	FI+ET 2.5M
8000	10.64	14.47
16000	11.55	15.08
32000	12.52	13.87

Tableau 5.2 – Résultats en %BLEU pour la paire de langues ET-EN sans apprentissage par transfert avec des vocabulaires comprenant des sous-mots provenant de différents modèles SPM que nous utiliserons pour le transfert ensuite. Cela montre l'impact des modèles de sous-mots et des vocabulaires utilisés.

précédemment sans transfert. Les vocabulaires utilisés, et donc les *embeddings* qui en découlent, ne sont pas tous présents dans le corpus ET-EN car certains peuvent être propres au corpus parent.

Nous pouvons voir dans la table 5.2, que les modèles fondés sur les unités SPM comprenant de l'allemand obtiennent de moins bons résultats que ceux comprenant du finnois pour l'apprentissage d'un système ET-EN (à quantité de données égales). Le finnois et l'estonien étant des langues proches, il est vraisemblable qu'ils partagent plus d'unités sous-mots qu'avec l'allemand, qui est plus éloigné. De ce fait, ils cohabitent mieux dans le vocabulaire. On le voit avec les résultats du modèle SPM allemand-estonien dont les résultats augmentent lorsqu'on augmente le nombre de sous-mots. Pour le modèle SPM estonien-finnois les résultats baissent lorsqu'on utilise 32000 unités comparées aux 16000 unités précédentes. Il semble donc qu'un plus grand nombre de sous-mots soit plus propice pour le système allemand-estonien, alors que 16000 unités suffisent pour le système finnois-estonien.

Pour la suite des expériences, j'ai choisi d'utiliser 16000 sous-mots car c'est avec cette quantité que nous obtenons les meilleures performances en ET-EN dans le tableau 5.1. Les scores obtenus avec 8000 sous-mots sont similaires, cependant, j'estime qu'un plus grand nombre de sous-mots est propice à un partage plus important entre les langues. Le second tableau nous le montrait ; je choisis donc le plus grand des deux pour poursuivre nos expériences. Ce choix pourrait favoriser le parent finnois, car c'est avec cette quantité que ce parent est le plus performant. Nous aurions pu faire le choix de conserver 32000 unités pour le système parent-enfant utilisant l'allemand. Cependant, cette valeur impacte le nombre de plongements de mots et donc les dimensions de l'architecture. Pour réaliser une comparaison la plus juste possible entre les systèmes parents/enfants j'ai choisi de conserver 16000 unités pour tous les systèmes afin de conserver des architectures de dimensions comparables.

5.6 Expériences avec transfert

Nous cherchons maintenant à vérifier nos différentes hypothèses avec nos expériences sur la paire ET-EN après transfert. Pour cela, nous apprenons les systèmes parents que nous laissons complètement converger. Ensuite, l'apprentissage des systèmes enfants ET-EN débute en utilisant les modèles parents appris comme initialisation.

Paire de langues	40M	20M	10M	5M	2.5M
FI-EN	-	-	-	16.55	16.55
DE-EN	16.10	10.46	11.28	10.92	11.18

Tableau 5.3 – Résultats en %BLEU des modèles enfants ET-EN avec les différents systèmes parents.

Le tableau 5.3 présente les résultats des systèmes ET-EN enfants qui ont été appris après transfert en distinguant les différentes quantités de données utilisées pour apprendre le système parent. Pour mieux comprendre l'impact des quantités de données, nous extrayons de façon aléatoire des parties du corpus de plus petites tailles telles que 20M, 10M etc. Les différentes quantités de données utilisées pour le parent DE sont extraites de façon aléatoire des 40M disponibles. L'apprentissage par transfert séquentiel nous permet d'obtenir un score de 16.10 avec le parent DE et de 16.55 avec le parent FI comparé aux 14.17 %BLEU obtenu par le meilleur système sans transfert dans le tableau 5.1. Ce gain de plus de 2 points %BLEU répond à notre première hypothèse en montrant clairement l'impact positif du transfert sur la paire enfant estonien-anglais et cela peu importe la proximité des langues du système parent. Néanmoins, le parent finnois obtient de meilleures performances que le parent DE. Ce résultat montre l'importance de la proximité des langues pour le transfert. Notre troisième hypothèse s'intéresse au lien entre proximité des langues et quantités de données comme critères pour le transfert. Les résultats montrent qu'avec 5M de données finnoises, les performances de l'enfant estonien sont meilleures qu'avec un parent allemand disposant de 40M de données. Plusieurs découpages du corpus allemand nous permettent de tester différentes quantités de données. Cela nous permet de comparer plus finement l'impact des quantités de données vis à vis des résultats obtenus par le parent finnois.

Le parent FI-EN engendre les mêmes performances chez l'enfant avec 5M ou 2.5M ; ce qui est surprenant car nous sommes sur de petites quantités de données. En revanche, les résultats varient beaucoup avec les différentes quantités de données allemandes utilisées. En effet, que ce soit avec 5M ou 20M de données, les

performances chez l'enfant varient autour de 11 points de %BLEU ce qui est très en dessous du score de 14.17 obtenu par l'enfant sans transfert.

5.7 Analyse qualitative de l'évolution des plongements de mots

Au sein des systèmes neuronaux, les entrées du vocabulaire (mots et sous-mots) sont notamment représentées par les embeddings. La pertinence de ces représentations impacte directement les performances des systèmes et ainsi le score BLEU de traduction. Cependant, l'évolution du score BLEU ne permet pas de comprendre comment ces unités évoluent. Le score BLEU apparait donc comme insuffisant pour comprendre comment le transfert s'opère. Nous proposons donc une analyse qualitative permettant de mettre en évidence l'évolution des embeddings lors du transfert.

Dans cette section, nous proposons un autre axe d'analyse du transfert en nous intéressant à l'évolution des *embeddings*. Pour cela, nous comparons les espaces d'embeddings pré-transfert, provenant des systèmes parents, et post-transfert, correspondant aux systèmes enfants. Notre objectif est de compléter et confirmer nos observations sur le transfert.

Les embeddings de mots permettent de représenter les mots connus d'un vocabulaire par un vecteur de nombre réels.Un vecteur d'embedding est une représentation d'un mot du vocabulaire dans un espace de faible dimension (quelques centaines de paramètres en général). Ces vecteurs forment une matrice organisée de telle façon que les mots observés dans des contextes similaires (et donc ayant une sémantique proche) sont également proches dans cet espace d'embeddings. Cette représentation vectorielle permet de comparer les différentes représentations obtenues avec des systèmes différents.

La distance cosine se calcule entre 2 vecteurs et correspond au cosinus de l'angle θ entre ces 2 vecteurs, la similarité cosine est obtenue en faisant 1-distance cosine. La distance cosine est calculée par la formule 5.1 comme suit :

$$dist(A, B) = 1 - cos\theta = 1 - \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} \tag{5.1}$$

Nous la calculons entre deux vecteurs représentant chacun le même token dans nos vocabulaires parent/enfant. Le calcul de la **similarité cosine** entre les vecteurs donne un résultat compris entre -1 et 1. Par exemple, des vecteurs colinéaires ont un score de 1 (cos(0)=1), des vecteurs orthogonaux ont un score de 0 (cos(90)=0) et des vecteurs opposés ont un score de -1 (cos(180)=-1). La **distance cosine** est comprise entre 0 et 2. La distance maximale (2) correspond à l'écart maximal de

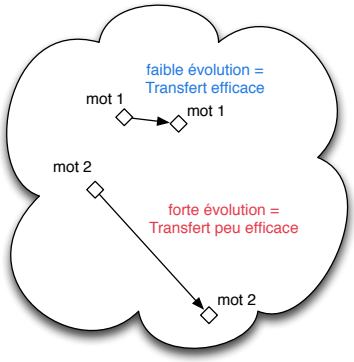

FIGURE 5.5 – Postulat d'analyse des plongements de mots basé sur leurs similarité cosine avant et après l'apprentissage du système enfant.

l'intervalle de similarité (de -1 à 1), une distance de 0 est obtenue avec deux vecteurs identiques.

La comparaison des embeddings d'un token avant et après transfert va nous permettre de quantifier l'évolution de la représentation de ce token. Pour cela, nous partons du postulat illustré par la figure 5.5.

Ce postulat se base sur l'évolution de la représentation d'un mot (ou sous-mot) entre le transfert depuis le parent servant donc de base d'apprentissage pour l'enfant et la fin de l'apprentissage de l'enfant. Cela nous permet d'analyser comment le système enfant à exploité la représentation des mots fournis par le système parent lors du transfert. Le postulat est qu'une représentation ayant peu évolué avec une forte similarité cosine est signe d'un transfert efficace car le système enfant n'a pas eu besoin de beaucoup modifier la représentation que le système parent lui a fourni et que celle-ci était donc pertinente pour l'enfant. À l'inverse, une faible similarité et donc une forte évolution seraient le signe d'un transfert peu efficace car le système enfant a beaucoup modifié la représentation transmise par le système parent. Ce postulat fonctionne en moyenne alors que certains cas spécifiques pris à part pourraient être sujet à débat.

5.7.1 Analyse de la similarité cosine des plongements de mots post-transfert

Pour nos systèmes, nous obtenons une similarité cosine entre les embeddings du parent FI et de l'enfant ET de 0.65 en moyenne. Le transfert DE->ET obtient, lui, une similarité moyenne de 0.60 entre ses embeddings respectifs. Ce premier résultat montre un avantage au transfert provenant du parent finnois au vu de la plus grande

similarité des embeddings après le transfert.

Nous faisons le choix de nous intéresser seulement aux tokens apparaissant dans le corpus estonien car ce sont les seuls embeddings qui pourront évoluer pendant le transfert (nos vocabulaires comprennent aussi des tokens n'apparaissant que dans la langue parente et dont les représentations n'évoluent pas pendant l'apprentissage de l'enfant).

Groupes de fréquence	10-100	100-1K	1K-10K	10K-100K	100K+
Nombre de tokens FI	39	2321	3336	1033	127
Nombre de tokens DE	88	3019	3009	1023	143

Tableau 5.4 – Nombre de tokens pour les deux parents suivant des regroupements de fréquences

Le tableau 5.4 présente les quantités de tokens dans la surface comprise entre deux valeurs de l'échelle logarithmique des fréquences. La première colonne regroupe les sous-mots apparaissant de 1 à 10 occurrences, une seconde de 10 à 100, puis 100 à 1000 etc.

La figure 5.6 est composée de deux cartes de chaleur (heatmap) qui présentent la similarité cosine en fonction de la fréquence des tokens (échelle logarithmique) entre les systèmes parent et enfant. La carte de chaleur en haut est celle du transfert depuis le parent DE et celle en bas depuis le parent FI. L'échelle de couleurs représente le nombre de sous-mots présents à une fréquence donnée en abscisse et une similarité cosine donnée en ordonnée. L'objectif est de mettre en évidence le lien entre fréquence des tokens dans le corpus et la variation de leurs embeddings. L'hypothèse est que les tokens de plus grandes fréquences dans le corpus enfant ont plus de chances d'évoluer beaucoup (faible similarité).

Sur la carte de chaleur du haut pour le transfert depuis le parent DE, on remarque deux groupes distincts. Ces groupes correspondent à une concentration des unités dont la fréquence est proche de 100 et 2000 respectivement et dont la similarité cosine moyenne varie entre 0.6 et 0.7. Dans la carte de chaleur en bas, correspondant au transfert depuis le parent finnois, on remarque un groupe principal d'unités dont la fréquence est comprise entre 100 et 10k dont la similarité est élevée (autour de 0.8).

Cette figure met en avant les différences de transfert entre les deux parents. Les différences présentes sur les groupes de grandes fréquences en jaune dans les cartes de chaleur montrent un lien entre similarité cosine et fréquence.

Nous estimons qu'une forte concentration à une meilleure similarité est une caractéristique d'un meilleur transfert. Selon ces critères, l'avantage est au parent

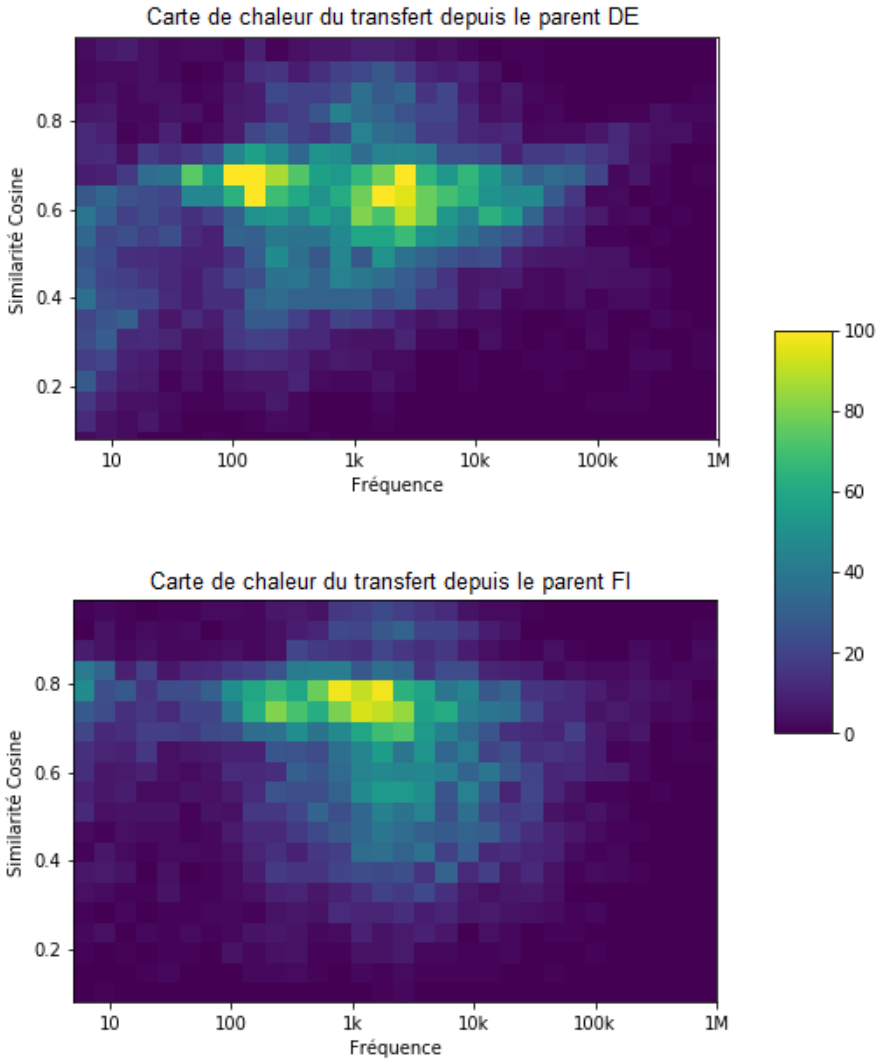

FIGURE 5.6 – Cartes de chaleur (heatmap) prenant en compte la similitude cosine et les regroupements en groupes de fréquences sur ses axes ainsi que la répartition de la fréquence en couleurs.

finnois où les mots fréquents sont regroupés en un groupe et obtiennent une similarité élevée. La carte de chaleur pour le parent DE montre deux groupes de plus forte fréquence (en jaune). Ils obtiennent une moins bonne similarité et un des groupes a une fréquence plus faible. Ce groupe de fréquence plus faible est une différence marquante entre les deux cartes de chaleur.

On note aussi des zones de concentration des unités chez le parent allemand, rayonnant autour des 0.4 de similarité cosine avec des fréquences autour de 10 occurrences. Cette zone n'apparait pas chez le parent finnois où seuls apparaissent autour de 0.8 de similarité des tokens aussi peu fréquents.

Ces observations suggèrent que pour la majorité des sous-mots (voir tableau 5.4), en jaune dans les deux cartes de chaleur, le parent finnois obtient une similarité plus élevée.

5.7.2 Analyse de la taille des sous-mots cosine des plongements selon leurs fréquences

Afin d'approfondir notre analyse, nous proposons une autre hypothèse de travail : le partage entre les langues et la taille des sous-mots communs en résultant sont des facteurs d'amélioration du transfert. Nous nous attendons à ce que la distance entre les langues allemande et estonienne mène à des tailles de sous-mots plus petites en moyenne car les langues ont peu en commun. Comment cela impacte-t-il nos systèmes ?

Pour répondre à cette question, la figure 5.7 présente la taille moyenne des sous-mots suivant nos regroupements de mots par fréquence. On retrouve en abscisse nos groupes de fréquences, en ordonnée la taille moyenne des sous-mots en caractères, et les couleurs qui représentent nos deux langues parents. Les quantités de tokens par regroupements sont toujours celles présentées dans le tableau 5.4.

Nous observons plusieurs comportements intéressants. Tout d'abord, à l'exception du premier groupe représentant les sous-mots les moins fréquents, plus un sous-mot est fréquent, plus il est petit. La tendance est assez claire ; les sous-mots finnois sont plus grands en moyenne que les sous-mots allemands. Les modèles SPM sont appris sur le corpus de la langue parent, finnois ou allemand, ainsi que sur la langue enfant, l'estonien dans les deux cas. Les mots communs aux langues parent et enfant influencent les découpages en sous-mots effectués, notamment lorsqu'un mot est fréquent dans les deux langues. Plus deux langues ont en commun, plus le modèle SPM pourra conserver des mots complets pour réaliser son vocabulaire. Cela conforte notre conclusion selon laquelle la proximité du finnois vis à vis de l'estonien conduit à des sous-mots plus grands et de plus grande similarité, ce qui

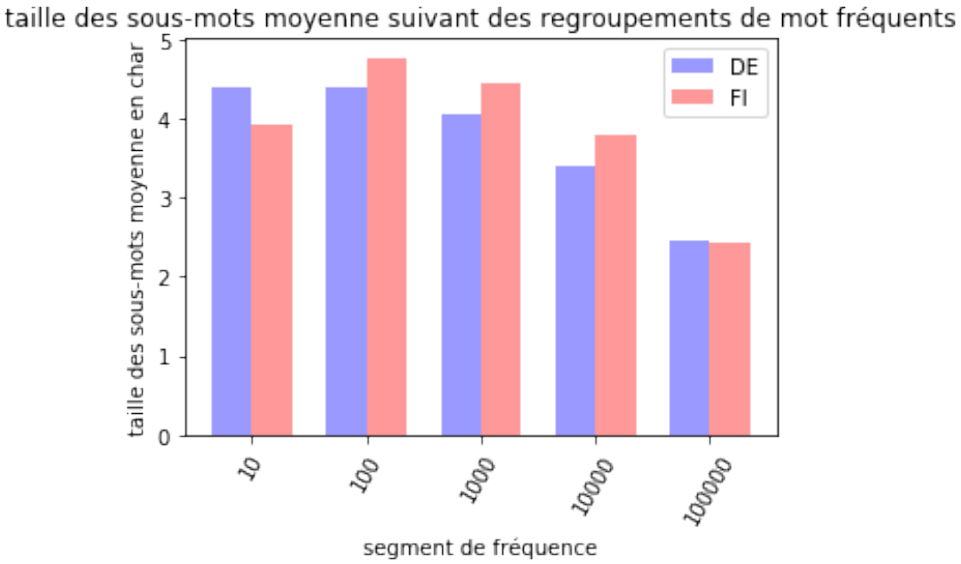

FIGURE 5.7 – Taille moyenne des sous-mots selon des regroupements de mots par fréquence (échelle logarithmique).

favorise le transfert.

Nous concluons que le parent finnois amène un meilleur transfert à l'enfant estonien pour deux raisons principales. Tout d'abord, une plus faible évolution des embeddings (plus haute similarité moyenne) implique certainement une meilleure réutilisation des embeddings du parent. Ensuite, la taille moyenne des unités plus élevées suggère un transfert plus important dû à la similarité des langues (elles utilisent plus de mots ou sous-mots communs).

5.8 Conclusions sur le transfert séquentiel

Dans ce chapitre, nous avons proposé un transfert séquentiel pour améliorer les performances d'un système estonien-anglais. Pour cela, nous avons utilisé deux systèmes parents différents, un système finnois-anglais et un système allemand-anglais. Le choix du finnois se base sur sa proximité à l'estonien qui est un facteur intéressant pour le transfert. À l'inverse, le choix de l'allemand, plus éloigné, repose sur la disponibilité d'une grande quantité de données pour amener un bon transfert.

L'approche par transfert séquentiel présentée permet d'améliorer les performances du système estonien-anglais avec les deux systèmes parents proposés.

Nous avons montré que pour obtenir un transfert de qualité, la quantité de données des parents et la proximité des langues avec l'enfant sont des caractéristiques

primordiales. Il apparait donc que, lorsque la langue du parent est éloignée de celle de l'enfant, il faille une plus grande quantité de données afin d'obtenir une performance similaire après transfert. Une langue proche nécessitera moins de données pour obtenir ces performances. Cela répond à notre troisième hypothèse sur le lien entre les quantités de données et la proximité des langues.

Notre analyse des plongements de mots, à l'aide notamment de cartes de chaleur, ont mis en évidence des groupes de sous-mots qui sont plus concentrés et qui obtiennent une similarité plus élevée chez le parent finnois démontrant que la proximité des langues engendre une meilleure réutilisation des embeddings du modèle parent sans avoir à les adapter avec les données de l'enfant (ce qui est l'objectif du transfert pour les langues peu dotées). Notre dernière expérience sur la taille des sous-mots a montré qu'ils sont aussi, en moyenne, plus grands avec le transfert provenant de ce système parent. Les différentes observations de cette analyse de l'évolution des plongements de mots lors du transfert me font conclure que le parent finnois s'avère être source d'un transfert plus pertinent que le parent allemand.

Nos expériences sur le transfert séquentiel apportent une réponse à la question « quand transférer ? » dans notre cadre de transfert séquentiel. Nos expériences nous ont montré que si un choix doit être fait entre proximité des langues et quantité de données, la proximité offre le meilleur transfert. Cela confirme les observations de Dabre et al. (2017), contredit celles de Kocmi and Bojar (2018) et apporte une réponse au débat sur ces critères dans la communauté (présentés dans la section 3.2.2).

CHAPITRE 6

TRANSFERT SÉQUENTIEL MULTILINGUE EN TRADUCTION AUTOMATIQUE

6.1 Introduction

Nous avons vu que la proximité du finnois vis à vis de l'estonien, ainsi que la quantité de données disponibles en allemand sont deux critères qui ont amené un transfert positif important aux systèmes enfants estoniens. La question se pose de la complémentarité de ces deux critères. Serait-il possible de définir une architecture qui tire partie de la langue finnoise afin de mieux modéliser les unités partagées avec l'estonien tout en exploitant la grande quantité de données allemandes. Dans ce but, nous proposons donc de combiner ces deux facteurs dans un système parent multilingue. Pour cela, nous avons utilisé un système avec encodeur universel (Ha et al., 2016) avec le corpus finnois et allemand en source de notre système.

Un encodeur universel permet de modéliser des unités représentant l'ensemble des langues sources disponibles (finnois, allemand et estonien dans notre cas). Dans notre cas, nous avons toujours de l'anglais en cible, donc il n'y a pas de changement pour le décodeur. En revanche, nous allons utiliser plusieurs langues dans l'encodeur. Le principal avantage d'un système universel est d'ajouter une, ou plusieurs langues, à notre système, sans avoir à faire évoluer l'architecture. Ce point est particulièrement important, car il nous permet de passer à un parent multilingue tout en continuant de respecter le concept d'apprentissage par transfert trivial. Cela permet d'utiliser un parent multilingue et de réaliser un transfert vers l'enfant sans changement d'architecture. En revanche, le choix du vocabulaire devient encore plus important car ce sont maintenant 3 langues sources qui sont présentes côté source.

Nous pouvons ainsi, comparer les résultats de ce parent multilingue aux pa-

rents finnois-anglais et allemand-anglais seuls, alors que nous avons maintenant un système multilingue comme parent. Johnson et al. (2017) ont montré que l'apprentissage en parallèle de plusieurs paires de langues avec une architecture universelle a un impact positif sur les résultats de traduction. Nous voulons vérifier si c'est aussi le cas dans notre approche de transfert.

6.2 Expériences en transfert séquentiel multilingue

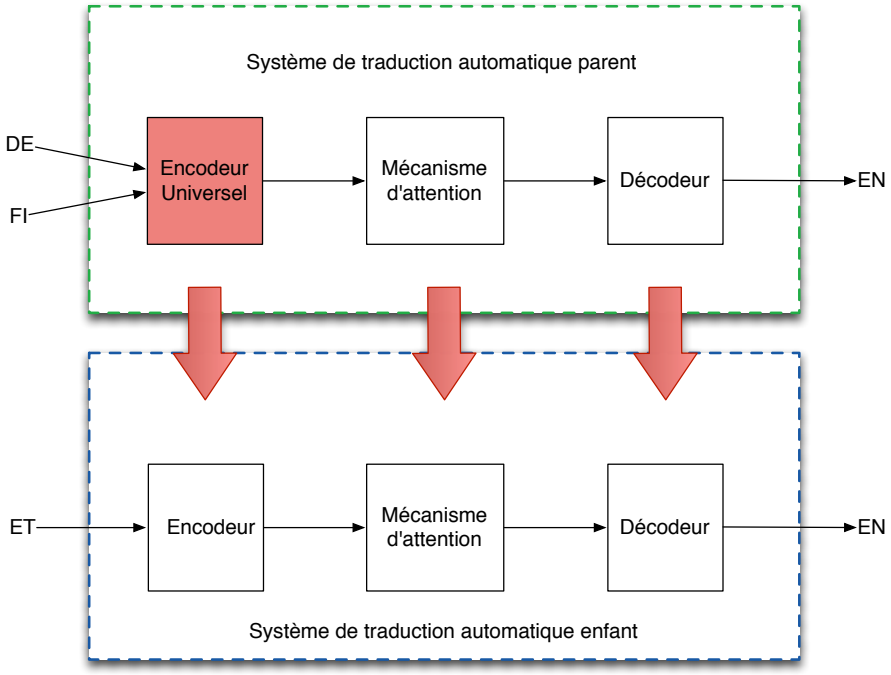

FIGURE 6.1 – Apprentissage par Transfert trivial d'un système parent multilingue DE+FI-EN vers un système enfant ET-EN en Traduction Automatique Neuronale

La figure 6.1 montre notre nouveau système parent multilingue avec 2 langues en entrée et l'utilisation d'un encodeur universel. Afin de pouvoir comparer les résultats obtenus avec un système parent à encodeur universel avec les systèmes précédents n'utilisant qu'une seule langue dans le parent, nous avons tenu à conserver un nombre similaire de paramètres du modèle. Par conséquent, nous conservons les mêmes dimensions (400 dimensions pour les embeddings, 800 pour l'encodeur et 800 pour le décodeur). Ainsi le modèle conserve le même pouvoir expressif, et les résultats seront dus à la combinaison des deux langues dans le parent. Le système se retrouve avec une tâche plus complexe à modéliser, la charge demandée au

mécanisme d'attention est bien plus importante notamment avec deux langues à traduire.

Nous utilisons donc un modèle SPM différent des précédents car il comporte cette fois-ci de l'allemand et du finnois provenant du système parent, en plus de l'estonien du système enfant pour le côté source. C'est donc un modèle SPM et un vocabulaire comprenant 3 langues côté sources pour les systèmes.

Le choix avait été fait par Johnson et al. (2017) de tagger les sous-mots propres à chaque langue, en ajoutant un suffixe représentant la langue dont il provient. Cependant, dans notre configuration, ce choix irait à l'opposé de la volonté de partager le plus possible de paramètres entre les langues pour favoriser le transfert, tel que présenté dans Nguyen and Chiang (2017).

Paire de langues	45M	40M	20M	10M	5M	2.5M
FI-EN	-	-	-	-	16.55	16.55
DE-EN	-	16.10	10.46	11.28	10.92	11.18
FI+DE-EN	15.71	14.06	14.44	14.37	14.53	14.47

Tableau 6.1 – Résultats en %BLEU des modèles enfants ET-EN avec les différents systèmes parents.

Les découpages de quantité de données pour l'apprentissage du système parent multilingue comprennent 5M de données finnoises qui sont complétées par des données allemandes pour atteindre les différentes valeurs (45M, 40M, 20M, 10M). Une répartition équitable entre les deux langues est utilisée pour les corpus comprenant 5M et 2.5M de données. Donc, pour les systèmes comprenant 20M et plus de données, l'agrandissement du corpus est uniquement composé de données allemandes. Lorsque un corpus n'est pas utilisé dans son intégralité, les données utilisées sont extraites aléatoirement de celui-ci.

Les résultats sont supérieurs au système contrastif ET-EN sans transfert, cependant, ils ne sont pas meilleurs qu'avec un parent classique (cf. Tableau 6.1). Ces résultats sont obtenus sur l'apprentissage du système enfant ET-EN sur l'ensemble des 2.5M de données disponibles pour celui-ci. Le meilleur score %BLEU que nous obtenons est 15.71 et ne dépasse pas les 16.55 obtenus par le parent finnois seul.

Lorsque moins de données sont disponibles, le parent multilingue obtient des résultats supérieurs au parent allemand. Ces résultats montrent plus de 3 points de %BLEU en moyenne en faveur du parent multilingue vis à vis du parent allemand pour 20M, 10M, 5M et 2.5M de données. Cependant, ces résultats ne dépassent pas ceux du parent finnois, c'est donc l'ajout de l'allemand dans le système multilingue qui dégrade les résultats. C'est seulement avec la plus grande quantité de données

allemandes (45M total dont 40M allemand) que les résultats augmentent avec 15.71 %BLEU surpassant d'un point le second meilleur système multilingue.

6.3 Expériences avec enfant très peu doté

L'hypothèse selon laquelle un système enfant disposant de très peu de données est plus dépendant du transfert provenant du parent a été étudié par Kocmi and Bojar (2018). Ils montrent une amélioration des performances importantes pour des systèmes enfants disposant de plus petites quantités de données. Nous nous plaçons dans une configuration similaire qui devrait donc accentuer les effets du transfert. Nous utilisons donc 200k phrases comme décrit dans la section 5.4. Nous voulons vérifier l'impact du transfert du système parent multilingue lorsque le système enfant est plus dépendant du parent.

Paire de langues	45M	40M	20M	10M	5M	2.5M
FI-EN	-	-	-	-	13.03	12.24
DE-EN	-	11.12	6.87	6.99	7.10	6.96
FI+DE-EN	11.05	10.41	11.29	11.68	11.54	11.72

Tableau 6.2 – Résultats en %BLEU des modèles enfants ET-EN avec 200k phrases avec les différents systèmes parents.

Pour rappel, notre meilleur système ET-EN dans cette configuration obtenait un score %BLEU de 10.70 dans la table 5.1. Les résultats de la table 6.2 nous montrent que les 3 systèmes parents utilisant l'ensemble des données disponibles offrent une amélioration des performances de l'enfant. Cependant, avec plus de 2 points d'écart, le parent finnois surpasse les autres mettant en avant que, lorsque nous avons peu de données pour le système enfant, la proximité des langues apporte les meilleures performances à l'enfant. Le système parent FI-EN surpasse clairement les autres dans cette configuration avec un score de 13.03 %BLEU. Le parent allemand et le parent multilingue ont des scores proches, autour de 11 %BLEU, améliorant peu les performances comparé au système sans transfert (10.70 %BLEU).

On note néanmoins la différence de résultats entre les deux parents FI-EN (13.03 %BLEU et 12.24 %BLEU). Là où les deux systèmes obtenaient les mêmes scores sur l'ensemble des données dans le tableau 6.1, une différence apparait maintenant avec moins de données. Cela met en évidence qu'avec moins de données, l'impact du transfert du parent est plus important.

Il n'y a pas de changement pour le parent multilingue, qui donne toujours un transfert moins performant que les meilleurs parents mono-paires. Cependant,

l'écart de performances entre celui-ci et le parent DE-EN est considérablement réduit et même non-significatif avec moins de 0.1 %BLEU d'écart (11.12 %BLEU vs 11.05 %BLEU). On observe toujours de meilleures performances du parent multilingue vis à vis du parent allemand lorsque moins de données sont disponibles. En revanche, plusieurs résultats de parent multilingue surpassent le meilleur parent allemand. La réduction de données pour le parent multilingue a pour conséquence une part plus importante de finnois dans les données d'apprentissage. On observe des scores supérieurs au parent allemand pour le parent multilingue utilisant 20M et moins de données allant de 11.29 à 11.72 de score %BLEU comparé au meilleur système parent allemand obtenant un score %BLEU de 11.12. Le finnois a clairement montré son impact positif pour l'enfant estonien et c'est avec la plus petite quantité de données que le système multilingue offre les meilleures performances pour l'enfant.

Nous avons vu la supériorité du système parent finnois-anglais ainsi que l'importance des données finnoises dans les performances de système parent multilingue. Ces éléments nous font conclure que la proximité des langues est un critère plus important que la quantité de données lorsque peu de données sont disponibles pour le système enfant.

6.4 Analyse des sous-mots

Étant donné que nous utilisons une architecture fixe, les différences de performances peuvent s'expliquer par trois critères principaux :

1. Un apprentissage plus robuste avec davantage de données.
2. Un apprentissage plus robuste avec des données proches de la langue ciblée.
3. Les unités présentes dans le vocabulaire.

Dans les sections précédentes, nous avons quantifié l'impact de la quantité et de la proximité des données disponibles pour le parent et l'enfant sur les résultats en terme de score BLEU. Cela correspond aux deux premiers critères.

Avec pour but de mettre en évidence les changements dans les unités incluses dans le vocabulaire opérés par la distribution des données d'entrainement, nous avons inspectés les différents corpus pré-traités utilisés lors de l'apprentissage et le transfert. Tout d'abord, nous avons porté notre attention sur les unités SPM et la manière dont les mots sont segmentés en unités sublexicales pour corpus finnois et allemand.

Les figures 6.2 et 6.3 montrent un extrait des corpus allemand et finnois traités par le modèle SPM obtenus à partir des données suivantes : 40M allemands + 5M

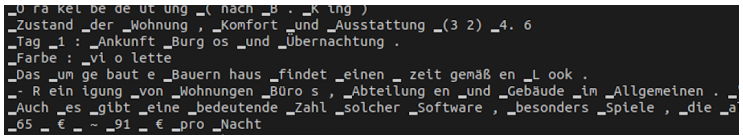

FIGURE 6.2 – Exemple de mots du vocabulaire DE avec un déséquilibre lors de la création des sous-mots

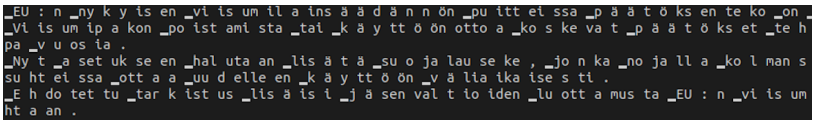

FIGURE 6.3 – Exemple de mots du vocabulaire FI avec un déséquilibre lors de la création des sous-mots

finnois + 2.5M estonien. Une différence marquante existe entre les deux corpus ; le corpus allemand est composé principalement de mots complets avec quelques sous-mots alors que l'on observe une sur-segmentation flagrante des mots du corpus finnois en unités très petites.

D'abord, SPM cherche à obtenir la meilleure couverture possible des données qu'il utilise pour éviter les mots hors vocabulaire. La valeur donnée à l'algorithme est celle de la taille de vocabulaire final qu'il doit optimiser. Il va donc chercher à découper en sous-mots les mots les moins fréquents, et conserver intacts les plus fréquents. De ce fait, les mots finnois qui n'ont pas été vus par l'algorithme se sont retrouvés découpés en une somme de petit sous-mots.

Ce déséquilibre vient de notre utilisation de SPM. En effet, SPM utilise 10M de phrases parallèles par défaut pour définir les sous-mots adaptés par souci de temps et de mémoire nécessaires au calcul. Le modèle SPM utilisé pour réaliser ces pré-traitements n'a pas utilisé équitablement des données allemandes et finnoises. On peut donc se poser la question de savoir si cette sur-représentation des unités allemandes ne serait pas profitable à cette langue et si un rééquilibrage n'apporterait pas de meilleures performances pour le modèle multilingue.

Ce déséquilibre a surement un impact sur les performances du système. Nous cherchons à définir si la distribution des unités dans le vocabulaire a un impact sur les performances du système, auquel cas, un rééquilibrage pourrait permettre de les améliorer.

6.5 Rééquilibrage des sous-mots

Afin d'évaluer l'impact de la distribution des unités dans le vocabulaire, nous avons considéré plusieurs configurations d'entrainement du modèle SPM servant

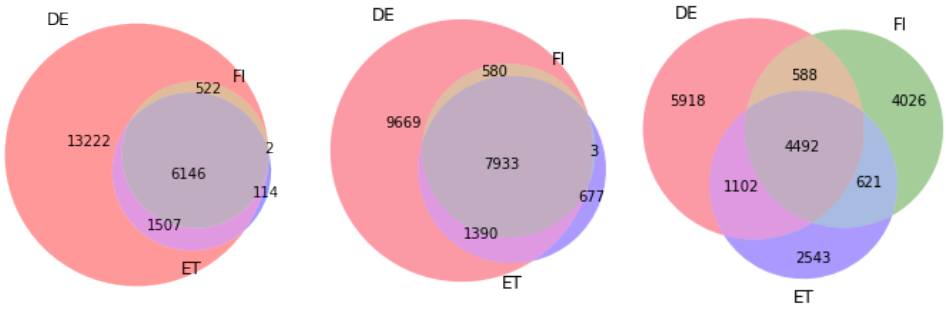

FIGURE 6.4 – Diagramme des répartitions de sous-mots entre les langues dans les vocabulaires des systèmes de traduction.

à pré-traiter les corpus d'apprentissage. Notre hypothèse est qu'une répartition plus équilibrée mènera à un meilleur système parent et un meilleur transfert pour l'enfant.

Nous avons considéré 3 configurations des données d'apprentissage du modèle SPM correspondant à 3 distributions différentes entre les langues mises en jeu (DE, FI et ET). Chaque configuration rassemble 10M de phrases parallèles au total.

L'algorithme utilisant 10M de phrases, nous appellerons donc notre distribution originelle 10-0-0 (10M DE + 0M FI + 0M ET). Pour obtenir une distribution plus équilibrée j'ai voulu conserver un maximum de finnois et d'estonien tout en donnant une plus grande valeur à l'allemand. Pour cela, j'utilise 5M de données DE, 3M de données FI et les 2.5M de données ET. Notre distribution originelle, est maintenant comparée à une distribution plus équilibrée 5-3-2.5 . Enfin, je réalise une distribution la plus équilibrée possible entre les langues avec 3M DE, 3M FI et 2.5M ET tout en respectant la limite des 10M de phrases.

Afin de mieux visualiser les répartitions, nous avons réparti les unités dans une vue ensembliste, présentée en figure 6.4. Ces graphiques correspondent à la répartition des unités selon qu'elles apparaissent dans le corpus allemand, finnois et/ou estonien. On remarque que la répartition 10-0-0 (à gauche) contient principalement des tokens spécifiques à l'allemand, ce qui était attendu, mais aussi des tokens communs aux 3 langues en plus petite proportion. Il n'y a pas, ou peu, de tokens spécifiques aux autres langues. La distribution 5-3-2.5 (au milieu) respectant mieux la distribution des données, montre un résultat similaire. En revanche, il y a moins de tokens spécifiques à l'allemand et plus de communs aux trois langues. Une fois de plus, très peu de tokens sont spécifiques au finnois ou à l'estonien. Enfin, la distribution équilibrée 3-3-2.5 (à droite) obtient la distribution sur le vocabulaire le plus équilibré avec des tokens spécifiques à chaque langue en quantité, ainsi que

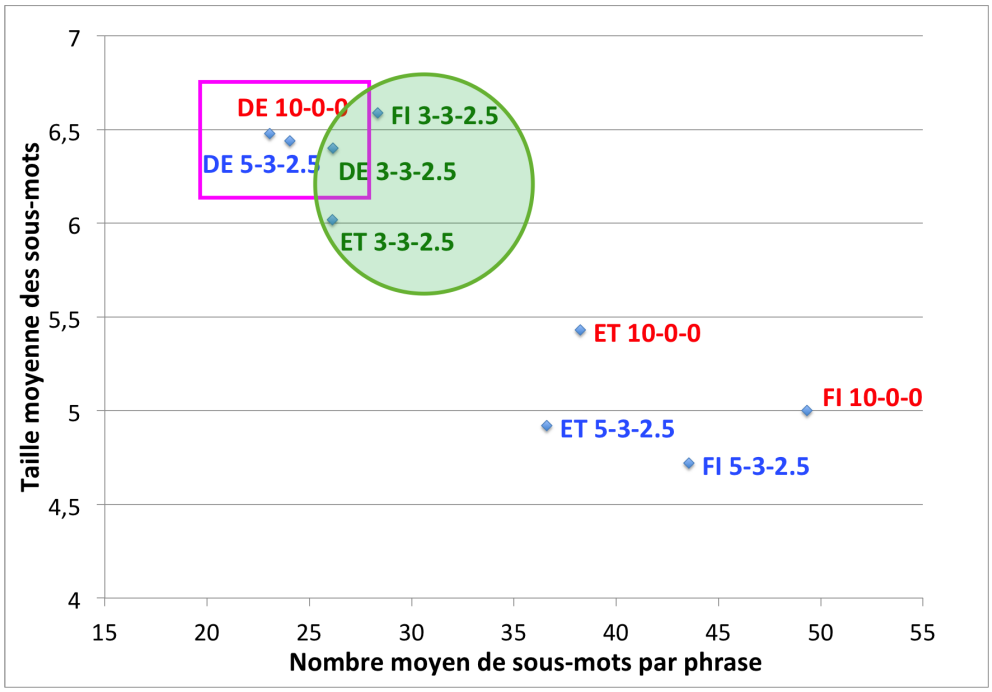

FIGURE 6.5 – Graphique de distribution des sous-mots des différentes distributions suivant le nombre de tokens par phrase ainsi que la taille moyenne des tokens.

des tokens communs aux 3 langues.

La distribution 10-0-0 pourrait représenter un concept présenté dans Kocmi and Bojar (2018) selon laquelle la qualité du système parent est primordiale pour un transfert efficace. En effet, avec cette distribution, les sous-mots exclusivement appris sur la langue du système parent sont optimisés pour celle-ci. Selon les conclusions de Kocmi and Bojar (2018) on pourrait s'attendre à ce que le système parent obtienne de bonnes performances et soit une source de transfert intéressante malgré son déséquilibre vis à vis du système global.

La seconde distribution reflète mieux la répartition réelle des données d'apprentissage (5-3-2.5). Elle prend en compte la majorité de données allemandes et des données finnoises et estoniennes, avec une plus grande part pour le finnois.

Enfin, la distribution 3-3-2.5 qui est la plus équilibrée entre les 3 langues sources est un choix évident, forçant une équité entre les langues dans la répartition des sous-mots et des vocabulaires. Cette distribution équilibrée a pour but de tester l'hypothèse selon laquelle l'équilibre de la répartition des langues dans un système multilingue est primordiale pour obtenir les meilleures performances sur ses différentes langues.

Afin de visualiser l'impact des modèles SPM entrainés avec les différentes confi-

gurations mentionnées ci-dessus, nous avons calculé des indicateurs de segmentation des phrases et des mots en unités sublexicales. La figure 6.5 présente la taille moyenne des unités sublexicales en fonction de la longueur moyenne des phrases pour les corpus de chaque langue.

Elle montre la taille moyenne des sous-mots selon la taille moyenne de la phrase en nombre d'unités. La distribution 10-0-0 est en rouge, distribution 5-3-2.5 est en bleu, et la distribution 3-3-2.5 est en vert.

On peut voir dans le cadre rose que, quelque soit la distribution, le rapport nombre de tokens/taille de tokens est similaire pour l'allemand (DE) dans toutes les distributions. Il n'y a pas d'impact significatif sur l'allemand qui conserve une segmentation similaire malgré l'équilibrage des langues. Pour le finnois et l'estonien, la configuration 5-3-2.5 a un impact similaire sur la segmentation : réduction de la taille moyenne des sous-mots et réduction de la moyenne des phrases. Dans le cercle vert, nous observons que, quelque soit la langue, les langues de la distribution équilibrée 3-3-2.5 sont proches les unes des autres. C'est une conséquence directe et espérée de l'équilibrage des langues lors de la création des sous-mots. En effet, cette distribution regroupée est un signe clair d'équilibre des 3 langues dans le vocabulaire. On peut espérer que les modèles récurrents puissent mieux capturer les dépendances à longue distance avec une diminution de la taille des phrases et l'augmentation de la taille des tokens. Notre hypothèse est que cet équilibre retrouvé devrait avoir un impact positif sur le transfert du parent multilingue.

6.6 Expériences avec vocabulaire équilibré

Les différentes distributions de sous-mots pour le parent multilingue obtiennent les résultats présentés dans le tableau 6.3. Des quantités de données comparables ont été conservées pour les systèmes parent multilingues vis à vis des systèmes bilingues.

Une première observation est que les systèmes parents multilingues n'offrent pas de meilleures performances que les meilleurs systèmes parents mono-paires. Cependant, certains systèmes s'en approchent. Le meilleur résultat obtenu par un système parent multilingue est de 15.71 par le système 10-0-0, soit 0,84 de moins que le meilleur parent bilingue (16,55). Cela tend à confirmer les conclusions de Kocmi and Bojar (2018) selon lesquelles, un parent de bonne qualité est primordial pour la qualité du transfert. En effet, ici sans tokens propres à l'enfant ET-EN avec seulement ceux allemands, le système enfant obtient de meilleures performances qu'une distribution équilibrée entre les langues.

Les performances des systèmes multilingues sont proches les unes des autres. La

Parent Language Pair	45M	40M	20M	10M	5M	2.5M
FI-EN	-	-	-	-	**16.55**	**16.55**
DE-EN	-	**16.10**	10.46	11.28	10.92	11.18
FI+DE-EN 10-0-0 SPM	**15.71**	14.06	14.44	14.37	14.53	14.47
FI+DE-EN 5-3-2.5 SPM	13.20	13.45	13.86	14.05	14.01	13.77
FI+DE-EN 3-3-2.5 SPM	14.09	14.51	14.22	14.64	14.52	14.71

Tableau 6.3 – Résultats en %BLEU des systèmes enfants ET-EN utilisant l'ensemble des données du système enfant par rapport aux différents systèmes parent utilisés pour le transfert.

quantité de données du système parent ne semble pas avoir d'impact significatif sur le système enfant.

Cependant, nos différents découpages de corpus pour le parent multilingue utilise les 5M disponibles de données finnoises. Ce sont exclusivement des données allemandes qui viennent les compléter pour les corpus multilingues plus grands tels que 20M, 40M et 45M.

Une fois cela pris en considération, on observe que les parents multilingues les plus performants avec les configurations 5-3-2.5 et 3-3-2.5 sont obtenus par des systèmes disposant de 10M ou moins de données. Cela correspond aux systèmes utilisant un corpus multilingue composé à 50% d'allemand et 50% de finnois. Ces systèmes obtiennent des résultats autour de 14 points %BLEU surpassant le parent allemand de plus de 2 points (11.28 %BLEU). Les résultats nous poussent à penser que le rééquilibrage de données a bien un effet positif sur les performances du système parent multilingue. Le rééquilibrage permet notamment de ne plus être aussi dépendant des grandes quantités de données du parent bien doté car dans plusieurs cas, à quantité de données équivalentes, les parents multilingues sont plus performants que le parent allemand seul.

6.7 Expériences avec vocabulaire équilibré sur systèmes peu dotés

Dans un cadre où l'enfant est peu doté, l'impact du parent s'est révélé plus important sur les performances de l'enfant (section 6.3).

Nous cherchons maintenant à évaluer l'impact du rééquilibrage des distributions dans ce cadre. Nous utilisons toujours les mêmes 200k phrases parallèles extraites, aléatoirement, du corpus du système enfant pour entraîner celui-ci en utilisant le

Parent Language Pair	45M	40M	20M	10M	5M	2.5M
FI-EN	-	-	-	-	**13.03**	**12.24**
DE-EN	-	**11.12**	6.87	6.99	7.10	6.96
FI+DE-EN 10-0-0 SPM	**11.05**	10.41	11.29	11.68	11.54	11.72
FI+DE-EN 5-3-2.5 SPM	10.26	9.79	10.52	11.00	10.85	10.65
FI+DE-EN 3-3-2.5 SPM	**12.19**	12.05	11.89	**12.56**	11.93	**12.45**

Tableau 6.4 – Résultats en%BLEU des systèmes enfants ET-EN utilisant 200k phrases parallèles des données du système enfant par rapport aux différents systèmes parent utilisés pour le transfert.

transfert provenant des différentes distributions SPM de parents multilingues. Les résultats présentés dans le tableau 6.4 diffèrent de ceux obtenus sur l'enfant mieux doté. En effet, le meilleur système parent est toujours le mono-paire FI-EN. Cependant, le système parent DE-EN est surpassé par des systèmes multilingues.

Le parent multilingue 10-0-0 obtient de moins bonnes performances dans cette configuration avec 11.05 de score %BLEU. C'est le parent équilibré 3-3-2.5 qui est le plus performant avec 12.19 %BLEU utilisant l'intégralité de 45M de données disponibles. Cependant, le système obtient de meilleures performances encore avec 10M de données obtenant un score de 12.56 %BLEU qui surpasse l'intégralité des autres parents multilingues mais aussi les parents allemands mono-paires.

Ces résultats montrent que, dans cette configuration, c'est le parent équilibré qui est le plus efficace pour le transfert. Il obtient des performances avec seulement 0,47 %BLEU de moins que le parent mono-paire FI-EN qui reste le plus performant. C'est la première fois qu'un parent multilingue s'approche autant des performances du meilleur parent en passant sous les 0.5 point de %BLEU de différence. Dans ce contexte où le parent dispose de très peu de données, le vocabulaire équilibré a le potentiel le plus important car il laisse une plus grande place au finnois (langue proche) et à l'estonien (langue cible).

Les expériences avec le parent multilingue ont mis en avant une forme de transfert négatif observé vis à vis des parents classiques. Il est possiblement lié à l'utilisation de l'approche multilingue, avec l'encodeur universel, et des quantités de données pour les langues du parent. Un équilibrage des données a montré un impact significatif, notamment dans un contexte où l'enfant dépend plus de la qualité du parent. Ces expériences répondent à notre quatrième hypothèse sur l'importance des sous-mots et leur équilibre entre les langues pour le transfert.

6.8 Conclusion sur le transfert séquentiel multilingue

Dans ce chapitre nous avons proposé une approche qui combine le transfert séquentiel et multilingue. Pour cela, nous utilisons un système parent multilingue dont les poids sont utilisés comme initialisation d'un système enfant. Cette approche s'est avérée plus performante que certains systèmes parents classiques, notamment dans un cadre où peu de données sont disponibles pour le système enfant. Nous avons analysé les vocabulaires et les distributions de sous-mots à travers les langues de nos systèmes. Cette analyse a mis en lumière un déséquilibre entre les langues que nous traduisons. Un rééquilibrage des vocabulaires a montré une amélioration des performances du système enfant après transfert. Le système parent multilingue équilibré a même mené à de meilleures performances que le système parent allemand-anglais dans un cadre où l'enfant dispose de peu de données. Ces résultats confirment encore notre conclusion sur la proximité des langues et son importance par rapport aux quantités de données.

Cette approche a mis en évidence que l'équilibre des langues dans les systèmes multilingues est un critère important. L'analyse et le rééquilibrage des vocabulaires que nous avons effectués apportent une réponse à la question « comment transférer ? ». Nous avons vu que l'équilibre des langues dans le vocabulaire mène à une amélioration des performances. Cet équilibre des langues dans les systèmes multilingues n'est pas, à mon avis, assez pris en compte dans les travaux multilingues présentés dans la communauté.

CHAPITRE 7

CONCLUSIONS ET PERSPECTIVES

7.1 Conclusions

Les approches à base de transfert deviennent très courantes avec l'évolution des techniques à base de pré-entraînements notamment (Mikolov et al., 2013; Devlin et al., 2019; Peters et al., 2018; Radford, 2018). En traduction automatique, et plus globalement en traitement de la langue naturelle, il devient plus intéressant de réutiliser des modèles déjà appris, plutôt que d'apprendre des modèles en partant de zéro (Kocmi and Bojar, 2019).

Mes travaux de recherche se sont orientés sur le transfert s'opérant lors de l'utilisation de plusieurs langues dans des systèmes de traduction automatique. L'originalité de cette thèse repose sur les analyses du transfert ainsi que sur une proposition d'approche de transfert séquentiel multilingue. Nous avons apporté des réponses aux questions du transfert évoquées en introduction de ce manuscrit.

L'utilisation de parties spécifiques est une proposition de données à transférer qui répond à la question « **quelles parties doivent être transférées** ». Elles permettent de conserver les spécificités des langues et ainsi de maintenir de meilleures performances pour celles-ci, tout en réduisant le transfert négatif.

Nous nous sommes intéressés aux critères les plus importants pour le transfert séquentiel que sont la proximité des langues et les quantités de données. Notre objectif, à travers ces travaux a été l'explicabilité du transfert. Nous avons analysé ces critères grâce à des expériences en comparant les scores obtenus par les systèmes. Une analyse des plongements de mots et notamment de leur évolution durant le transfert est venue compléter nos expériences. D'une part, la proximité des langues mises en jeu et la quantité de données disponibles sont des facteurs importants

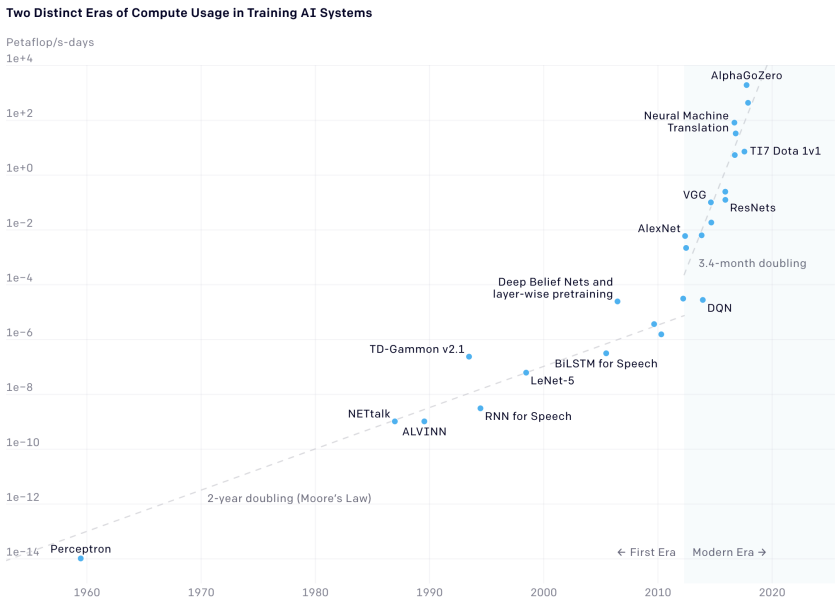

FIGURE 7.1 – Évolution des besoins en puissance de calcul pour apprendre des systèmes connus d'apprentissage automatique.

pour assurer un bon transfert. L'une des caractéristiques compense l'autre, ainsi un système appris sur une grande quantité de données d'une langue peu similaire à la langue d'intérêt fournira des performances proches de celles d'un système entrainé sur une moindre quantité de données mais impliquant une langue proche (finnois dans notre cas). D'autre part, l'analyse de la répartition des unités du vocabulaire (l'un des vecteurs principaux du transfert) entre les langues mises en jeu a permis de mettre en évidence l'impact de ce critère sur les performances du modèle. Les critères de sélection des données pour le transfert que nous avons présenté apportent des éléments de **réponse pour la pertinence du transfert**.

Enfin, nous avons utilisé une approche de transfert séquentiel multilingue qui cherche à combiner les deux approches de transfert. Ces expériences ont montré que l'équilibre des langues dans les vocabulaires des systèmes multilingues est un critère important pour de meilleures performances, notamment dans un cadre où peu de données sont disponibles pour la paire de langues enfant. L'équilibrage des langues est une technique pour améliorer les performances des systèmes multilingues de façon plus générale et apporte des éléments de réponse pour les **techniques employées pour le transfert**.

Les dernières avancées en apprentissage automatique poussent vers de plus grandes architectures nécessitant toujours plus de puissance de calcul. La figure 7.1

provenant du blog de OpenAI [1] présente l'évolution des besoins en puissance de calcul des principaux systèmes depuis 10 ans. De Krizhevsky et al. (2012) jusqu'à Silver et al. (2017) les besoins de puissances de calcul ont été multipliés par 300000. Cette augmentation massive des besoins de calculs soulève notamment des questions éthiques tel que le coût lié à l'apprentissage d'aussi grands modèles (Lacoste et al., 2019). L'omniprésence actuelle du transfert dans la communauté conforte notre choix d'orientation sur ce sujet dans nos travaux de thèse, comme en atteste les nombreux articles figurant dans la BERTologie (Rogers et al., 2020).

Nos expériences permettent de mettre en évidence l'importance des pratiques utilisées en apprentissage par transfert. Elles mettent en avant les réflexions nécessaires en amont de la réalisation de systèmes de traduction automatique utilisant un transfert. Nous espérons qu'elles permettront à la communauté de mieux prendre en compte ces critères lors de prochaines expériences de transfert.

7.2 Perspectives

Les travaux sur le **transfert multilingue** ont beaucoup évolué depuis ceux que nous avons expérimentés. L'analyse du comportement de ces approches sur les langues les mieux dotées sur lesquelles nos expériences montraient une perte de performances est une étude pertinente. Notre proposition est d'utiliser des parties spécifiques dans l'encodeur afin d'utiliser un espace particulier pour chaque langue afin que le modèle puisse s'exprimer. Il serait intéressant d'étendre notre approche à plus d'éléments des systèmes neuronaux ainsi que d'explorer cette approche avec des techniques d'apprentissage multilingue récentes (Aharoni et al., 2019; Tan et al., 2019; Heyman et al., 2019).

Nous avons porté nos recherches sur **l'analyse de la proximité des langues et des quantités de données** pour l'apprentissage par transfert en traduction automatique neuronale. Nous avons montré avec l'exemple du finnois et de l'allemand, représentant chacun un de ces aspects, l'impact de ces critères sur le transfert vers un système enfant estonien. Il y aurait un intérêt à poursuivre cette analyse avec une comparaison sur plus de langues différentes avec des proximités et des quantités de données différentes. Cela permettrait une analyse encore plus détaillée notamment sur le partage entre plusieurs langues dans le vocabulaire. Il serait intéressant pour l'analyse des plongements de mots de s'intéresser à certains mots du vocabulaire. L'évolution des plongements sur des mots particuliers apparaissant dans les langues parent et enfant pourraient permettre de mieux comprendre le transfert. Les mots transparents dont la similarité devrait être élevée et les faux-amis où la similarité

1. https ://openai.com/blog/ai-and-compute/

devrait être faible sont des exemples intéressants.

Nous avons expérimenté des **approches de parents multilingues** pour le transfert. Ces approches se sont montrées intéressantes, notamment dans un cadre où peu de données sont disponibles pour l'enfant. Les récentes avancées sur les systèmes multilingues avec de très grandes architectures (Wang et al., 2019a; Junczys-Dowmunt, 2019) sont une opportunité pour des expériences sur les parents multilingues. En obtenant toujours de meilleures performances, il est probable que ces systèmes procurent un meilleur transfert. Il serait intéressant d'analyser l'impact de ce genre d'évolution malgré le changement de cadre qui sera amené par l'évolution des architectures neuronales.

Publications

Mercedes García-Martínez, Ozan Caglayan, Walid Aransa, Adrien Bardet, Fethi Bougares, Loïc Barrault. 2017. LIUM Machine Translation Systems for WMT17 News Translation Task. In Proceedings of the Conference on Machine Translation (WMT), Volume 2 : Shared Task Papers, Association for Computational Linguistics, Copenhagen, Denmark, pages 288–295.

Ozan Caglayan, Walid Aransa, Adrien Bardet, Mercedes García-Martínez, Fethi Bougares, Loïc Barrault, Marc Masana, Luis Herranz, and Joost van de Weijer. 2017a. LIUM-CVC submissions for WMT17 multimodal translation task. In Proceedings of the Second Conference on Machine Translation, Volume 2 : Shared Task Papers. Association for Computational Linguistics, Copenhagen, Denmark, pages 432–439.

Ozan Caglayan, Mercedes García-Martínez, Adrien Bardet, Walid Aransa, Fethi Bougares, and Loïc Barrault. 2017b. NMTPY : A flexible toolkit for advanced neural machine translation systems. Prague Bull. Math. Linguistics 109 :15–28.

Ozan Caglayan, Adrien Bardet, Fethi Bougares, Loïc Barrault, Kai Wang, Marc Masana, Luis Herranz, and Joost van de Weijer. 2018. LIUM-CVC submissions for WMT18 multimodal translation task. In Proceedings of the third Conference on Machine Translation. Association for Computational Linguistics, Brussels,Belgium, pages 603–608.

Fethi Bougares, Jane Wottawa, Anne Baillot, Loïc Barrault, Adrien Bardet. LIUM's Contributions to the WMT2019 News Translation Task : Data and Systems for German-French Language Pairs. In Proceedings of the Fourth Conference on Machine Translation (Volume 2 : Shared Task Papers, Day 1). Florence, Italy, pages 129-133.

Adrien Bardet, Fethi Bougares et Loïc Barrault. 2019. Étude de l'apprentissage par transfert de systèmes de traduction automatique neuronaux. Actes de la Conférence sur le Traitement Automatique des Langues Naturelles (TALN) PFIA 2019. Volume II : Articles courts, Toulouse, France, pages 129-137.

Adrien Bardet, Fethi Bougares and Loïc Barrault. A Study on Multilingual Transfer Learning in Neural Machine Translation : Finding the Balance Between Languages. In proceddings of the 7th International Conference, Statistical Language and Speech Processing (SLSP) 2019, Ljubljana, Slovenia, October 14–16, 2019, pages 59-70.

BIBLIOGRAPHIE

Aharoni, R., Johnson, M., and Firat, O. (2019). Massively multilingual neural machine translation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3874–3884, Minneapolis, Minnesota. Association for Computational Linguistics.

Bahdanau, D., Cho, K., and Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. In Bengio, Y. and LeCun, Y., editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Banerjee, S. and Lavie, A. (2005). METEOR : An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Barrault, L., Bojar, O., Costa-jussà, M. R., Federmann, C., Fishel, M., Graham, Y., Haddow, B., Huck, M., Koehn, P., Malmasi, S., Monz, C., Müller, M., Pal, S., Post, M., and Zampieri, M. (2019). Findings of the 2019 conference on machine translation (WMT19). In *Proceedings of the Fourth Conference on Machine Translation (Volume 2 : Shared Task Papers, Day 1)*, pages 1–61, Florence, Italy. Association for Computational Linguistics.

Bengio, Y. (2012). Deep learning of representations for unsupervised and transfer learning. In Guyon, I., Dror, G., Lemaire, V., Taylor, G., and Silver, D., editors, *Proceedings of ICML Workshop on Unsupervised and Transfer Learning*,

volume 27 of *Proceedings of Machine Learning Research*, pages 17–36, Bellevue, Washington, USA. PMLR.

Bengio, Y., Bastien, F., Bergeron, A., Boulanger–Lewandowski, N., Breuel, T., Chherawala, Y., Cisse, M., Côté, M., Erhan, D., Eustache, J., Glorot, X., Muller, X., Lebeuf, S. P., Pascanu, R., Rifai, S., Savard, F., and Sicard, G. (2011). Deep learners benefit more from out-of-distribution examples. In Gordon, G., Dunson, D., and Dudík, M., editors, *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pages 164–172, Fort Lauderdale, FL, USA. PMLR.

Bojar, O., Federmann, C., Fishel, M., Graham, Y., Haddow, B., Huck, M., Koehn, P., and Monz, C. (2018). Findings of the 2018 conference on machine translation (wmt18). In *Proceedings of the Third Conference on Machine Translation, Volume 2 : Shared Task Papers*, pages 272–307, Belgium, Brussels. Association for Computational Linguistics.

Brown, P. F., Cocke, J., Della Pietra, S. A., Della Pietra, V. J., Jelinek, F., Lafferty, J. D., Mercer, R. L., and Roossin, P. S. (1990). A statistical approach to machine translation. *Computational Linguistics*, 16(2) :79–85.

Brown, P. F., Della Pietra, S. A., Della Pietra, V. J., and Mercer, R. L. (1993). The mathematics of statistical machine translation : Parameter estimation. *Computational Linguistics*, 19(2) :263–311.

Caglayan, O. (2019). *Multimodal Machine Translation*. PhD thesis. Thèse de doctorat dirigée par Deléglise, Paul et Barrault, Loïc Informatique Le Mans 2019.

Caglayan, O., García-Martínez, M., Bardet, A., Aransa, W., Bougares, F., and Barrault, L. (2017). Nmtpy : A flexible toolkit for advanced neural machine translation systems. *Prague Bull. Math. Linguistics*, 109 :15–28.

Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar. Association for Computational Linguistics.

Conneau, A., Baevski, A., Collobert, R., Mohamed, A., and Auli, M. (2020). Unsupervised cross-lingual representation learning for speech recognition.

Conneau, A., Schwenk, H., Barrault, L., and Lecun, Y. (2017). Very deep convolutional networks for text classification. In *Proceedings of the 15th Conference of*

the European Chapter of the Association for Computational Linguistics : Volume 1, Long Papers, pages 1107–1116, Valencia, Spain. Association for Computational Linguistics.

Dabre, R., Nakagawa, T., and Kazawa, H. (2017). An empirical study of language relatedness for transfer learning in neural machine translation. In *Proceedings of the 31st Pacific Asia Conference on Language, Information and Computation*, pages 282–286. The National University (Phillippines).

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). Bert : Pre-training of deep bidirectional transformers for language understanding. In Burstein, J., Doran, C., and Solorio, T., editors, *NAACL-HLT (1)*, pages 4171–4186. Association for Computational Linguistics.

Durrani, N., Dalvi, F., Sajjad, H., Belinkov, Y., and Nakov, P. (2019). One size does not fit all : Comparing NMT representations of different granularities. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1504–1516, Minneapolis, Minnesota. Association for Computational Linguistics.

Firat, O., Cho, K., and Bengio, Y. (2016). Multi-way, multilingual neural machine translation with a shared attention mechanism. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, pages 866–875, San Diego, California. Association for Computational Linguistics.

García Martínez, M. (2018). *Factored neural machine translation*. PhD thesis. Thèse de doctorat dirigée par Estève, YannickBarrault, Loïc et Bougares, Fethi Informatique Le Mans 2018.

García-Martínez, M., Caglayan, O., Aransa, W., Bardet, A., Bougares, F., and Barrault, L. (2017). Lium machine translation systems for wmt17 news translation task. In *Proceedings of the Second Conference on Machine Translation, Volume 2 : Shared Task Papers*, pages 288–295, Copenhagen, Denmark. Association for Computational Linguistics.

Glorot, X. and Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In Teh, Y. W. and Titterington, M., editors, *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, pages 249–256, Chia Laguna Resort, Sardinia, Italy. PMLR.

Gülçehre, Ç., Firat, O., Xu, K., Cho, K., Barrault, L., Lin, H., Bougares, F., Schwenk, H., and Bengio, Y. (2015). On using monolingual corpora in neural machine translation. *CoRR*, abs/1503.03535.

Ha, T., Niehues, J., and Waibel, A. H. (2016). Toward multilingual neural machine translation with universal encoder and decoder.

He, K., Zhang, X., Ren, S., and Sun, J. (2015). Delving deep into rectifiers : Surpassing human-level performance on imagenet classification. In *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, ICCV '15, page 1026–1034, USA. IEEE Computer Society.

Heyman, G., Verreet, B., Vulić, I., and Moens, M.-F. (2019). Learning unsupervised multilingual word embeddings with incremental multilingual hubs. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1890–1902, Minneapolis, Minnesota. Association for Computational Linguistics.

Hihi, S. E. and Bengio, Y. (1996). Hierarchical recurrent neural networks for long-term dependencies. In Touretzky, D. S., Mozer, M. C., and Hasselmo, M. E., editors, *Advances in Neural Information Processing Systems 8*, pages 493–499. MIT Press.

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

Hutchins, W. J. (2001). Machine translation over fifty years.

Jean, S., Cho, K., Memisevic, R., and Bengio, Y. (2015). On using very large target vocabulary for neural machine translation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1 : Long Papers)*, pages 1–10, Beijing, China. Association for Computational Linguistics.

Johnson, M., Schuster, M., Le, Q. V., Krikun, M., Wu, Y., Chen, Z., Thorat, N., Viégas, F., Wattenberg, M., Corrado, G., Hughes, M., and Dean, J. (2017). Google's multilingual neural machine translation system : Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5 :339–351.

Junczys-Dowmunt, M. (2019). Microsoft translator at WMT 2019 : Towards large-scale document-level neural machine translation. In *Proceedings of the Fourth*

Conference on Machine Translation (Volume 2 : Shared Task Papers, Day 1), pages 225–233, Florence, Italy. Association for Computational Linguistics.

Kim, Y., Jernite, Y., Sontag, D., and Rush, A. M. (2016). Character-aware neural language models. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, AAAI'16, page 2741–2749. AAAI Press.

Kingma, D. P. and Ba, J. (2015). Adam : A method for stochastic optimization. In Bengio, Y. and LeCun, Y., editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Kocmi, T. and Bojar, O. (2018). Trivial transfer learning for low-resource neural machine translation. In *Proceedings of the Third Conference on Machine Translation : Research Papers, WMT 2018, Belgium, Brussels, October 31 - November 1, 2018*, pages 244–252.

Kocmi, T. and Bojar, O. (2019). Transfer learning across languages from someone else's nmt model.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., and Herbst, E. (2007). Moses : Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic. Association for Computational Linguistics.

Koehn, P. and Knowles, R. (2017). Six challenges for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pages 28–39, Vancouver. Association for Computational Linguistics.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Pereira, F., Burges, C. J. C., Bottou, L., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc.

Kudo, T. and Richardson, J. (2018). Sentencepiece : A simple and language independent subword tokenizer and detokenizer for neural text processing. In Blanco, E. and Lu, W., editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018 : System Demonstrations,*

Brussels, Belgium, October 31 - November 4, 2018, pages 66–71. Association for Computational Linguistics.

Lacoste, A., Luccioni, A., Schmidt, V., and Dandres, T. (2019). Quantifying the carbon emissions of machine learning. *CoRR*, abs/1910.09700.

Lample, G., Conneau, A., Denoyer, L., and Ranzato, M. (2018). Unsupervised machine translation using monolingual corpora only. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11) :2278–2324.

LeCun, Y., Denker, J. S., and Solla, S. A. (1990). Optimal brain damage. In Touretzky, D. S., editor, *Advances in Neural Information Processing Systems 2*, pages 598–605. Morgan-Kaufmann.

Lee, J., Cho, K., and Hofmann, T. (2016). Fully character-level neural machine translation without explicit segmentation. *CoRR*, abs/1610.03017.

Lee, J., Cho, K., and Hofmann, T. (2017). Fully character-level neural machine translation without explicit segmentation. *Transactions of the Association for Computational Linguistics*, 5 :365–378.

Madnani, N., Tetreault, J., and Chodorow, M. (2012). Re-examining machine translation metrics for paraphrase identification. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, NAACL HLT '12, page 182–190, USA. Association for Computational Linguistics.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. In Bengio, Y. and LeCun, Y., editors, *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*.

Munkova, D., Hajek, P., Munk, M., and Skalka, J. (2020). Evaluation of machine translation quality through the metrics of error rate and accuracy. *Procedia Computer Science*, 171 :1327 – 1336. Third International Conference on Computing and Network Communications (CoCoNet'19).

Nguyen, T. Q. and Chiang, D. (2017). Transfer learning across low-resource, related languages for neural machine translation. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2 : Short Papers)*, pages 296–301, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Pan, S. J. and Yang, Q. (2010). A survey on transfer learning. *IEEE Trans. on Knowl. and Data Eng.*, 22(10) :1345–1359.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu : a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Peters, M., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Pratt, L. Y. (1993). Discriminability-based transfer between neural networks. In Hanson, S. J., Cowan, J. D., and Giles, C. L., editors, *Advances in Neural Information Processing Systems 5*, pages 204–211. Morgan-Kaufmann.

Pratt, L. Y., Mostow, J., and Kamm, C. A. (1991). Direct transfer of learned information among neural networks. In *Proceedings of the Ninth National Conference on Artificial Intelligence - Volume 2*, AAAI'91, page 584–589. AAAI Press.

Radford, A. (2018). Improving language understanding by generative pre-training.

Rogers, A., Kovaleva, O., and Rumshisky, A. (2020). A Primer in BERTology : What we know about how BERT works. *arXiv e-prints*, page arXiv :2002.12327.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning Representations by Back-propagating Errors. *Nature*, 323(6088) :533–536.

Sabour, S., Frosst, N., and Hinton, G. E. (2017). Dynamic routing between capsules. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 30, pages 3856–3866. Curran Associates, Inc.

Sennrich, R., Haddow, B., and Birch, A. (2016a). Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting*

of the Association for Computational Linguistics (Volume 1 : Long Papers), pages 86–96, Berlin, Germany. Association for Computational Linguistics.

Sennrich, R., Haddow, B., and Birch, A. (2016b). Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., Hubert, T., Baker, L., Lai, M., Bolton, A., Chen, Y., Lillicrap, T., Hui, F., Sifre, L., van den Driessche, G., Graepel, T., and Hassabis, D. (2017). Mastering the game of go without human knowledge. *Nature*, 550 :354–.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout : A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15 :1929–1958.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems*, volume 27, pages 3104–3112. Curran Associates, Inc.

Tan, X., Ren, Y., He, D., Qin, T., Zhao, Z., and Liu, T.-Y. (2019). Multilingual Neural Machine Translation with Knowledge Distillation. *arXiv e-prints*, page arXiv :1902.10461.

Thompson, B., Khayrallah, H., Anastasopoulos, A., McCarthy, A. D., Duh, K., Marvin, R., McNamee, P., Gwinnup, J., Anderson, T., and Koehn, P. (2018). Freezing subnetworks to analyze domain adaptation in neural machine translation. In *Proceedings of the Third Conference on Machine Translation, Volume 1 : Research Papers*, pages 124–132, Belgium, Brussels. Association for Computational Linguistics.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need. *CoRR*, abs/1706.03762.

Wang, H., Raj, B., and Xing, E. P. (2017). On the origin of deep learning. *CoRR*, abs/1702.07800.

Wang, M. (2019). Towards linear time neural machine translation with capsule networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural*

Language Processing (EMNLP-IJCNLP), pages 803–812, Hong Kong, China. Association for Computational Linguistics.

Wang, Q., Li, B., Xiao, T., Zhu, J., Li, C., Wong, D. F., and Chao, L. S. (2019a). Learning deep transformer models for machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1810–1822, Florence, Italy. Association for Computational Linguistics.

Wang, Z., Dai, Z., Poczos, B., and Carbonell, J. (2019b). Characterizing and avoiding negative transfer. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11285–11294, Los Alamitos, CA, USA. IEEE Computer Society.

Wieting, J., Berg-Kirkpatrick, T., Gimpel, K., and Neubig, G. (2019). Beyond BLEU :training neural machine translation with semantic similarity. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4344–4355, Florence, Italy. Association for Computational Linguistics.

Yang, B., Bender, G., Le, Q. V., and Ngiam, J. (2019). Condconv : Conditionally parameterized convolutions for efficient inference. In Wallach, H., Larochelle, H., Beygelzimer, A., d' Alché-Buc, F., Fox, E., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 32, pages 1307–1318. Curran Associates, Inc.

Yosinski, J., Clune, J., Bengio, Y., and Lipson, H. (2014). How transferable are features in deep neural networks ? In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems*, volume 27, pages 3320–3328. Curran Associates, Inc.

Zaremoodi, P., Buntine, W., and Haffari, G. (2018). Adaptive knowledge sharing in multi-task learning : Improving low-resource neural machine translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2 : Short Papers)*, pages 656–661, Melbourne, Australia. Association for Computational Linguistics.

Zhou, B., Bau, D., Oliva, A., and Torralba, A. (2019). Interpreting deep visual representations via network dissection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(9) :2131–2145.

Zoph, B., Yuret, D., May, J., and Knight, K. (2016). Transfer learning for low-resource neural machine translation. In *Proceedings of the 2016 Conference on*

Empirical Methods in Natural Language Processing, pages 1568–1575, Austin, Texas. Association for Computational Linguistics.

DOCTORAT BRETAGNE LOIRE / MATHSTIC

Le Mans Université

Titre : Architectures neuronales multilingues pour le traitement automatique des langues naturelles

Mots clés : Traduction Automatique, Multilingue, Systèmes Neuronaux

Résumé : La traduction des langues est devenue un besoin essentiel pour la communication entre humains dans un monde où les possibilités de communication s'élargissent. La traduction automatique est une réponse à l'évolution de ce besoin. Plus récemment, la traduction automatique neuronale s'est imposée avec les grandes performances des systèmes neuronaux qui ouvrent une nouvelle aire de l'apprentissage automatique. Les systèmes neuronaux exploitent de grandes quantités de données pour apprendre à réaliser une tâche automatiquement. Dans le cadre de la traduction automatique, les quantités de données parfois importantes et nécessaires pour apprendre des systèmes performants ne sont pas toujours disponibles pour toutes les langues. L'utilisation de systèmes multilingues est une solution pour répondre à ce problème.

Les systèmes de traduction automatique multilingues permettent de traduire plusieurs langues au sein d'un même système. Ils permettent aux langues disposant de peu de données d'être apprises aux côtés de langues disposant de plus de données, améliorant ainsi les performances du système de traduction.
Cette thèse se concentre sur des approches de traduction automatique multilingues en vue d'améliorer les performances pour les langues disposant de peu de données. J'ai travaillé sur plusieurs approches de traduction multilingues reposant sur différentes techniques de transfert entre les langues. Les différentes approches proposées ainsi que des analyses complémentaires ont révélé l'impact des critères pertinents pour le transfert. Elles montrent aussi l'importance, parfois négligée, de l'équilibre des langues au sein d'approches multilingues.

Title : Multilingual neural architectures for natural language processing

Keywords : Machine Translation, Multilingual, Neural Systems

Abstract : The translation of languages has become an essential need for communication between humans in a world where the possibilities of communication are expanding. Machine translation is a response to this evolving need. More recently, neural machine translation has come to the fore with the great performance of neural systems, opening up a new area of machine learning. Neural systems use large amounts of data to learn how to perform a task automatically. In the context of machine translation, the sometimes large amounts of data needed to learn efficient systems are not always available for all languages. The use of multilingual systems is one solution to this problem.

Multilingual machine translation systems make it possible to translate several languages within the same system. They allow languages with little data to be learned alongside languages with more data, thus improving the performance of the translation system. This thesis focuses on multilingual machine translation approaches to improve performance for languages with limited data. I have worked on several multilingual translation approaches based on different transfer techniques between languages. The different approaches proposed, as well as additional analyses, have revealed the impact of the relevant criteria for transfer. They also show the importance, sometimes neglected, of the balance of languages within multilingual approaches.