Compréhension de la parole dans un contexte multilingue

Gaëlle Laperrière

AVIGNON UNIVERSITÉ

THÈSE DE DOCTORAT D'AVIGNON UNIVERSITÉ

École Doctorale n°536
Agrosciences & Sciences

Mention de doctorat :
Informatique

Laboratoire Informatique d'Avignon

Présentée par
Gaëlle LAPERRIÈRE

Compréhension de la parole dans un contexte multilingue

Soutenue publiquement le 9 septembre 2024 devant le jury composé de :

Alexandre ALLAUZEN	PR à Université Paris Dauphine-PSL, LAMSADE	Rapporteur
Benoit FAVRE	PR à Aix-Marseille Université, LIS	Rapporteur
Fabrice LEFÈVRE	PR à Avignon Université, LIA	Président de jury
Marco DINARELLI	CR au CNRS, LIG	Examinateur
Nathalie CAMELIN	MCF à Le Mans Université, LIUM	Examinatrice
Philippe LANGLAIS	PR à Université de Montréal, DIRO, RALI	Examinateur
Yannick ESTÈVE	PR à Avignon Université, LIA	Directeur de thèse
Sahar GHANNAY	MCF à Université Paris-Saclay, LISN, CNRS	Co-encadrante de thèse
Bassam JABAIAN	MCF à Avignon Université, LIA	Co-encadrant de thèse

REMERCIEMENTS

Je tiens à remercier l'ensemble des membres de mon jury de thèse pour le temps qu'ils auront accordé à l'examen des travaux présentés dans ce manuscrit. Je souhaite remercier tout particulièrement mon directeur de thèse Yannick Estève ainsi que mes encadrants Sahar Ghannay et Bassam Jabaian pour leurs conseils avisés et le temps qu'ils ont pu m'accorder au cours de ces trois années.

Je tiens également à remercier les membres du Laboratoire Informatique d'Avignon, pour leurs collaborations et pauses café. Je remercie évidemment mes collègues de bureau pour leur soutien et les longues discussions que nous avons pu avoir, *parfois* par ma faute.

Enfin, merci à ma famille et amis pour m'avoir redemandé sans cesse le sujet de mes travaux, me poussant toujours plus dans l'exercice de vulgarisation scientifique.

Nous remercions l'Institut du Développement et des Ressources en Informatique Scientifique (IDRIS) pour leurs ressources de calcul (dossiers AD011012565 et AD011011838 GENCI).

RÉSUMÉ

Cette thèse s'inscrit dans le cadre de l'Apprentissage Profond appliqué au domaine de la Compréhension Automatique de la Parole. Son objectif principal consiste à tirer bénéfice de données existantes dans des langues bien dotées en annotation sémantique de la parole afin de développer des systèmes de compréhension performants dans des langues moins dotées.

Ces dernières années ont connu des avancées considérables dans le domaine de la traduction automatique de la parole grâce à de nouvelles approches permettant de faire converger les modalités audio et textuelle, cette dernière disposant de vastes quantités de données. Associant la compréhension de la parole à une traduction depuis une langue source naturelle vers une langue cible conceptuelle, nous considérons l'encodeur de parole SAMU-XLSR dont l'encodage enrichi sémantiquement est agnostique à la langue. Nous montrons l'impact positif de ce type d'encodeur dans un modèle neuronal de compréhension de la parole de bout-en-bout et étudions finement ses capacités d'encodage linguistique et sémantique. Cette étude se poursuit par la spécialisation de l'enrichissement de cet encodeur, dans l'objectif d'orienter son encodage vers le domaine sémantique des ensembles de données françaises MEDIA, italiennes PortMEDIA et tunisiennes TARIC-SLU. Une double spécialisation est proposée afin de préserver la faculté de l'encodeur à générer certaines abstractions sémantiques tout en limitant la perte de ses capacités cross-lingues pendant la phase classique de *fine-tuning* du modèle sur la tâche finale. Nos contributions ont permis de faire avancer l'état-de-l'art de la portabilité entre langues et domaines pour les ensembles de données MEDIA, PortMEDIA et TARIC-SLU.

Le projet SpeechBrain a été déterminant pour l'implémentation de nos expérimentations. Nous avons apporté notre contribution à ce projet open-source par l'intégration dans sa distribution officielle d'une recette complète pour l'ensemble de données MEDIA.

Mots-Clefs : Compréhension Automatique de la Parole, Apprentissage Profond, extraction de concepts sémantiques, représentations de la parole, multilinguisme, cross-linguisme, portabilité cross-lingue, portabilité cross-domaine

ABSTRACT

This thesis falls within the scope of Deep Learning applied to Spoken Language Understanding. Its primary objective is to leverage existing data of large resourced annotated languages for speech semantics to develop effective understanding systems in low resourced languages.

In recent years, significant advances were made in the field of automatic speech translation through new approaches that converge audio and textual modalities, the latter benefiting from vast amounts of data. By visualizing spoken language understanding as a translation task from a natural source language to a conceptual target language, we consider the SAMU-XLSR speech encoder, which generates a semantically enriched language-agnostic encoding. We demonstrate the positive impact of this type of encoder in an end-to-end speech understanding neural network and closely examine its linguistic and semantic encoding capabilities. This study continues with the specialization of its enrichment, aiming to direct its encoding towards the semantic domain of the French MEDIA, Italian PortMEDIA, and Tunisian TARIC-SLU tasks. A dual specialization is proposed to preserve the encoder's ability to generate certain semantic abstractions while limiting the loss of its cross-lingual abilities during the usual fine-tuning phase of the model on the final task. Our contributions participated in improving state-of-the-art in portability between languages and domains for MEDIA, PortMEDIA, and TARIC-SLU datasets.

The SpeechBrain project has played a crucial role in implementing our experiments. We contributed to this open-source project by integrating an exhaustive recipe for the MEDIA benchmark into its official distribution.

Keywords : Spoken Language Understanding, Deep Learning, semantic concepts extraction, speech embeddings, multilingual, cross-lingual, language portability, domain portability

TABLE DES MATIÈRES

TABLE DES FIGURES

LISTE DES TABLEAUX

INTRODUCTION

Ces dernières années ont vu naître de nombreuses applications concrètes au domaine de la Compréhension Automatique de la Parole, la recherche dans ce secteur ne cessant d'évoluer. Sur le plan industriel, l'extraction de la sémantique d'un signal de parole peut être utilisée pour faciliter la communication entre les humains et les machines, notamment pour le routage d'appels, les assistants vocaux, les réservations en ligne, la domotique et bien plus encore.

La compréhension de la parole fut définie par De Mori et al. [2007] comme l'interprétation des informations transportées par un signal de parole. Béchet [2007] définit le sens comme porteur de différentes perspectives : philosophique, linguistique, cognitive, mathématique voire computationelle. La sémantique est définie par Woods [1975] comme une organisation des relations entre symboles et signes d'un langage et leur signification. Cette thèse aborde l'extraction automatique du sens d'un signal de parole à travers ses signes et symboles afin de les conceptualiser en des représentations sémantiques structurées.

Évaluer la compréhension acquise par un système est complexe, c'est pourquoi il existe de nombreuses tâches de différents niveaux de précision [RABINER 2005]. Certaines catégorisent des documents audio en thématique [BOUCHEKIF 2016 ; MDHAFFAR, LAURENT et al. 2018] ou en génèrent des résumés automatiques [MURRAY et al. 2010 ; MASKEY et HIRSCHBERG 2003]. D'autres se focalisent sur la détection d'intention dans un segment de parole afin de connaître la volonté globale de l'utilisateur [B. LIU et LANE 2016 ; DESOT et al. 2019]. Certaines tâches auront pour but d'extraire la sémantique à un niveau plus fin, visant la compréhension de chaque mot prononcé. Il s'agira de tâches comme celles d'extraction d'entités nommés [NOUVEL et al. 2015 ; GALLIANO et al. 2009 ; GROUIN et al. 2011] ou de concepts sémantiques.

Un concept sémantique peut être défini comme une unité abstraite représentant une idée, allant au-delà des aspects syntaxiques formels du langage pour en capturer le sens [WOODS 1975]. L'extraction de concepts sémantiques sera principalement exploitée dans le cadre d'interactions humain-machine et dialogues transactionnels [JABAIAN 2012 ; MESNIL, DAUPHIN et al. 2015 ; TÜR et DE MORI 2011]. La forte liaison entre le domaine et le lexique sémantique utilisé pour une tâche d'extraction de concepts sémantiques exige un degré de précision important quant à la capture du sens de l'interaction.

Les tâches de Compréhension Automatique de la Parole tirent leur difficulté de la densité et complexité des informations acoustiques. Bien que les systèmes de bout-en-bout réalisent de nos

jours une extraction directe de la sémantique d'un signal de parole, cette transformation directe d'une représentation acoustique à une représentation sémantique fut longtemps considérée trop complexe [SERDYUK et al. 2018], la communauté se tournant vers des architectures en cascade composées d'une chaîne de traitements successifs [RAYMOND et RICCARDI 2007]. Dans ces systèmes, les erreurs de transcription se propagent inévitablement jusqu'au module d'extraction sémantique textuel [Y. GONG 1995 ; GHANNAY 2017 ; Y. WANG, ACERO et CHELBA 2003]. Les modèles de bout-en-bout ont de plus la capacité d'exploiter des informations acoustiques paralinguistiques contenues dans le signal de parole [P. PRICE et al. 1991 ; TRAN et al. 2017 ; SHRIBERG 2005]. Leur apprentissage unique permet de viser une optimisation pour la tâche d'extraction sémantique dès le traitement des données acoustiques, adaptant directement leur projection.

Avec la généralisation des approches de bout-en-bout pour la reconnaissance de la parole [AMODEI et al. 2016 ; Y. ZHANG et al. 2016], l'intérêt croît pour les appliquer à sa compréhension [HATMI et al. 2013 ; GHANNAY, CAUBRIERE, ESTÈVE et al. 2018]. Cette thèse suit ce mouvement, souhaitant tirer bénéfice de toute information acoustique significative pour l'extraction de sémantique depuis le signal de parole tout en minimisant la densité de nos systèmes.

Bien qu'il soit possible de fournir à un système neuronal les ondes brutes d'un enregistrement audio, Davis et Mermelstein [1980] démontrèrent l'utilité d'une paramétrisation fréquentielle du signal. Des méthodes non-neuronales furent longuement utilisées. Les nouvelles méthodes d'apprentissage couplées à des réseaux de neurones denses et complexes obtiennent ces dernières années le monopole de l'encodage de la parole. En 2019, Schneider et al. présentaient wav2vec. Avec l'arrivée des modèles Transformers [VASWANI et al. 2017], celui-ci fut retravaillé, donnant naissance à wav2vec 2.0 [2020], base de nombreux encodeurs de parole neuronaux. Des encodeurs monolingues ont vu le jour, comme l'encodeur français LeBenchmark [EVAIN, H. NGUYEN et al. 2021] que nous utilisons pour certaines expérimentations sur l'ensemble de données françaises MEDIA [BONNEAU-MAYNARD, AYACHE et al. 2006]. Ces modèles peuvent néanmoins être appris sur plusieurs langues à la fois. C'est le cas de XLS-R [BABU et al. 2022] et du modèle encodeur-décodeur multilingue et multi-modal Whisper [RADFORD, J. KIM et al. 2023].

Le domaine du traitement du langage écrit bénéficie grandement d'approches multilingues [DEVLIN et al. 2019 ; CONNEAU, KHANDELWAL et al. 2020 ; XUE et al. 2020], y compris dans un contexte de compréhension du langage [CONNEAU, LAMPLE et al. 2018 ; RUDER et al. 2021]. Ces avancées ont ouvert notamment la voie au traitement de langues peu dotées via l'utilisation de grandes quantités de données externes. Parmi les modèles proposés, LaBSE [FENG et al. 2022] se démarque par son agnosticisme à la langue. Son architecture à l'état-de-l'art mais aussi sa grande couverture linguistique lui permettent de généraliser ses compétences d'extraction sémantique à des langues n'ayant jamais été traitées durant son apprentissage.

SAMU-XLSR fut initialement proposé par Khurana et al. [2022] pour une tâche de recherche d'information vocale et traduction de la parole. Cet encodeur affine XLS-R grâce aux représen-

tations sémantiques de l'encodeur textuel LaBSE, le rendant intrinsèquement agnostique à la langue. Son enrichissement sémantique est analysé en profondeur dans cette thèse puis adapté à des tâches d'extraction sémantique complexes pour la compréhension de la parole. L'encodeur de parole SONAR [DUQUENNE, SCHWENK et al. 2023] tire lui aussi parti d'un apprentissage multimodal et multilingue.

Un modèle cross-lingue vise la projection dans un espace vectoriel proche de mots sémantiquement identiques bien qu'issus de langues différentes [CONNEAU et LAMPLE 2019]. Tandis que la compréhension de la parole se tournait déjà vers le cross-linguisme pour des systèmes non-neuronaux [KOMATANI et al. 2001 ; MENG et SIU 2002 ; HAHN et al. 2011], l'élan du multilinguisme pour une reconnaissance neuronale de la parole [H. LIN et al. 2009 ; Z. WANG et al. 2002 ; FÉR et al. 2017] incite le développement de modèles auto-supervisés multilingues [KAWAKAMI et al. 2020] et cross-lingues [CONNEAU, BAEVSKI et al. 2020]. Les premiers systèmes de bout-en-bout multilingues pour la compréhension de la parole ont par la suite été proposés [MÜLLER et al. 2021 ; X. ZHANG et L. HE 2021].

La portabilité d'un système peut se définir par sa capacité à être pré-appris puis adapté à une tâche, un domaine, ou une langue différente. Comme pour le multilinguisme et le cross-linguisme, la portabilité des langues était d'abord explorée pour la reconnaissance de la parole [SARIKAYA 2008 ; LEFÈVRE, GAUVAIN et al. 2005] et proposée à des fins d'amélioration d'une tâche monolingue [J. HUANG et al. 2013] ou multilingue [VU et al. 2014]. Dans le domaine de la compréhension de la parole, la portabilité des langues s'est tout d'abord développée pour des systèmes en cascade [SERVAN et al. 2010 ; JABAIAN et al. 2013] avant de se tourner vers des systèmes de bout-en-bout [TOMASHENKO et al. 2019 ; LUGOSCH, RAVANELLI et al. 2019 ; SCHUSTER et al. 2018]. Ceux-ci avaient principalement pour objectif le traitement de données monolingues [BHOSALE et al. 2019 ; R. PRICE 2020], souvent pour une langue peu dotée [W. CHEN et al. 2018 ; JIA et al. 2020].

Motivations

Cette thèse vise l'extraction de concepts sémantiques depuis un signal de parole. Ceux-ci sont pré-définis pour une tâche précise, adaptés autant à son domaine qu'à son objectif. Les ensembles de données MEDIA [BONNEAU-MAYNARD, ROSSET et al. 2005] et PortMEDIA [LEFÈVRE, MOSTEFA et al. 2012] utilisent un formalisme d'annotation complexe, avec une représentation sémantique de haut niveau considérée comme l'une des plus riches à traiter [BÉCHET et RAYMOND 2019]. Bien que la communauté scientifique se focalise généralement sur le traitement de sa version *relax* [HAHN et al. 2011 ; VUKOTIC et al. 2015 ; DINARELLI et TELLIER 2016], cette thèse prend le parti de se concentrer sur sa version *full* disposant d'annotations sémantiques plus détaillées et étant l'unique version disponible pour les données italiennes PortMEDIA.

Bien que faisant partie des très rares ensembles de données disponibles pour une tâche d'extraction sémantique depuis la parole, MEDIA est rarement utilisé au-delà de la communauté scientifique française. Nous décrivons dans cette thèse une recette liant préparation des données et apprentissage de bout-en-bout [LAPERRIÈRE, PELLOIN, CAUBRIERE et al. 2022b ; LAPERRIÈRE, PELLOIN, CAUBRIERE et al. 2022a], permettant de remettre en lumière cet ensemble de données dans l'outil open-source et dûment maintenu SpeechBrain [RAVANELLI, PARCOLLET et al. 2021].

L'extraction de concepts sémantiques peut être perçue comme une tâche de traduction depuis une langue source naturelle vers une langue cible conceptuelle. C'est de cette idée que part l'intention d'utiliser des systèmes à l'état-de-l'art dans le domaine de la traduction automatique afin de viser la compréhension de la parole. Les encodeurs de parole auto-supervisés sont depuis quelques années de plus en plus présents dans ces deux domaines, pouvant bénéficier d'un apport multi-modal significatif grâce à l'utilisation d'encodeurs textuels multilingues performants, comme c'est le cas de SAMU-XLSR [KHURANA et al. 2022].

Cette thèse étudie l'enrichissement sémantique apporté par son apprentissage avant d'en proposer diverses spécialisations pour une tâche complexe de compréhension de la parole. Nous nous intéressons entre autres à la réalisation d'une double spécialisation sémantique, conjointe à notre tâche d'extraction sémantique. En réunissant ces deux apprentissages en un unique modèle, nous souhaitons :

— orienter l'encodage vers le domaine sémantique visé ;
— préserver la faculté de l'encodeur à générer certaines abstractions sémantiques ;
— limiter la perte de ses capacités cross-lingues ;

Tandis que les tâches d'extraction sémantique utilisent généralement des encodages de la parole d'un niveau de granularité de l'ordre de 20 millisecondes, l'encodeur de parole SAMU-XLSR génère également une représentation sémantique sous la forme d'un vecteur continu pour chaque segment de parole. Cette thèse explore l'utilisation d'une telle représentation, cherchant à tirer profit d'éventuelles informations pertinentes à notre tâche d'extraction sémantique, que ce soit par la contextualisation ou les abstractions sémantiques pouvant être apportées par ce niveau de représentation.

L'accessibilité aux nouvelles technologies est essentielle pour assurer une disponibilité équitable de l'information et des services à travers le globe. Il est donc important de ne pas limiter ces technologies à certaines langues et domaines mais de proposer des systèmes performants pour un maximum de d'entre eux. Cet objectif se heurte à des restrictions en terme de quantités de données disponibles, nécessaires pour le développement de systèmes de traitement de la parole [BASTIANELLI et al. 2020]. Une solution réside dans l'utilisation d'ensembles de données issus de langues et domaines différents de la tâche ciblée.

De nos jours, de nombreux modèles tirent leur efficacité de l'utilisation d'importantes quantités de données dans les langues pour lesquelles ils visent le traitement. Pour autant, ce besoin de larges ensembles de données mène à des coûts humains et matériels non négligeables [WAIBEL, SCHULTZ et al. 2004 ; SCHULTZ et BLACK 2006]. Nous étudions dans cette thèse l'apport du multilinguisme pour l'extraction sémantique d'ensembles d'ensembles de données peu dotés à travers nos différentes méthodes de spécialisation de SAMU-XLSR. Nous considérons des langues proches comme le français avec MEDIA et l'italien avec PortMEDIA-it puis une langue plus distante telle que le tunisien avec TARIC-SLU [MDHAFFAR, BOUGARES et al. 2024]. L'intérêt principal de l'utilisation de données multilingues réside dans la prise en charge de langues et dialectes pour lesquels très peu de données annotées sont disponibles [SAMSON JUAN 2015]. SAMU-XLSR n'ayant pas été appris sur des données tunisiennes, nos expérimentations avec TARIC-SLU visent à déterminer sa capacité à traiter de nouvelles langues.

Enfin, nous analysons l'impact de la spécialisation de SAMU-XLSR sur le traitement d'ensembles de données de domaines plus ou moins distants. Il s'agit ici de constater l'évolution des capacités d'encodage linguistique et sémantique de SAMU-XLSR pour le traitement de domaines considérés lors de son premier pré-apprentissage mais aussi de nouveaux domaines jamais considérés jusqu'alors.

Contexte de la thèse

Cette thèse s'inscrit dans le cadre du projet européen SELMA (accord de subvention numéro 957017), financé par la Commission Européenne grâce au programme de recherche et d'innovation Horizon 2020.

Le consortium multinational SELMA a débuté le premier janvier 2021 pour une date de clôture initialement prévue le 31 décembre 2023 mais repoussée à mars 2024. Les partenaires de ce projet étaient la Deutsche Welle, l'University of Latvia avec l'Institute of Mathematics and Computer Science (IMCS), Priberam, Fraunhofer-Gesellschaft (FhG) et Avignon Université avec le Laboratoire Informatique d'Avignon (LIA).

L'objectif du projet SELMA était de proposer une plate-forme clefs en main et open-source pour aider les divers corps de métiers journalistiques à analyser et traiter de vastes flux de contenu audiovisuel. À cette aide s'ajoutait l'intention d'enrichir leurs productions grâce à des outils de transcription, traduction, doublage et sous-titrage afin de les rendre le plus accessible possible. Tout cela aura été en grande partie mis en place grâce à des solutions d'Apprentissage Profond.

Cette thèse aura participé aussi plus brièvement lors de l'atelier JSALT 2022 au projet européen ESPERANTO (accord de subvention numéro 101007666), financé par le programme de bourses européennes Marie Skodowska-Curie et coordonné par l'université du Mans, notamment son laboratoire d'informatique (LIUM). Il débuta le 1er janvier 2020 pour une clôture prévue de 31 décembre 2025.

ESPERANTO est un programme de recherche incluant un total de 20 partenaires provenant de 11 pays sur 4 continents. Centré sur le Traitement Automatique de la Parole, ce projet vise le développement de systèmes explicables pour le domaine de l'interraction humain-machine avec leur diffusion au plus grand nombre à travers, entre autres, leur prise en charge de dialectes et langues rares.

Structure du manuscrit

Ce document est divisé en deux parties.

La première partie met en contexte cette thèse à travers trois chapitres. Ceux-ci présentent le domaine de la compréhension de la parole et l'évolution de ses outils à travers ces dernières décennies, en insistant sur les technologies neuronales à l'état-de-l'art pour l'encodage de la parole.

La seconde partie présente les contributions apportées par cette thèse concernant le traitement et la diffusion d'un ensemble de données pour une tâche complexe d'extraction sémantique ainsi que nos travaux pour la spécialisation de l'enrichissement sémantique d'un système de compréhension de la parole dans un contexte multilingue.

Le **premier chapitre** donne une vue d'ensemble de l'Apprentissage Profond pour le Traitement Automatique du Langage Naturel, notamment celui de la parole, en décrivant en profondeur les architectures neuronales et techniques d'apprentissage à l'état-de-l'art dans ce domaine.

Pour ce faire, les deux premières sections posent les bases théoriques du fonctionnement des réseaux de neurones artificiels. S'en suit la description d'architectures neuronales plus complexes, telles que les couches récurrentes et réseaux Transformer utilisés durant cette thèse. Nous abordons ensuite le principe d'apprentissage auto-supervisé, qui sera approfondi dans notre troisième chapitre à travers une description détaillée d'encodeurs de parole l'utilisant. Ce chapitre présente également différentes méthodes d'optimisation utilisées pour nos expérimentations, telles que certaines fonctions de coût, optimiseurs et algorithmes de recherche, tout en mettant en avant les principes de pré-apprentissage et affinage de réseaux de neurones.

Le **second chapitre** donne un aperçu du domaine de la Compréhension Automatique de la Parole et définit l'apport de données multilingues pour ce domaine.

Une première section détaille diverses tâches du domaine et méthodes d'annotation textuelle pour l'extraction de sémantique.

Nous poursuivons ce chapitre avec deux sections présentant les architectures en cascade et architectures de bout-en-bout, mettant en avant la volonté de la communauté scientifique à se tourner vers ces dernières. Après avoir rappelé les différences, avantages et inconvénients de ces deux types d'architecture, nous décrivons l'évolution des systèmes en cascade à travers les techniques employées pour leur module de traitement de la parole et de compréhension du

language écrit. Nous faisons de même pour les architectures de bout-en-bout pour leur unique module de compréhension directe depuis la parole.

Enfin, nous faisons le lien entre le domaine de la Traduction Automatique et les avancées multilingues et cross-lingues de ces dernières années pour celui de la Compréhension Automatique, notamment de par l'utilisation de cross-modalités textuelles de plus en plus performantes.

Le **troisième chapitre** présente diverses méthodes d'encodage de la parole, dont sa paramétrisation acoustique en spectrogramme et autres coefficients et son apprentissage neuronal via des encodeurs de parole monolingues, multilingues et ayant été enrichis sémantiquement. Ce sont ces derniers qui seront principalement étudiés dans les chapitres suivants pour la résolution d'une tâche complexe d'extraction sémantique depuis la parole, bien qu'initialement proposés pour des tâches de traduction de la parole ou recherche d'information vocale. La description de leur fonctionnement et phases d'apprentissage se fera après celle détaillée du modèle wav2vec 2.0, base de nombreux encodeurs à l'état-de-l'art pour le domaine de la compréhension de la parole.

Le **quatrième chapitre** présente nos premières contributions concernant l'ensemble de données françaises MEDIA à travers une recette complète retravaillant entre autres la segmentation de ses échantillons. Celle-ci est diffusée actuellement grâce à l'outil SpeechBrain, dont la popularité croît depuis quelques années pour le développement d'architectures neuronales liées au traitement du langage naturel. Cette recette est amenée avec une discussion autour des métriques communément utilisées pour MEDIA, notamment concernant la normalisation imparfaite souvent réalisée pour un de ses taux d'erreur.

Ce chapitre décrit en détail l'ontologie complexe de MEDIA et donne des statistiques précises sur ses segments et son lexique. Nous insistons sur notre volonté d'utiliser sa version *full* composée d'annotations plus complexes que la version *relax* communément utilisée par la communauté.

Deux autres ensembles de données associés à MEDIA sont présentés. PortMEDIA-it est utilisé tout au long de nos chapitres suivants en tant qu'ensemble de données peu doté pour des expérimentations cross-lingues principalement entre le français et l'italien. PortMEDIA-fr est utilisé plus brièvement afin d'évaluer la portabilité entre domaines proches de certains de nos systèmes.

Le **cinquième chapitre** présente nos contributions concernant l'étude de l'enrichissement sémantique d'un encodeur de parole appliquée à une tâche complexe d'extraction sémantique. Nous y réalisons une analyse poussée des représentations intermédiaires des modèle SAMU-XLSR et XLS-R, tant au niveau de leur encodage sémantique que linguistique, et étudions l'apport possible de la représentation sémantique contextuelle générée par SAMU-XLSR. Nous indiquons par ailleurs nos mesures de confiance pour nos expérimentations sur MEDIA, PortMEDIA-it et TARIC-SLU, un ensemble de données tunisien présenté dans ce chapitre.

Cette étude se poursuit par la spécialisation de l'enrichissement sémantique de SAMU-XLSR sur les transcriptions brutes et enregistrements audio de ces trois ensembles de données. Cette

spécialisation y est tout d'abord proposée comme première étape avant un second apprentissage pour la résolution de nos tâches cibles. Nous détaillons ensuite une approche de double spécialisation visant à fusionner cette spécialisation jusqu'alors indépendante à notre module d'extraction sémantique.

Le **sixième chapitre** apporte une dimension multilingue aux expérimentations menées au chapitre précédent à travers des apprentissages cross-lingues entre langues distantes telles que le tunisien face au français et à l'italien. Des expérimentations de portabilité cross-lingue sont menées entre langues plus proches, du français vers l'italien.

En reprenant les propositions architecturales du cinquième chapitre, le sixième chapitre présente les bénéfices d'une spécialisation multilingue de l'enrichissement sémantique de SAMU-XLSR, simple ou double, tout en conduisant une fois de plus une analyse poussée de son encodage intermédiaire et de ses représentations sémantiques contextuelles.

Enfin, une dernière section présente nos analyses concernant la portabilité cross-domaine de nos systèmes suite à de tels apprentissages afin de constater l'évolution des capacités d'encodage linguistique et sémantique de SAMU-XLSR. Celles-ci sont réalisées avec les données italiennes CommonVoice pour un domaine très éloigné de PortMEDIA-it et françaises PortMEDIA-fr pour un domaine proche de MEDIA.

En **annexes** sont résumées nos premières expérimentations sur MEDIA pour des systèmes en cascade, ainsi que notre collaboration pour l'ajout d'annotations d'intention à l'ensemble de données MEDIA, avec l'analyse de l'apport commun de ses deux tâches lorsque apprises dans un unique système. Celles-ci seront précédées de la conclusion de ce manuscrit ainsi que de quelques perspectives pour de futurs travaux de recherche.

Première partie

Contexte

Chapitre 1

Réseaux de neurones artificiels

Sommaire

Bien que le terme «Intelligence Artificielle» n'ait vu le jour qu'en 1956 avec John McCarthy, c'est dans les années 1930, avec Alonzo Church et Alan Turing, que naissent les modèles de calculabilité, prémisses des réseaux de neurones que l'on connaît aujourd'hui. À travers la thèse de Church-Turing, l'Intelligence Artificielle se dessine par la notion que tout raisonnement mathématique logique peut être calculé à partir de règles formelles.

De nos jours, l'Intelligence Artificielle est partie intégrante de notre quotidien. Elle est utilisée dans de multiples domaines d'activité tels que la finance, les transports, l'industrie ou encore la santé. De nombreuses nouvelles technologies utilisent ses algorithmes complexes afin d'optimiser leurs processus et résultats, dans le but de proposer des solutions toujours plus pertinentes et innovantes face aux enjeux actuels et futurs de notre société. L'Intelligence Artificielle (*Artificial Intelligence, AI*) rassemble les sciences mathématiques, logiques, probabilistes et statistiques afin de permettre à une machine de prendre des décisions face à un évènement nouveau. On peut citer parmi les capacités visées celles de communiquer, prendre une décision, ou encore créer du contenu.

Pour ce faire, le domaine de l'Intelligence Artificielle réunit de nombreuses méthodes, y compris l'apprentissage par réseaux de neurones artificiels dont nous allons parler dans ce chapitre. Ces algorithmes font partie des méthodes d'Apprentissage Profonds, elles-même appartenant au domaine de l'Apprentissage Automatique. Le diagramme présenté par la Figure 1.1 nous permet de visualiser cette imbrication.

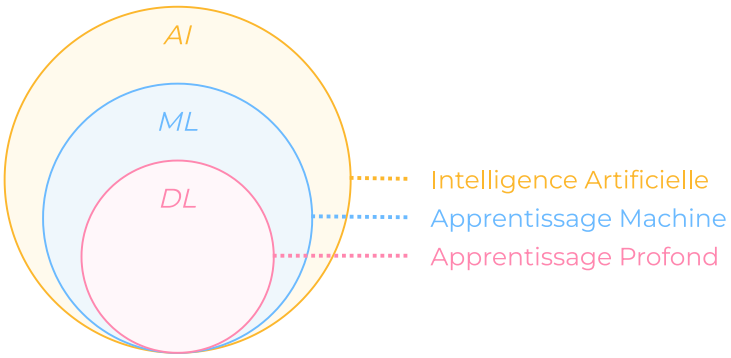

FIGURE 1.1 – Visualisation de l'imbrication du domaine de l'Apprentissage Profond dans l'Apprentissage Supervisé et l'Intelligence Artificielle.

L'Apprentissage Automatique (*Machine Learning, ML*) regroupe les systèmes programmés pour extraire les informations pertinentes d'un ensemble de données d'entraînement, afin de résoudre une tâche précise. Ces systèmes s'améliorent durant leur processus d'apprentissage sans être guidés explicitement par des règles humaines. Leur fonctionnement réside en l'optimisation des paramètres qui les composent, afin d'obtenir les meilleurs performances sur la tâche visée.

On peut distinguer deux méthodes principales d'Apprentissage Automatique : l'apprentissage supervisé et l'apprentissage auto-supervisé. Ces méthodes se distinguent par leurs données d'apprentissage et le traitement de celles-ci.

Les méthodes d'apprentissage supervisé nécessitent des données d'entrée annotées. L'objectif du modèle est de retrouver l'étiquette de l'annotation correspondante à chaque échantillon de données, ce même pour de nouvelles données ne faisant pas partie du corpus d'apprentissage. Il existe deux catégories de tâches liées à l'apprentissage supervisé. La première, la classification, consiste à prédire à quelles classes appartient une donnée d'entrée, en s'appuyant sur un étiquetage discret. La seconde, la régression, permet de prédire des sorties sous forme de valeurs numériques continues.

Les méthodes d'apprentissage non-supervisé sont utilisées pour traiter des données non-étiquetées. On ne pré-détermine alors pas quel résultat est attendu en sortie du modèle. Ce type d'apprentissage est généralement utilisé pour classifier les données en groupes d'individus ayant des caractéristiques communes.

L'Apprentissage Profond (*Deep Learning, DL*), sous-catégorie de l'Apprentissage Automatique, tire sa singularité de part sa capacité à générer des représentations abstraites. Sa profondeur vient du nombre conséquent de représentations intermédiaires utilisées. L'Apprentissage Profond utilise pour cela des réseaux de neurones artificiels, ayant permis d'importants progrès dans de nombreux domaines, notamment pour le traitement de données textuelles, sonores et visuelles.

Ce chapitre met en contexte le fonctionnement de ces réseaux de neurones artificiels tout en décrivant les diverses architectures neuronales utilisées durant cette thèse.

1.1 Architectures neuronales simples

Les premiers modèles de neurones formels furent mis au point en 1943 par Mc-Culloch et Pitts. Ce nouveau modèle mathématique s'appuie dès lors sur le fonctionnement des neurones biologiques.

1.1.1 Perceptron

En 1958, Rosenblatt inventa le Perceptron, un algorithme d'apprentissage applicable à un neurone formel. Le perceptron est la forme la plus simple de réseau de neurones artificiel, ne contenant qu'un unique neurone réalisant une classification binaire. Il ne résout donc que des problèmes linéairement séparables.

La Figure 1.2 met en évidence le fonctionnement de ce modèle neuronal. On peut y retrouver :

— les **entrées** ($X = x_1, \ldots, x_n$) pouvant être réelles ou binaires
— les **poids** ($W = w_1, \ldots, w_n$) ajustant l'importance relative à chaque entrée pour réaliser la classification
— le **biais** (b) permettant d'ajuster la frontière décisionnelle du modèle
— la **fonction d'activation** (σ) appliquée aux entrées et au biais
— la **sortie** ($\hat{y}$) prédite par le système, déterminant la classe de l'échantillon

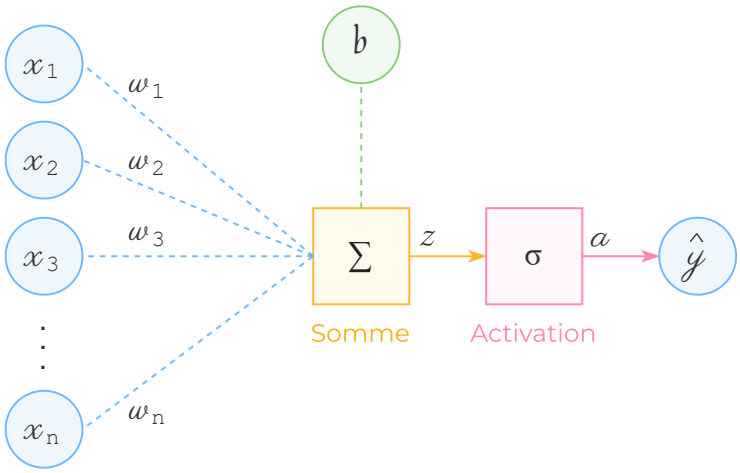

FIGURE 1.2 – Schématisation du fonctionnement d'un perceptron.

Un perceptron prend un nombre défini de n valeurs d'entrée qui seront multipliées par leur poids respectif puis sommées entre elles, avec soustraction du biais, formant autrement dit une somme pondérée telle que :

$$\widehat{y} = \sigma(\sum_{i=1}^{n} x_i w_i - b) \tag{1.1}$$

Le nombre résultant passe ensuite par une fonction d'activation, prédisant la classe de l'échantillon. Cette classe est donc directement dépendante des seuils d'activation de la fonction choisie. Durant l'apprentissage du modèle, les paramètres – poids et biais – sont mis à jours pour rapprocher au mieux la sortie prédite de celle attendue.

1.1.2 Perceptron multicouche

Afin de résoudre des problèmes non-linéaires, le perceptron multicouche (*Multilayer Perceptron, MLP*) fait son apparition en 1969 par Minsky et Papert. Le perceptron simple étant utilisé pour résoudre des problèmes strictement linéairement séparables, une mise en réseau de plusieurs perceptrons a été imaginée pour surmonter cette limitation.

Un perceptron multicouche est donc un réseau de plusieurs neurones artificiels, ordonnés par couches. Il peut être considéré comme le plus simple des réseaux de neurones profonds, sa profondeur étant directement liée au nombre de ses couches cachées dites linéaires.

Les liaisons entre neurones se font de la même manière que celles des n entrées d'un perceptron, par sommes pondérées, une sortie $\widehat{y}$ pouvant être utilisée comme entrée des neurones de la couche suivante. Les couches situées entre la couche d'entrée et celle de sortie sont appelées couches cachées. Elles disposent chacune d'un nombre de neurones indépendant, optimisé pour la tâche. L'information circulera de la couche d'entrée vers la couche de sortie, autrement dit par propagation vers l'avant (*feed forward*). Les fonctions d'activation sont alors utilisées à chaque neurone, afin de propager sa sortie vers la couche suivante. La Figure 1.3 schématise ce type de réseaux de neurones.

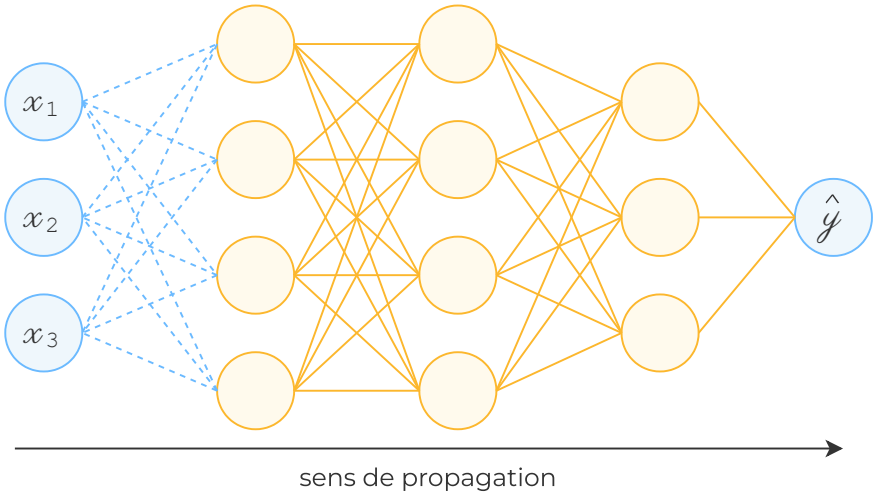

FIGURE 1.3 – Schématisation d'un réseau de neurones simple, appelé perceptron multicouche.

Une fonction d'activation σ transforme les sorties du modèle, potentiellement comprises dans l'intervalle $[-\infty; +\infty]$, vers un intervalle adapté à la tâche. Parmi les fonctions d'activations courantes, nous pouvons citer la Softmax qui réalise une approximation lissée de la fonction arithmétique Argmax, contraignant la prédiction à un intervalle $[0; 1]$. La fonction d'activation sigmoïde (*Sigmoid*) permet aussi d'obtenir des sorties dans ce même intervalle, à la différence qu'elle réalise une répartition logistique. Ses sorties sont donc similaires à des probabilités distribuées parmi l'ensemble des prédictions du modèle, faisant d'elle une fonction d'activation principalement utilisée en sortie de modèles plus complexes. L'activation par tangente hyperbolique (*tanh*), elle, réduit la valeur des données à l'intervalle $[-1; 1]$. Enfin, afin de pallier les problèmes liés à la profondeur grandissante des réseaux de neurones comme celui de disparition du gradient (*vanishing gradient*), Nair et Hinton [2010] ont proposé la fonction non-linaire de redressement (*ReLU*). Celle-ci remplace toute sortie négative par zéro.

1.2 Apprentissage

Afin de réaliser un apprentissage, l'ensemble de données disponible est divisé en trois parties distinctes : un corpus d'entraînement, un corpus de développement et un corpus de test (*train*, *dev*, *test*).

Une époque consiste en de nombreuses itérations. Une itération équivaut à l'utilisation d'une sous-partie des données d'apprentissage et la mise à jour des paramètres du modèle. Lorsque les paramètres sont mis à jour après avoir vu l'ensemble des données d'apprentissage, on appellera cela une descente de gradient par lot (*batch*). Cette méthode réalise une moyenne des gradients du lot, augmentant considérablement la vitesse de convergence de la fonction de coût vis-à-vis d'un passage échantillon par échantillon. La plus utilisée reste la descente de gradient par mini-lot (*mini-batch*), mélange des deux précédentes méthodes, réalisée après le passage d'un nombre définit d'échantillons. Chaque époque se termine par une inférence réalisée sur le corpus de développement. À la fin de l'apprentissage, après avoir réalisé un certain nombre d'époques, on pourra tester le modèle en réalisant une inférence sur le corpus de test.

L'apprentissage consiste à ajuster les paramètres du modèle pour minimiser la valeur obtenue par une fonction de coût (*loss*) adaptée à la tâche. En apprentissage supervisé, le fait de connaître les sorties attendues nous sert à calculer cette fonction de coût. Le résultat de ce calcul permettra de réaliser un algorithme d'optimisation nommé descente de gradient afin de mettre à jour les paramètres du modèle à chaque itération de l'apprentissage.

Un apprentissage concluant passe aussi par les choix techniques de nombreux hyper-paramètres avant apprentissage, tels que celui de l'Optimiseur d'apprentissage que nous détaillerons plus loin.

1.2.1 Fonctions de coût

La fonction de coût C permet de quantifier l'écart entre les prédictions du modèle et les sorties attendues. Comme pour tout autre hyper-paramètre d'un réseau de neurones, il est important d'utiliser une fonction de coût adaptée à la tâche visée. Pour réaliser la descente de gradient, la fonction de coût doit être dérivable.

Dans cette section, nous aborderons tout particulièrement les fonctions de coût de Classification Temporelle Connexionniste et de Similarité Cosinus car utilisées principalement durant cette thèse.

Classification Temporelle Connexionniste

La Classification Temporelle Connexionniste (*Connectionist Temporal Classification, CTC*), introduite par Graves et al. [2006], est une fonction de coût communément utilisée pour des tâches

séquence-vers-séquence, notamment pour le traitement de la parole. Elle permet à un réseau neuronal d'apprendre l'alignement entre deux séquences comme suit :

$$C_{CTC} = -log\ P(\widehat{Y}|X) \tag{1.2}$$

Avec $P(\widehat{Y}|X)$ la probabilité d'une séquence de sortie $\widehat{Y}$, sachant la séquence d'entrée X, calculée en parcourant les alignements A possibles entre $\widehat{Y}$ et X à tout instant t :

$$P(\widehat{Y}|X) = \sum_{a \in A} \prod_{t=1}^{T} P(a_t|X) \tag{1.3}$$

L'intérêt principal de cette fonction de coût réside dans sa faculté à traiter des séquences à tailles variables et distinctes. De ce fait, elle est particulièrement présente dans le domaine du Traitement Automatique du Langage Naturel.

Pour le traitement de la parole, elle permet d'apprendre à réaliser l'alignement entre un segment audio et sa transcription prédite. Elle peut réaliser cet alignement au niveau des caractères ou phonèmes d'une phrase, comme réalisé par Fernández et al. [2008]. Pour ce faire, après avoir segmenté en trames très fines les entrées audio et extrait leurs paramètres acoustiques, on peut réaliser un apprentissage neuronal qui prédira une distribution de probabilité pour l'ensemble du vocabulaire pré-défini. Ces distributions de probabilité permettront, grâce à l'utilisation de la CTC, de construire une séquence de sorties prédites pour un ensemble de trames.

Un autre intérêt pour le traitement de la parole est la gestion de la vitesse d'élocution de l'enregistrement audio fourni en entrée du modèle. En effet, une trame audio (*frame*) ne correspond pas toujours strictement à un unique caractère ou phonème. Ces segments de parole sont justement choisis très courts, souvent 20ms, afin de pouvoir prédire tous les éléments attendus. Cela signifie qu'une séquence peut être prédite comme suit : $[h\ h\ e\ l\ l\ l\ o\ o]$. Afin de corriger les répétitions, la CTC supprime tous les caractères identiques consécutifs, produisant : $[h\ e\ l\ o]$. Pour pouvoir préserver les caractères consécutifs souhaités, elle introduit donc un nouvel élément au vocabulaire : ϵ. Prédit entre deux éléments identiques, il permet de les préserver. Ainsi, il sera possible pour le modèle de prédire $[h\ h\ e\ l\ l\ \epsilon\ l\ o\ o]$, raccourcit par la CTC en $[h\ e\ l\ l\ o]$ après suppression de ϵ. En revanche, un inconvénient à l'introduction de ϵ réside dans son positionnement possible entre deux caractères distincts, diminuant la précision de l'alignement entre le texte et la parole car ne permettant pas de distinguer où s'arrête la prononciation de l'un et où commence celle de l'autre. La Figure 1.4 illustre le fonctionnement de cette fonction de coût.

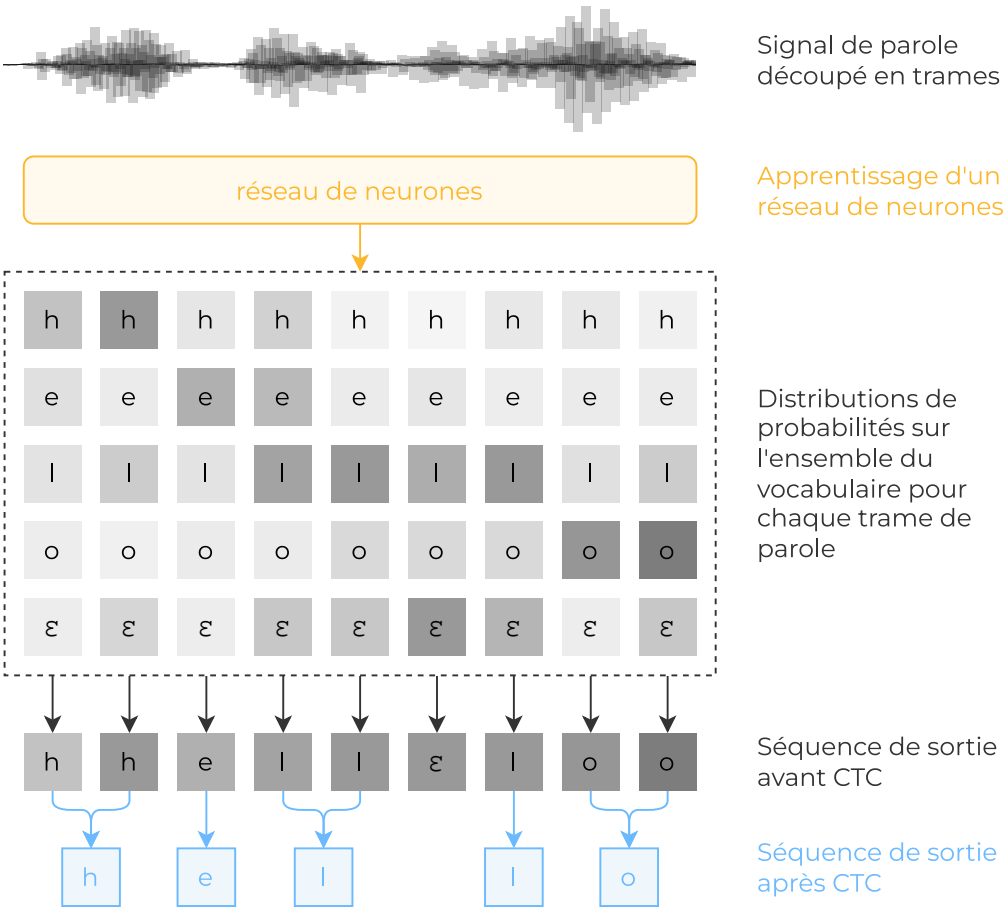

FIGURE 1.4 – Exemple de fonctionnement de la Classification Temporelle Connexionniste pour un segment de parole dont la sortie attendue est une séquence de caractères.

Similarité cosinus

La similarité cosinus (*cosine similarity*) mesure l'angle entre deux vecteurs avec leur cosinus. Principalement utilisée dans le domaine du langage écrit, que ce soit en recherche d'information ou classification, elle permet de mesurer la proximité entre deux représentations vectorielles. Pour une sortie attendue Y et une prédiction $\widehat{Y}$ de dimension N, on posera donc :

$$C_{CosineSimilarity} = \frac{Y \cdot \widehat{Y}}{||Y||\,||\widehat{Y}||} = \frac{\sum_{i=1}^{N} Y_i \widehat{Y}_i}{\sqrt{\sum_{i=1}^{N} Y_i^2}\sqrt{\sum_{i=1}^{N} \widehat{Y}_i^2}} \tag{1.4}$$

Son résultat est compris dans l'intervalle $[-1, 1]$. Une similarité cosinus de -1 indique des vecteurs diamétralement opposés. Une valeur de 0 signifie que ceux-ci sont indépendants, appelés aussi vecteurs orthogonaux. Enfin, une valeur positive proche de 1 démontre leur colinéarité unidirectionnelle.

Fonctions de coût communes

Parmi les autres fonctions de coût existantes, nous pouvons citer l'Erreur Quadratique Moyenne (*Mean Squared Error, MSE*) et l'entropie croisée (*cross entropy*).

La MSE permet de résoudre une tâche de régression en mesurant la proximité, dans l'espace, des sorties attendues Y et de la prédiction associée $\widehat{Y}$, toutes deux de dimension N. Sa formule est la suivante :

$$C_{MSE} = \frac{1}{N}\sum_{i=1}^{N}(Y_i - \widehat{Y}_i)^2 \tag{1.5}$$

Elle peut être difficile à interpréter car son résultat peut théoriquement être compris dans l'intervalle $[-\infty; +\infty]$. De manière absolue, un résultat éloigné de 0 ne signifiera pas nécessairement une mauvaise prédiction, car son interprétation dépendra de la nature des sorties attendues.

L'entropie croisée est utilisée pour des tâches de classification. Elle s'écrit :

$$C_{CrossEntropy} = -\sum_{i=1}^{N} P(Y_i)\, log\, P(\widehat{Y}_i) \tag{1.6}$$

En considérant une distribution de probabilités attendue en sortie du modèle appris, on notera $P(Y_{ij})$ la probabilité attendue pour un élément i et une classe j parmi N classes possibles, et $P(\widehat{Y}_{ij})$ la probabilité prédite par le modèle.

Il existe de nombreuses variantes de cette fonction de coût dont :

— l'**entropie croisée catégorique** (*Categorical Cross Entropy*) qui réalise une classification multi-classes avec une unique sortie binaire positive.

— l'**entropie croisée binaire** (*Binary Cross Entropy, BCE*) permettant une prédiction multi-classes avec l'aide d'un vecteur binaire de type *multi-hot*.

— l'**entropie croisée pondérée** (*Weighted Cross Entropy*) rectifiant la faiblesse de la BCE pour le traitement de données d'apprentissage mal équilibrées en pondérant les échantillons positifs.

— l'**entropie croisée équilibrée** (*Balanced Cross Entropy*) ajoutant à la précédente une pénalité sur les échantillons négatifs.

1.2.2 Descente de gradient

La descente de gradient est un algorithme d'optimisation permettant de modifier les paramètres du modèle pour minimiser sa fonction de coût et maximiser les performances sur la tâche ciblée. Il agit à la fin de chaque itération d'apprentissage, après calcul de la fonction de coût, pendant ce que l'on appelle la phase de rétro-propagation. On parle alors de convergence lorsque la valeur des paramètres du modèle ne change plus significativement entre deux itérations successives. Un

minimum local ou global est alors atteint pour la fonction de coût. Cette méthode d'optimisation fut utilisée par Rumelhart et al. [1986] pour l'apprentissage de réseaux de neurones.

Le gradient, une représentation vectorielle que l'on notera ∇W pour les poids et ∇B pour les biais, est composé des dérivées partielles de la fonction de coût C, en fonction des paramètres W et B du modèle :

$$\nabla W = \frac{\partial C}{\partial W} \qquad \nabla B = \frac{\partial C}{\partial B} \tag{1.7}$$

Il représente la direction de la plus forte croissance de coût pour chaque paramètre. La descente de gradient repose donc sur le fait de modifier les paramètres dans le sens inverse de leur gradient.

Pour cela, un taux d'apprentissage (*learning-rate*) λ permet de réguler la vitesse d'ajustement des paramètres du modèle. Ainsi, à la prochaine itération de l'apprentissage $t+1$, les nouveaux paramètres du modèle seront mis à jour comme suit :

$$\begin{aligned} W_{t+1} &= W_t - \lambda \times \nabla W_t \\ B_{t+1} &= B_t - \lambda \times \nabla B_t \end{aligned} \tag{1.8}$$

Taux d'apprentissage et Momentum

Un des inconvénients de la descente de gradient réside dans le fait de pouvoir converger vers un minimum local de sa fonction de coût, impliquant des performances sous-optimales en fin d'apprentissage. En effet, celle-ci détient généralement plusieurs minima locaux qu'il est parfois complexe d'éviter. Lorsque bien choisi, le taux d'apprentissage permet à la fonction de coût de converger plus aisément vers son minimum global. La Figure 1.5 illustre le principe de minimum global et local ainsi que l'importance du taux d'apprentissage.

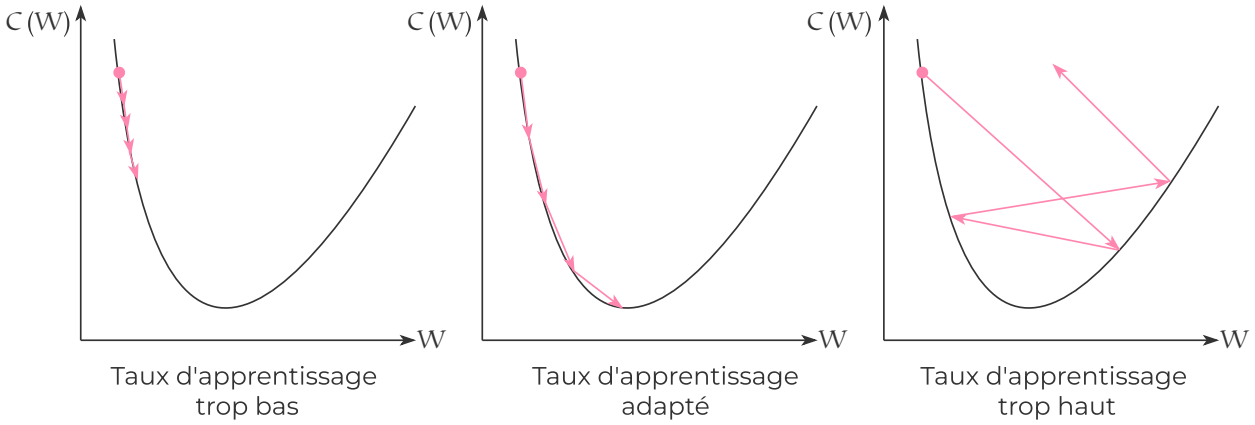

FIGURE 1.5 – Représentation de l'importance du taux d'apprentissage sur quatre époques.

Pour aider à optimiser le taux d'apprentissage, Qian introduit le momentum [1999], illustré par la Figure 1.6. Il permet d'ajuster la convergence des paramètres du modèle afin d'éviter au mieux un minimum local de sa fonction de coût tout en accélérant la descente de gradient.

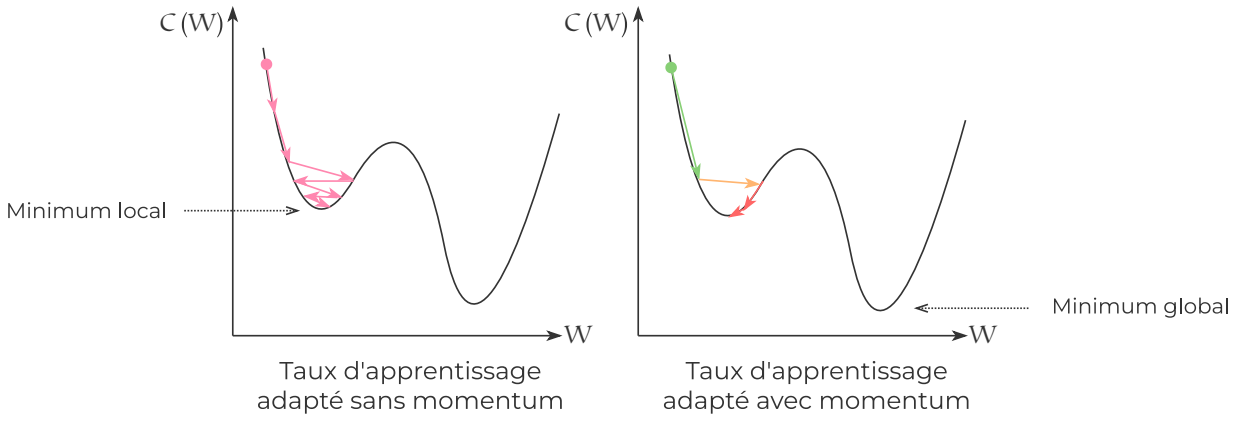

FIGURE 1.6 – Représentation du fonctionnement idéal d'un momentum.

Le momentum se traduit par un coefficient de vélocité qui régule le taux d'apprentissage. Lorsque les gradients à l'instant t et $t-1$ seront relativement orientés dans la même direction, il augmentera le taux d'apprentissage, permettant d'atteindre un minimum encore lointain. Lorsque les gradients à l'instant t et $t-1$ seront de direction relativement opposée, il réduira le taux d'apprentissage afin de tenter d'éviter les oscillations à l'approche d'un minimum, permettant à la convergence de se faire le plus précisément possible.

Lors de l'utilisation d'un momentum, la descente de gradient formulée par l'Équation (1.8) pour W_{t+1} et B_{t+1} se voit légèrement modifiée de la sorte :

$$W_{t+1} = W_t + \mu v - \lambda \times \nabla W_t \tag{1.9}$$

La vélocité v est initialisée à 0 et influencera directement la vitesse de descente du gradient. La constante μ, généralement comprise entre 0.1 et 0.9, est elle introduite afin de réguler à son tour la vélocité. Elle peut être programmée pour être modifiée durant l'apprentissage après un certain nombre d'itérations.

L'inconvénient principal de l'utilisation d'un momentum est son incapacité à prédire un changement de direction de gradient, et donc de réduire sa vélocité parfois trop importante pour atteindre un futur minimum local ou global. Le gradient accéléré de Nesterov [2013] propose une solution à ce problème, tout comme les algorithmes d'optimisation présentés en Section 1.2.3.

Il est important de noter que certaines tâches ont un minimum global difficile à atteindre. Il est donc nécessaire d'optimiser autant que possible tous les hyper-paramètres du modèle pour approximer au mieux le résultat escompté.

Rétro-Propagation du gradient

Enfin, nous pouvons introduire l'algorithme de rétro-propagation du gradient (*backpropagation*), dont l'engouement fut réellement déclenché par Rumelhart et al. en 1986, permettant d'optimiser le fonctionnement de la descente de gradient par chaînage de calculs. Celui-ci s'appuie sur le théorème de dérivation des fonctions composées en permettant un calcul plus efficace des gradients. Ce calcul se fait de la couche de sortie vers la couche d'entrée, chaque couche bénéficiant des gradients calculés pour la couche supérieure. Le gradient des unités neuronales d'une couche l se calcule donc avec celui des unités de la couche supérieure $l + 1$.

L'efficacité de l'algorithme de rétro-propagation du gradient réside en son calcul des dérivées partielles, maintenant uniquement dépendantes pour chaque couche de : son erreur δ, sa sortie brute nommée valeur d'agrégation z et sa sortie transformée nommée valeur d'activation a.

1.2.3 Optimiseurs

La descente de gradient stochastique (*SGD*) réalise un ajustement classique des paramètres après chaque passage d'un échantillon de données, avec un momentum. Réaliser une descente de gradient sur la totalité des données d'apprentissage une à une est cependant trop conséquent, d'autant plus considérant les larges corpus utilisés de nos jours. Pour cette raison, plusieurs variantes de descente de gradient ont vu le jour, appelées optimiseurs [RUDER 2016]. C'est le cas d'AdaGrad, AdaDelta et Adam, techniques doptimisation couramment utilisées que nous allons présenter plus en détail.

Comme leur nom l'indique, ces algorithmes sont utilisés pour optimiser la descente de gradient et donc l'apprentissage. Ils agissent ainsi en sur-couche aux autres hyper-paramètres afin d'adapter au mieux la mise à jour des paramètres du modèle. A l'instar du taux d'apprentissage, ces optimiseurs nous permettent également de réguler la vitesse de convergence du modèle.

AdaGrad

L'optimiseur AdaGrad (*Adaptative Gradient*) mis en place par Duchi et al. [2011] permet de réguler le taux d'apprentissage de chacun des paramètres du modèle en fonction de leur historique de modification par descente de gradient. De ce fait, l'ajustement d'un paramètre sera donc plus ou moins important en fonction de la fréquence et de l'impact de ses précédentes mises à jour. Le taux d'apprentissage est mis à jour au fur et à mesure de l'apprentissage, s'adaptant donc à la convergence du modèle.

L'intérêt de cet optimiseur semble mis en avant lors de l'apprentissage de données mal équilibrées, celui-ci tirant supposément mieux profit des échantillons moins représentés qu'une descente de gradient classique.

La mise à jour des poids présentée par l'Équation (1.8) se fait maintenant avec un taux d'apprentissage différent, ce pour chaque biais et poids comme suit :

$$w_{t+1} = w_t - \frac{\lambda}{\sqrt{h_{w_t} + \alpha}} \times \nabla w_t \tag{1.10}$$

On notera α une constante très faible empêchant une éventuelle division par 0 et h_{w_t} l'historique de mise à jour du poids w à l'instant t, initialisé à zéro pour $t = 0$, tel que :

$$h_{w_t} = h_{w_{t-1}} + \nabla w_t^2 \tag{1.11}$$

AdaGrad a ainsi l'inconvénient d'utiliser de manière répétitive le carré des gradients, pouvant rendre le taux d'apprentissage de certains paramètres extrêmement faible et ainsi limiter les capacités d'apprentissage du modèle.

Adam

Afin de pallier l'inconvénient principal de l'optimiseur AdaGrad, Kingma et Ba [2015] ont mis en place l'algorithme d'optimisation nommé Adam (*Adaptative Moment Estimation*). Cet optimiseur permet lui-aussi d'adapter le taux d'apprentissage spécifiquement à tout paramètre, en limitant l'utilisation d'un gradient au carré.

Avec la constante β_1 dans l'intervalle $[0; 1]$, conseillée 0.999, on calcule ainsi l'historique d'un paramètre, par exemple celui ici d'un poids w_t, comme suit :

$$h_{w_t} = \frac{\beta_1 \, h_{w_{t-1}} + (1 - \beta_1)\nabla w_t^2}{1 - \beta_1} \tag{1.12}$$

Adam modifie une fois de plus la mise à jour des paramètres du modèle en rectifiant le calcul des gradients de la Formule (1.10) :

$$w_{t+1} = w_t - \frac{\lambda}{\sqrt{h_{w_t} + \alpha}} \times m_{w_t} \tag{1.13}$$

Avec m_{w_t}, initialisée pour $t = 0$ à zéro, une accumulation de gradients utilisant une constante β_2 comprise dans le même intervalle que β_1, conseillée 0.9 :

$$m_{w_t} = \frac{\beta_2 \, m_{w_{t-1}} + (1 - \beta_2)\nabla w_t}{1 - \beta_2} \tag{1.14}$$

De nombreuses variantes d'Adam existent, telles que AdamW [Loshchilov et Hutter 2019], AdaMax [Kingma et Ba 2015] et NAdam [Dozat 2016], Adam restant le plus utilisé.

AdaDelta

L'optimiseur AdaDelta de Zeiler [2012] permet une optimisation de la descente de gradient en se basant sur l'optimiseur AdaGrad.

Afin de résoudre le problème de baisse importante et monotone du taux d'apprentissage d'AdaGrad, AdaDelta restreint l'accumulation d'historique des gradients en réalisant récursivement leur moyenne décroissante (*decaying average*). En faisant cela, l'algorithme permet d'accorder plus d'importance aux descentes de gradient récentes tout en ne préservant qu'une moyenne des gradients de chaque paramètre.

Avec a_{w_t}, initialisée à zéro pour $t = 0$, la moyenne décroissante des gradients de w à l'instant t remplaçant l'historique h_{w_t} de AdaGrad dans l'Équation (1.10), AdaDelta s'écrit :

$$w_{t+1} = w_t - \frac{\lambda}{\sqrt{a_{w_t} + \alpha}} \times \nabla w_t \tag{1.15}$$

De la même manière que pour l'historique d'Adam présenté par l'Équation (1.12), AdaDelta introduit une constante β, conseillée 0.9, dans le calcul de a_{w_t} :

$$a_{w_t} = \beta\, a_{w_{t-1}} + (1 - \beta)\nabla w_t^2 \tag{1.16}$$

1.2.4 Initialisation des paramètres

Afin d'optimiser le plus rapidement et efficacement les paramètres d'un réseau de neurones artificiel, il est important d'initialiser ceux-ci au mieux. Cette initialisation servira de point de départ à l'apprentissage et donc aux premières descentes de gradient.

Initialisation aléatoire

Il est courant d'utiliser une initialisation aléatoire des paramètres d'un modèle. Dans ce cas, on parlera de modèle appris de zéro (*from scratch*). L'inconvénient de cette méthode est qu'elle peut réduire l'efficacité de la descente de gradient en début d'apprentissage, voire orienter celle-ci vers un minimum local non-souhaité.

Initialisation de Xavier

L'initialisation de Xavier de Glorot et Bengio [2010] permet d'améliorer les premières descentes de gradient en limitant les valeurs initiales des biais et des poids à une moyenne de 0 ainsi qu'à une variance $\frac{1}{n_{l-1}}$ identique pour chaque couche de neurones, avec n le nombre de neurones de la couche considérée l. Les paramètres, tout en suivant ces deux conditions, sont initialisés aléatoirement en suivant une distribution normale.

Transfert d'apprentissage

Contrairement aux initialisations dites de zéro, le transfert d'apprentissage proposé par Pan et Yang [2010] repose sur un pré-apprentissage des paramètres, ensuite ré-utilisés pour l'initialisation de notre modèle. Ce pré-apprentissage est réalisé sur une tâche proche de celle visée. Le but est ainsi de tirer bénéfice du système pré-entraîné qui a supposément atteint une convergence stable de ses paramètres et a appris à capturer et traiter les informations pertinences pour une tâche similaire, accélérant la descente de gradient sur le modèle que souhaitons entraîner et allant jusqu'à améliorer ses résultats finaux.

On parle alors, pour ce modèle, d'affinage des paramètres ou de *fine-tuning*. Nous déciderons d'utiliser le terme plus commun *fine-tuning* et ses dérivées dans la suite de cette thèse.

Outre l'accélération conséquente de la convergence du modèle *fine-tuné*, ce type d'initialisation a d'autres avantages. Il permet notamment de réaliser l'apprentissage automatique d'une tâche contrainte par peu de données d'entraînement, grâce à un pré-apprentissage sur un nombre plus conséquent de données tierces proches.

1.3 Architectures neuronales avancées

Le perceptron et perceptron multicouche constituent une base solide pour de nombreux réseaux de neurones à l'architecture bien plus complexe. Afin de pouvoir traiter des données séquentielles, une fenêtre glissante peut être réalisée, mais elle ne permettrait pas de tirer partie de l'ensemble des informations contenues dans la séquence. C'est pourquoi des architectures neuronales plus complexes ont été proposées.

On définira une séquence comme une suite d'éléments inter-dépendants. Dans le domaine du Traitement Automatique du Langage Naturel abordé par cette thèse, les phrases sont considérées comme des séquences de caractères inter-dépendants, ou, à plus grande échelle, de mots inter-dépendants. En choisissant une granularité au niveau mot, on peut dire qu'ils sont contextualisés par les mots précédant ou succédant ceux en cours de traitement. Par exemple, on peut imaginer difficile pour un humain la compréhension d'une phrase dont on aurait coupé une partie du début et de la fin. C'est pourquoi il est nécessaire aux réseaux de neurones de pouvoir traiter une donnée séquentielle en ayant connaissance de son entièreté, afin de performer au mieux sur la tâche visée.

Les séquences d'entrée peuvent aussi être de longueur variable, ajoutant une complexité à prendre en charge dans l'architecture neuronale, ce que ne peut pas faire un perceptron multicouche dont le nombre d'entrées est fixe.

1.3.1 Couches récurrentes

Afin de pallier ces problématiques d'inter-dépendance et de longueur variables de données d'apprentissage, plusieurs modèles récurrents ont vu le jour. Pour cela, ils intègrent l'utilisation de boucles récurrentes mais aussi l'ajout de variables gardant en mémoire les informations pertinentes contenues dans une séquence d'entrée. Il est à noter que ces ajouts contribuent à augmenter considérablement le nombre de paramètres de leurs couches neuronales.

Ces différents modèles récurrents, présentés dans cette section, sont très utilisés dans le domaine du langage naturel, qu'il soit parlé ou écrit. Cette thèse étant focalisée sur le traitement de la parole, nous présenterons donc les architectures récurrentes à venir suivant le point de vue du traitement de la parole.

RNN

Les réseaux de neurones récurrents (*Recurrent Neural Network, RNN*) furent introduits par Jordan [1990] et Rumelhart [1986] et deviendront la base stable de nombreux autres réseaux récurrents plus complexes.

Lorsque nous parlons de boucle récurrente sur une couche neuronale, nous pouvons nous la représenter comme décrite par la Figure 1.7. L'ensemble d'une couche neuronale d'un *RNN* est communément rassemblée sous la schématisation d'une «cellule».

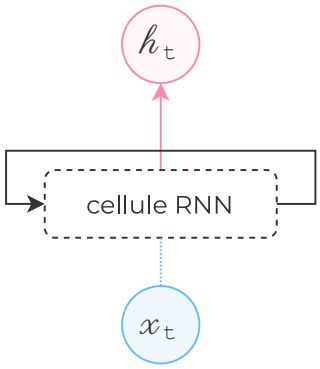

FIGURE 1.7 – Schématisation d'une boucle sur une cellule de *RNN*.

Il faut ainsi voir cette récurrence comme un stockage en mémoire des informations capturées précédemment par la cellule, utilisées telle une entrée additionnelle.

Pour réaliser ce stockage d'information, l'entrée traitée à un instant t peut considérer les précédentes, de t_0 à $t-1$. À chaque instant t d'une séquence d'entrée segmentée, par exemple en trames de parole, on notera x_t la trame d'entrée en cours de traitement. La Figure 1.8 illustre son traitement dans une cellule récurrente.

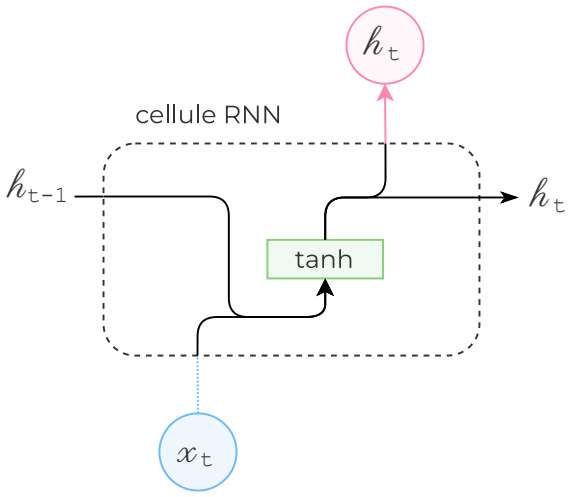

FIGURE 1.8 – Schématisation du fonctionnement d'une cellule de *RNN*.

La cellule récupère une représentation cachée h_{t-1} issue du traitement de la trame précédente à $t-1$ et en génère une nouvelle h_t à fournir lors du traitement de la prochaine trame à $t+1$. Suivant la tâche, on choisira d'utiliser toutes les représentations cachées précédentes ou uniquement la dernière à $t-1$. C'est cette même représentation cachée h_t à l'instant t qui sera la sortie de notre cellule, fournie aux couches supérieures du réseau.

Pour calculer ces représentations cachées, on réalisera une concaténation de notre entrée x_t à h_{t-1}. On appliquera la matrice de poids de la couche concernée à cette concaténation puis on passera sa sortie dans une fonction de tangente hyperbolique. Les paramètres appris par le modèle sont donc appliqués de la même façon que pour un modèle classique.

Dans le cas d'un *RNN* bi-directionnel dont nous parlerons plus loin, seront aussi considérées les trames de $t+1$ à t_n.

LSTM

Bien que les modèles récurrents aient été proposés pour tenter de préserver les informations pertinentes de la séquence traitée, il a été démontré par Bengio et al. [1994] qu'en pratique les *RNN* classiques souffrent d'une importante perte d'information lorsque celle-ci est temporellement trop éloignée, bien que pertinente.

C'est dans l'optique de trouver une solution à ce problème que les modèles récurrents à mémoire de court et long terme (*Long Short Term Memory, LSTM*) ont été développés par Hochreiter et Schmidhuber [1997].

Comme pour un *RNN* classique, des représentations cachées de l'instant $t-1$ seront utilisées à l'instant t et de nouvelles représentations seront créées et passées à leur tour à l'instant $t+1$.

Afin de mieux contrôler ce flux d'information dans une cellule récurrente, le *LSTM* met en place trois «portes» présentées par la Figure 1.9 : la porte d'oubli, la porte d'entrée et la porte de sortie.

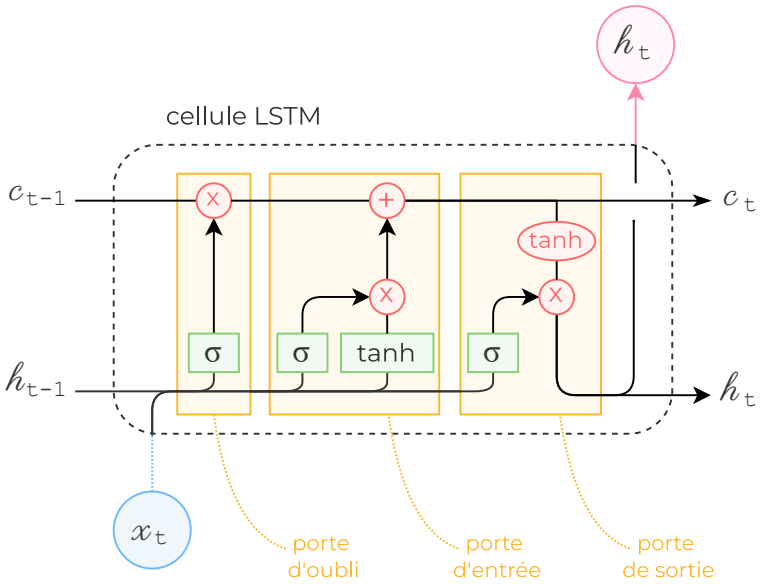

FIGURE 1.9 – Schématisation du fonctionnement d'une cellule récurrente de type *LSTM*.

Tout d'abord, il est important de voir chaque bloc avec indication d'une fonction d'activation – σ ou *tanh* – comme une couche neuronale de type perceptron à part entière. De plus, nous avons maintenant deux historiques à faire passer dans la récurrence de la cellule. Le premier, h, est le même que pour un *RNN* classique. La différence se situe dans c, appelé état de la cellule, qui permettra de garder un historique n'étant jamais influencé par la porte de sortie.

La porte d'oubli permet de réguler la sauvegarde des informations transmises par h_{t-1} mais aussi celles contenues dans l'entrée x_t. En générant des filtres compris entre 0 et 1 qui seront appliqués à c_{t-1} avant d'être réutilisé dans les portes suivantes, elle pourra réguler la présence des informations pertinences à la tâche, qu'elles proviennent de l'historique à l'instant $t-1$ ou de notre entrée.

La porte d'entrée permet d'affecter une pondération aux informations à modifier. Comme la porte d'oubli, elle nécessite donc de générer des valeurs dans l'intervalle $[0;1]$ qui géreront des taux de modification. La couche neuronale qui générera des valeurs issues d'une tangente hyperbolique nous donnera alors l'historique à ajouter à c_t et h_t.

La porte de sortie duplique le vecteur c_t résultant des précédentes portes et le modifie à son tour afin de générer h_t. Elle filtrera une fois de plus les données à sortir de la couche neuronale avec l'aide d'une Sigmoïde.

Il existe d'autres types de couches récurrentes, variantes du *LSTM*, tels que les unités récurrentes fermées (*Gated Recurrent Unit, GRU*) proposées par Cho et al. [2014], qui réduisent et ré-organisent les portes et assemblent l'état de la cellule c et l'historique h. Bien que plus légers que les *LSTM* [MOUMEN et PARCOLLET 2023], ils sont pourtant moins utilisés pour les tâches de traitement de la parole lorsque le but recherché se limite à la performance finale du système.

Récemment, le *Sli-GRU* (*Stabilised Light-GRU*) de Moumen et Parcollet [2023] a proposé une solution pour améliorer le *GRU* sur ces tâches, en partant du *Li-GRU* (*Light-GRU*) de Ravanelli et al. [2018], une version plus légère du *GRU* d'origine.

Enfin, nous parlions plus tôt de bi-directionnalité appliquée aux *RNN*. Pour un *LSTM*, on parle alors de *bi-LSTM*, tout comme nous parlerions de *bi-GRU*. Cette notion permet de récupérer les informations pertinentes situées n'importe où dans la séquence avec un second passage récurrent dans la cellule, réalisé grâce à la duplication de ses couches. Chaque couche bi-directionnelle sera donc composée de deux couches. La première traitera la séquence fournie de manière chronologique, comme dans un *LSTM* classique, tandis que la seconde la traitera de manière antéchronologique. Les sorties de ces deux couches seront par la suite concaténées, générant une sortie deux fois plus grande que d'ordinaire. En conclusion, le modèle bi-directionnel permettra de capturer les informations pertinentes passées et futures. La Figure 1.10 finit d'illustrer ce principe.

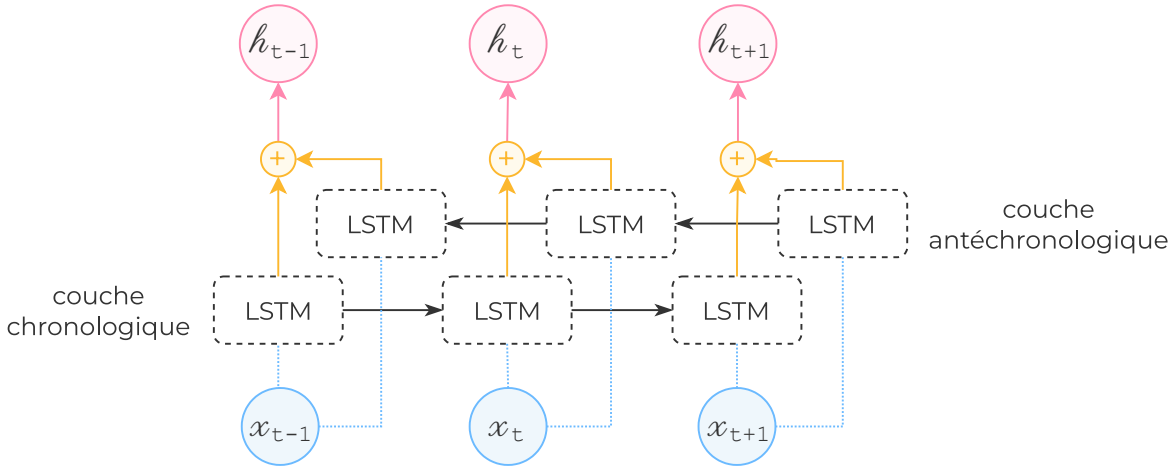

FIGURE 1.10 – Schématisation du fonctionnement d'un *RNN* bi-directionnel.

1.3.2 Couches convolutives

Il est nécessaire d'introduire les couches convolutives afin de pouvoir aborder les modèles auto-supervisés largement utilisés en traitement de la parole. Les réseaux de neurones convolutifs (*Convolutional Neural Networks, CNN*) introduits par Le Cun et al. [1989] sont très répendus dans le domaine de la vision, pouvant traiter toute sorte de matrices ordonnées. De ce fait, ils sont aussi

efficaces pour traiter des spectrogrammes, représentation imagées du signal de parole présentées au Chapitre 3. De nombreux travaux ont montré leur pertinence dans ce domaine, comme ceux de Peddinti et al. [2015] et Amodei et al. [2016].

On peut voir les images, qu'elles soient spectrogrammes ou autre, comme des matrices à deux dimensions principales : hauteur et largeur. Afin de traiter ces matrices de manière convolutive, on définit une fenêtre glissante qui permettra au *CNN* d'en apprendre un sous-ensemble. L'apprentissage consiste principalement à apprendre des filtres convolutifs, équivalents à une matrice de pondération appliquée par fenêtres locales sur les données d'entrée, dont la sortie sera la somme des éléments pondérés de chaque fenêtre. Chaque entrée est multipliée à un unique paramètre avant qu'elles ne soient toutes sommées entre elles. La Figure 1.11 schématise un *CNN*.

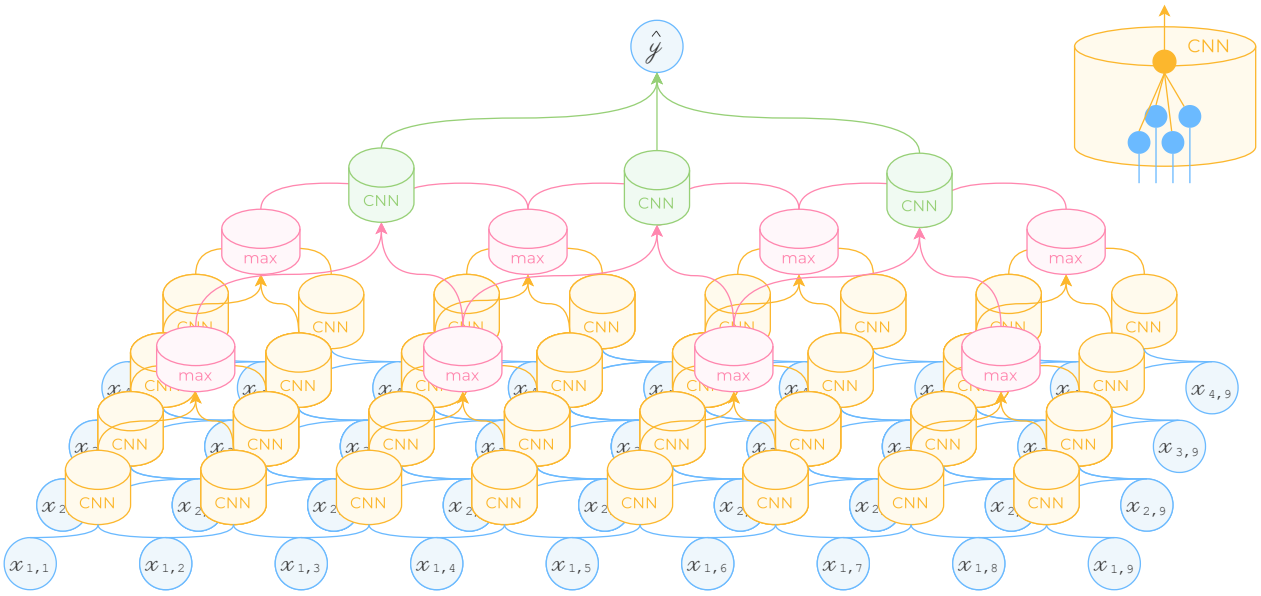

FIGURE 1.11 – Schématisation d'un exemple de *CNN* et de l'intérieur d'un de ses blocs.

On peut représenter une couche *CNN* comme la duplication d'un unique bloc neuronal équivalent à une couche linéaire classique structurée de manière matricielle. On parle alors de champs récepteur pour indiquer la sortie des blocs neuronaux passée à la couche supérieure du réseau.

Le bloc neuronal prendra en entrée non pas un mais plusieurs éléments de notre fenêtre glissante, suivant la taille du noyau souhaitée. Cette fenêtre se déplacera suivant un pas défini. Sur la Figure 1.11, nous n'avons pu représenter qu'une gestion de 4 éléments par bloc par soucis de lisibilité, mais les blocs convolutifs traitent généralement de bien plus grand nombres d'éléments.

Dans un *CNN*, plusieurs couches convolutives peuvent être accumulées. L'utilisation de fonctions de sous-échantillonage (*pooling*) est nécessaire afin de réduire la dimensionnalité de la sortie en limitant la perte d'information.

1.3.3 Transformers

Les architectures encodeur-décodeur et les mécanismes d'attention ont rapidement conquis le domaine du Traitement Automatique du Langage Naturel. Tout d'abord utilisés pour la Traduction Automatique, ils ont vu leur popularité s'élargir aux autres domaines tels que celui de la Compréhension Automatique de la Parole. Les architectures Transformer, utilisant ces deux mécanismes, furent introduits par Vaswani et al. [2017], initiant une nouvelle aire dans le domaine de l'Apprentissage Profond. Avant l'apparition des Transformers, d'autres architectures encodeur-décodeur utilisant des *RNN* avaient été proposées par Cho et al. [2014].

Architecture encodeur-décodeur

Comme nous avons pu le voir, certaines tâches traitent des entrées séquentielles de longueur variable. Ces tâches nécessitent parfois de générer une sortie elle aussi séquentielle et de longueur variable, différente de celle de la séquence d'entrée. C'est par exemple le cas dans le domaine de la Traduction Automatique, mais aussi dans celui de la Compréhension Automatique de la Parole. Ces tâches sont référencées comme réalisant un traitement dit «séquence vers séquence».

Les premiers modèles encodeur-décodeur ont été proposés par Cho et al. [2014] dans un contexte de traduction. L'architecture est séparée en deux modules distincts : l'encodeur et le décodeur, tous deux jusqu'alors consistant en une série de couches récurrentes.

L'encodeur traite les données d'entrée grâce à un *RNN* pour générer une sortie intermédiaire. Cette sortie prend la forme d'une représentation vectorielle de taille fixe. C'est dans celle-ci que sera stockée toute l'information essentielle à la réalisation de la tâche. Le décodeur traitera ensuite cette représentation vectorielle avec un second *RNN* afin de générer la sortie séquentielle attendue pour la tâche. La Figure 1.12 schématise un modèle de type encodeur-décodeur avec une entrée séquentielle segmentée en 4 éléments, et une sortie séquentielle segmentée en 3 éléments. Pour plus de clarté, nous utiliserons un exemple de Traduction Automatique avec l'entrée «Je bois du café» et la sortie attendue «I drink coffee».

L'encodeur est représenté dans la phase d'encodage à un instant t, allant ici de 1 à $n = 4$, par Enc_t. Le décodeur est représenté dans la phase de décodage à un instant t', allant ici de 1 à $m = 3$, par $Dec_{t'}$. Les entrées x_t et sorties h_t^{Enc} pour l'encodeur et $h_{t'}^{Dec}$ pour le décodeur sont celles d'un *RNN* comme présenté dans la Section 1.3.1. C'est la sortie de l'encodeur à h_n^{Enc}, avec $t = n$, qui sera notre représentation intermédiaire fournie comme entrée au *RNN* de la phase de décodage à tout instant t'. En d'autres termes, le vecteur généré par la couche cachée de l'encodeur pour le dernier élément de la séquence d'entrée représente l'encodage de l'ensemble de cette séquence.

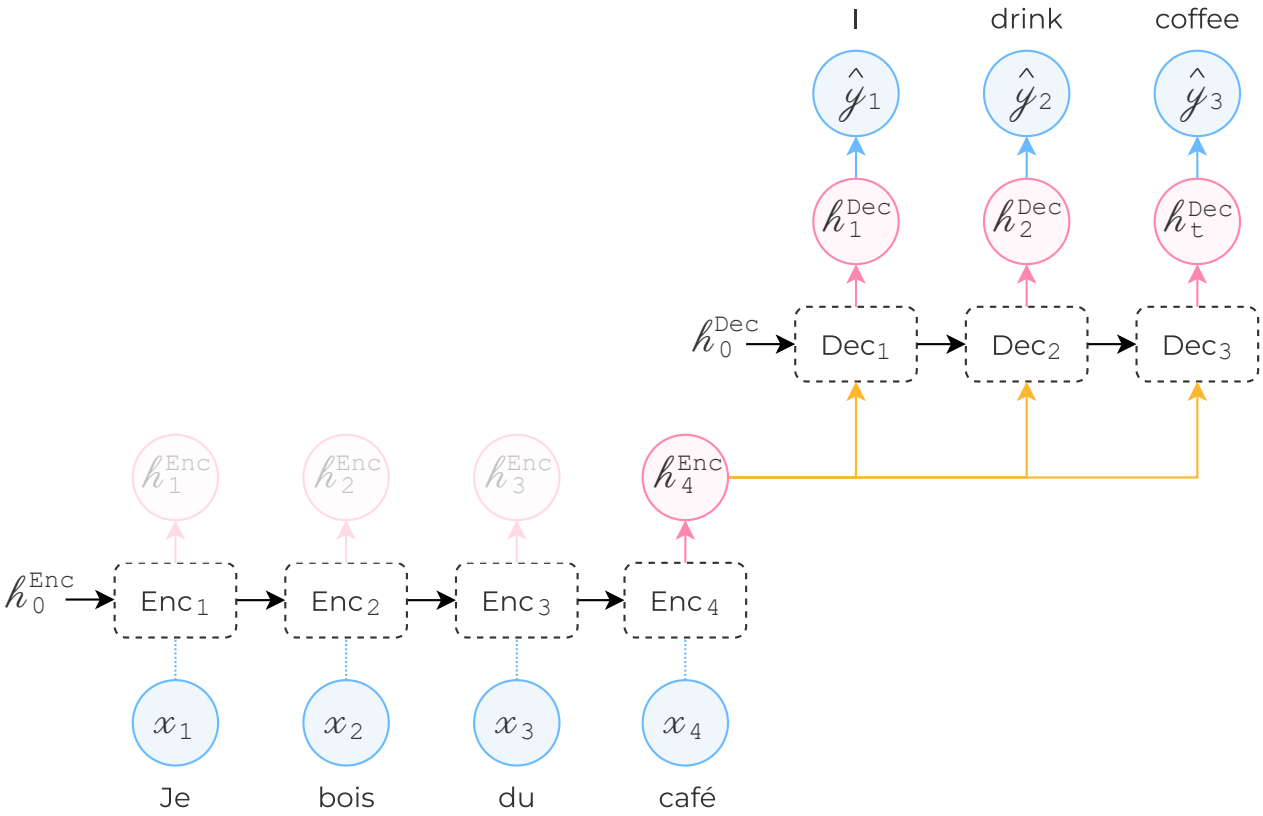

FIGURE 1.12 – Schématisation d'une architecture encodeur-décodeur pour un exemple de Traduction Automatique.

Un modèle de type encodeur-décodeur classique a pour inconvénient principal de fournir au module de décodage une représentation figée de la séquence d'entrée. Cette représentation intermédiaire entraîne une perte de précision de l'information transmise au décodeur.

Une solution à ce problème consiste en l'ajout de mécanismes d'attention entre l'encodeur et le décodeur. Cet ajout permet au modèle de se focaliser sur l'information pertinente contenue dans les données d'entrée afin de mieux la modéliser.

Mécanismes d'attention

Les mécanismes d'attention furent introduits par Bahdanau et al. [2015] bien qu'une variante fut proposée par Luong et al. [2015]. L'attention appliquée à une architecture encodeur-décodeur peut se schématiser comme sur la Figure 1.13.

L'approche modifie le transfert d'information entre encodeur et décodeur, rendant la représentation intermédiaire en sortie de l'encodeur plus dynamique.

Avec un mécanisme d'attention, toutes les sorties h_t^{Enc} quel que soit l'instant t seront pondé-

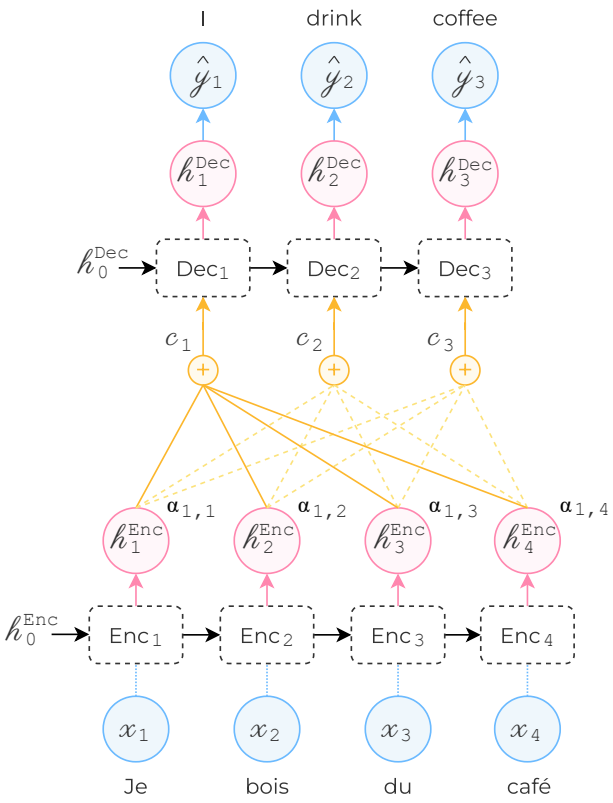

FIGURE 1.13 – Schématisation d'un encodeur-décodeur avec mécanisme d'attention.

rées puis sommées en un vecteur de contexte $c_{t'}$, fourni au décodeur à chaque instant t'. Afin de réaliser cette pondération pour mettre en avant l'information pertinente contenue dans $c_{t'}$, on utilise des coefficients d'attention $\alpha_{tt'}$, propres à chaque sortie de l'encodeur h_t^{Enc}. Ainsi, pour une entrée du décodeur à l'instant t' et un nombre n d'entrées de l'encodeur, on notera :

$$c_{t'} = \sum_{t=1}^{n} \alpha_{tt'} \, h_t^{Enc} \tag{1.17}$$

Chaque coefficient d'attention $\alpha_{tt'}$ sera obtenu par un réseau de neurones prenant en entrée h_t^{Enc} et $h_{t'-1}^{Dec}$ et calculant son énergie. Cette énergie et l'ensemble des énergies obtenues pour toute sortie h_t^{Enc} passeront ensuite par une fonction Softmax afin d'obtenir $\alpha_{tt'}$ compris dans l'intervalle $[0, 1]$.

L'attention multi-têtes (*multi-head attention*) introduite par Vaswani et al. [2017] est un mécanisme d'attention optimisant parallèlement plusieurs têtes d'attention au sein d'un même bloc d'encodage ou de décodage. Optimiser une de ces têtes d'attention revient à optimiser trois projections linéaires qui lui sont propres : *Query*, *Key* et *Value*. Ces projections fournissent à chaque instant t des représentations vectorielles nommées q_t, k_t et v_t. Nous noterons ces projections Q,

K et V lorsque nous considérerons tout instant t. De la même manière, on notera h_t^{Bloc} la sortie de la couche cachée d'un bloc d'encodage ou de décodage à un instant t et H^{Bloc} l'ensemble des sorties d'un bloc d'encodage ou de décodage pour tout instant t.

La Figure 1.14 schématise le calcul d'un vecteur de contexte c_t obtenu avec trois têtes d'attention. Hormis pour q_t, on considérera les projections K et V pour tout instant t.

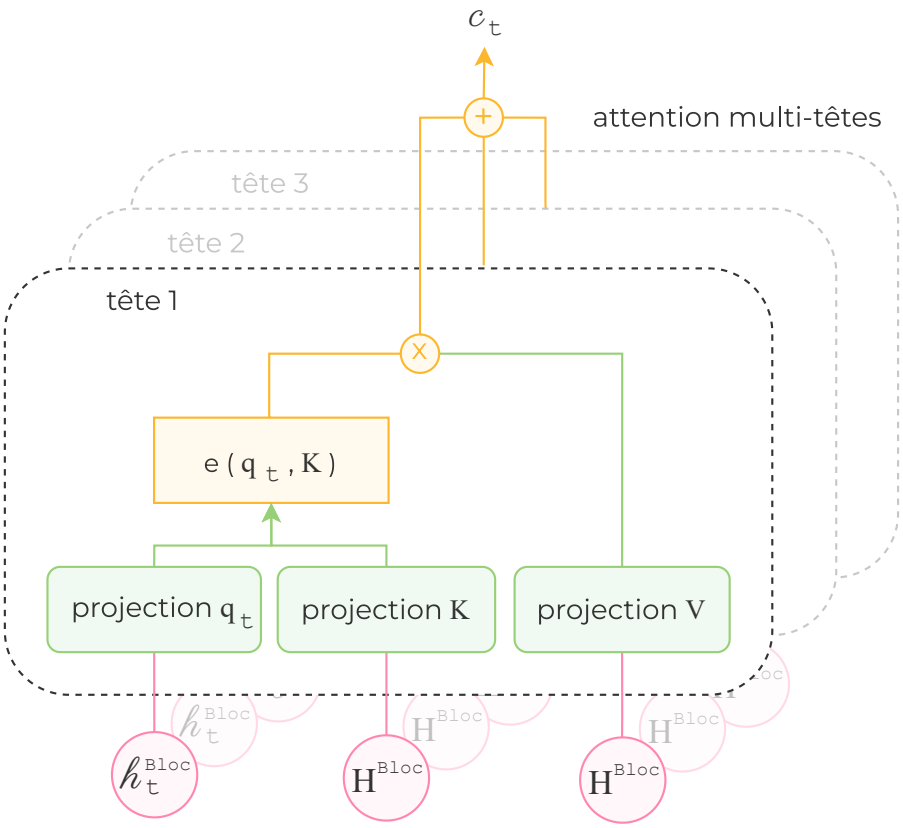

FIGURE 1.14 – Schématisation d'un mécanisme d'attention multi-têtes de Vaswani et al. [2017].

L'énergie e des vecteurs q_t et K pour tout instant t est obtenue par la formule :

$$e(q_t, K) = Softmax(q_t \times K) \tag{1.18}$$

Le vecteur résultant pondérera les projections V à tout instant t comme suit :

$$c_t = \sum V \times e(q_t, K) \tag{1.19}$$

L'auto-attention (*self-attention*) est un type de mécanisme d'attention popularisé par Vaswani et al. [2017] dans l'article introductif des modèles Transformer. Elle est réalisée à partir des sorties H pour tout instant t produites par la couche cachée du précédent bloc d'encodage ou de décodage.

Pour l'attention croisée multi-têtes, utilisée uniquement dans le décodeur, on projettera Q à partir des sorties cachées provenant du bloc de décodage précédent, tandis que K et V seront projetés grâce à la sortie du dernier bloc de l'encodeur.

Pour le premier bloc d'une pile, ce sont les entrées de cette pile qui sont utilisées à la place de celles sorties d'un bloc précédent.

Transformers

Les Transformers sont des modèles de type encodeur-décodeur qui n'utilisent pas de réseaux récurrents. Ils sont composés de ce qu'on appelle des «blocs» d'encodage et de décodage, appris en parallèle. La Figure 1.15 présente une simplification d'un Transformer à n blocs d'encodage et m blocs de décodage.

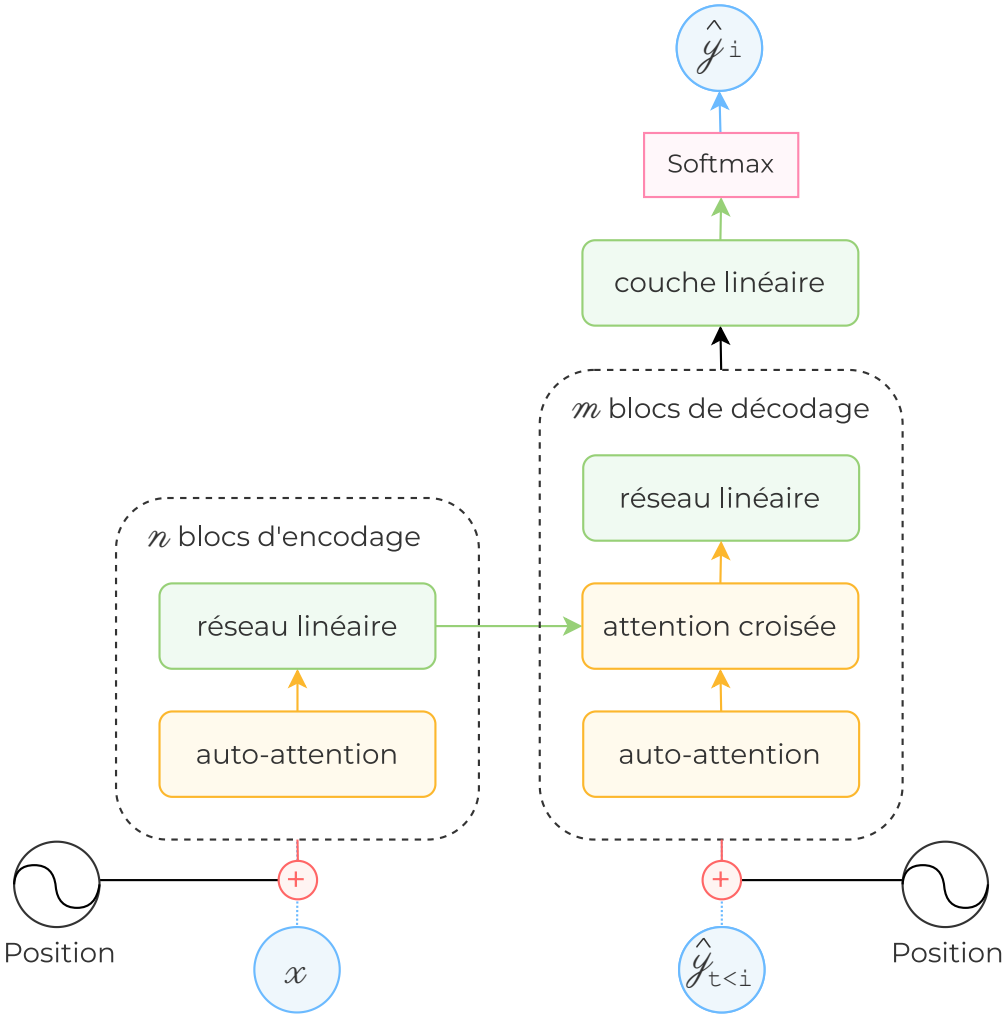

FIGURE 1.15 – Schématisation simplifiée d'un modèle Transformer.

Chaque bloc d'encodage contient un bloc d'auto-attention multi-têtes suivie d'un réseau linéaire. Les blocs de décodage contiennent eux aussi un bloc d'auto-attention multi-têtes qui traite les sorties précédemment prédites par le modèle. Suit un bloc d'attention croisée multi-têtes faisant la liaison entre l'encodeur et les blocs de décodage, ainsi qu'un autre réseau linéaire. On notera aussi l'ajout d'une représentation positionnelle (*positional embedding*) des entrées de l'encodeur et du décodeur, permettant d'indiquer leur position temporelle dans le signal d'entrée.

Bien que les réseaux Transformer soient très performants et largement utilisés de nos jours, il faut prendre en considération le coût important de leur apprentissage dû au nombre conséquent de paramètres à optimiser et de données à apprendre, comme étudié par Devlin et al. [2019] et Brown et al. [2020]. Cela n'a pas freiné l'évolution du Transformer, performant toujours plus.

1.4 Apprentissage auto-supervisé

Réaliser l'apprentissage de modèles comme le Transformer, constitués d'un grand nombre de paramètres, nécessite une très grande quantité de données étiquetées. Or, dans de nombreux domaines comme celui du Traitement Automatique du Langage Naturel, il est difficile de réunir une quantité suffisante de données pour réaliser ces apprentissages.

Cependant, de grands ensembles de données non-étiquetées existent, comme c'est le cas pour Libri-Light [KAHN et al. 2020] avec 60 000 heures d'enregistrements audio. Afin de tirer profit de ces corpus, l'apprentissage auto-supervisé (*self-supervised learning, SSL*) utilise ces données pour optimiser des réseaux de neurones denses et profonds. Cet apprentissage sert principalement de pré-apprentissage, les modèles appris étant ensuite *fine-tunés* sur une tâche plus précise.

Le terme d'auto-supervision vient du fait que c'est le modèle lui-même qui va créer ses propres «étiquettes» durant l'apprentissage, qui reste par nature supervisé. Des tâches ont dont été pensées pour pouvoir créer des étiquettes pertinentes. Généralement, il s'agira de masquer un élément de la séquence d'entrée du modèle. Le modèle en question tentera de retrouver l'élément manquant. On nommera cette tâche, illustrée par un exemple en Figure 1.16, la Modélisation de Langage Masqué (*Masked Language Modeling, MLM*).

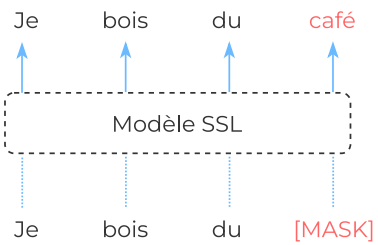

FIGURE 1.16 – Exemple d'une tâche de masquage en apprentissage auto-supervisé.

Les modèles SSL pour l'encodage de la parole, étudiés durant cette thèse, seront abordés en profondeur au Chapitre 3. Il sera alors question d'en préciser leur architecture et fonctionnement.

1.5 Algorithmes de recherche

En Traitement Automatique du Langage Naturel, les algorithmes de recherche permettent de produire une séquence de sortie à partir des distributions de probabilité prédites par le modèle. Dans cette thèse, nous nous sommes focalisés sur la prédiction de séquences de caractères. Nous illustrerons donc le fonctionnement des algorithmes de recherche suivants par une prédiction de séquence de caractères.

Algorithme glouton

L'algorithme glouton (*greedy decoding*) est l'algorithme de recherche le plus basique et rapide. Il consiste à choisir l'hypothèse la plus probable à chaque instant t, indépendamment du reste de la séquence prédite. La Figure 1.17 donne un exemple de recherche, avec comme sortie attendue le mot «salut». Nous avons représenté le chemin réalisé par l'algorithme de manière verticale, chaque flèche représentant le passage à l'instant $t + 1$.

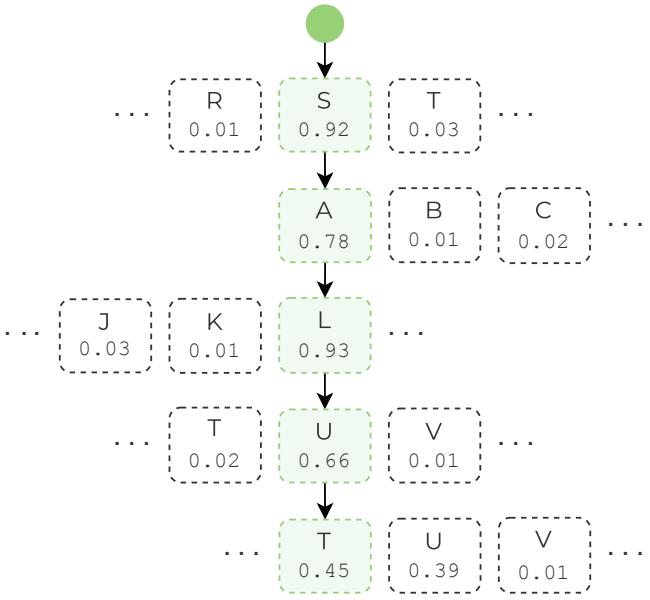

FIGURE 1.17 – Exemple de recherche par algorithme glouton pour une séquence de caractères attendue «salut».

À chaque instant t, le caractère le plus probable sera choisi pour alimenter la branche de recherche. Pour des raisons de lisibilité, nous n'illustrons qu'un choix parmi trois caractères de l'ensemble du vocabulaire possible.

Recherche par faisceau

L'algorithme de recherche par faisceau (*beam search*) parcourt plusieurs branches de recherche, explorant ainsi bien plus l'arbre de probabilités fourni par le modèle. Contrairement à l'algorithme glouton, il est d'usage d'utiliser un modèle externe en combinaison avec l'algorithme de recherche par faisceau. Pour les traitements de la langue, on utilisera un modèle de langue, comme présentés au Chapitre 2. Ce modèle externe attribuera des scores supplémentaires aux sorties, en fonction de la séquence prédite. Il s'agira ensuite de cumuler les probabilités de chaque branche de recherche afin de déterminer quelle séquence est la plus probable. La Figure 1.18 illustre ce principe, toujours avec un nombre limité de vocabulaire illustré à chaque instant t pour des raisons de lisibilité.

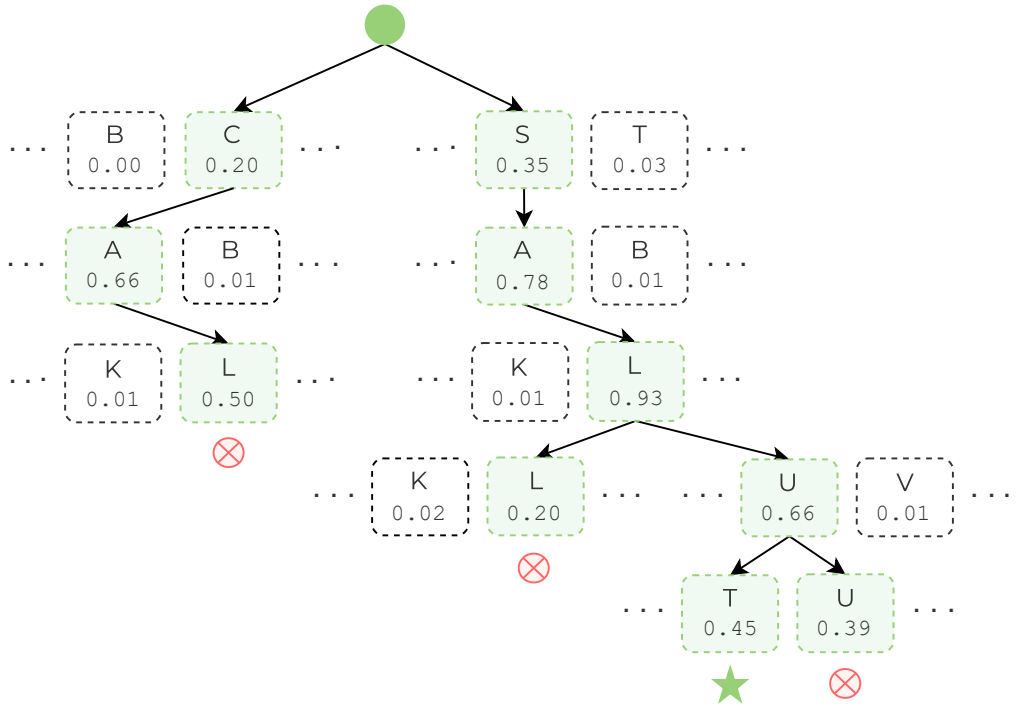

FIGURE 1.18 – Exemple de recherche par faisceau de largeur 2 pour une séquence de caractères attendue «salut».

Il est à noter que malgré les meilleures performances de cette approche, celle-ci est plus gourmande en calculs et stockage d'information, surtout pour des vocabulaires conséquents.

L'intérêt de l'algorithme de recherche par faisceau est de diminuer ces coûts computationnels importants par l'ajout d'une limitation de n branches pouvant être étudiées à chaque instant t, nommée largeur de faisceau. Une largeur de faisceau élevée augmente donc ces coûts, tandis que définir $n = 1$ équivaut à réaliser un algorithme glouton. La Figure 1.18 représente une recherche par faisceau avec $n = 2$.

Il existe des variantes à ces deux méthodes de recherche, comme top-k de Fan et al. [2018] et top-p de Holtzman et al. [2020].

1.6 Conclusion

Ce chapitre donne une vue d'ensemble de l'Apprentissage Profond orienté pour le Traitement Automatique du Langage Naturel et notamment celui de la Parole, présenté au Chapitre 2.

À travers la présentation de réseaux de neurones simples et plus complexes tels que les réseaux récurrents, réseaux convolutifs et Transformers, il permet de constituer une base théorique pour l'introduction aux encodeurs de paroles présentés au Chapitre 3. Ce chapitre décrit aussi les différentes méthodes d'optimisation utilisées durant les expérimentations de cette thèse, comme c'est le cas pour les fonctions de coût de Classification Temporelle Connexionniste et similarité cosinus, les optimiseurs AdaDelta et Adam, et l'algorithme de recherche glouton. Par ailleurs, il met en avant les concepts de pré-apprentissage et de *fine-tuning*, essentiels à nos expérimentations multilingues et cross-lingues discutées au Chapitre 6.

Chapitre 2

Compréhension Automatique de la Parole

Sommaire

La Compréhension Automatique de la Parole (*Spoken Language Understanding, SLU*), est une ambition visée par diverses sous-tâches dont nous présentons les principales en Section 2.1. Elle fut définie par De Mori et al. [2007] comme l'interprétation des informations transportées par un signal de parole. Cette thèse aborde l'extraction du sens de ce signal de parole en se reposant sur l'extraction de la sémantique des mots prononcés dans un langage naturel. On peut parler ici de projection depuis un domaine acoustique vers un domaine sémantique.

Béchet [2007] définit le sens comme porteur de différentes perspectives : philosophique, linguistique, cognitive, mathématique ou computationelle. La sémantique est définie par Woods [1975] comme une organisation des relations entre les symboles et signes d'un langage et leur signification. En pratique, il s'agira d'extraire automatiquement ces signes et symboles afin de les conceptualiser en des représentations sémantiques plus ou moins structurées. Différents formalismes sont présentés en Section 2.1.4.

Les domaines applicatifs pour la Compréhension de la Parole sont nombreux et la recherche dans ce domaine ne cesse de croître depuis ces dernières décennies. D'un point de vue industriel, les représentations sémantiques peuvent par exemple être utilisées pour la communication entre un humain et une machine (routage d'appels, assistants vocaux, réservation en ligne, domotique, etc.).

Après avoir abordé quelques unes des principales tâches visant la Compréhension Automatique de la Parole et présenté les méthodes de représentation utilisées pour l'extraction sémantique d'un signal de parole, ce chapitre abordera deux catégories de systèmes utilisés dans ce domaine : les systèmes dits en cascade, utilisant plusieurs modules appris séparément, et les systèmes de bout-en-bout.

Les tâches de Compréhension Automatique de la Parole tirent leur difficulté de l'extraction d'informations vocales bien plus denses et complexes que celles contenues dans un simple texte. Il s'agira d'adapter le système au style du locuteur (discours spontané ou lecture), à sa prononciation (accent, intonation, vitesse d'élocution), à l'environnement d'enregistrement (bruits, réverbérations) et à la tâche en elle-même (taille du vocabulaire, langue, domaine, quantité de données limitée). Les systèmes en cascade contournent la majorité de ces difficultés en réalisant une première transcription automatique du signal de parole avant d'y appliquer des systèmes de compréhension du langage naturel écrit. Les systèmes de bout-en-bout réalisent la prise en charge directe du signal de parole et de tous ses paramètres acoustiques au sein d'un unique système.

Nous terminerons ce chapitre en faisant la liaison entre Compréhension Automatique de la Parole et évolution du domaine de la Traduction Automatique, menant à un état-de-l'art des systèmes cross-lingues dans notre domaine.

2.1 Tâches pour la compréhension de la parole

Les premiers pas vers la compréhension de la parole remontent aux *MUC* (*Message Understanding Conferences*), organisées dès 1987 par la *DARPA* (*Defense Advanced Research Projects Agency*) afin de réaliser l'analyse automatique de messages textuels militaires. Ces campagnes d'évaluation ont permis de populariser la recherche pour l'extraction sémantique textuelle, proposant lors de la sixième édition [SUNDHEIM 1995] une nouvelle tâche d'extraction d'entités nommées et d'extraction de co-références.

Pour plus de détails, Grishman et Sundheim [1996] donnent un aperçu de l'évolution des *MUC* et des tâches qui y furent traitées.

De nos jours, le domaine de la Compréhension Automatique de la Parole peut être abordé de diverses manières. En effet, la difficulté principale du domaine réside en l'évaluation de la «compréhension» acquise par le système, c'est pourquoi il existe de nombreuses tâches et métriques d'évaluation de niveaux de précision variés, dédiées à un domaine applicatif spécifique.

Certaines traiteront l'ensemble d'un document audio pour le catégoriser en thématique ou domaine. D'autres se focaliseront sur la détection d'intention dans un segment de parole – équivalant à une phrase dans le domaine textuel – afin de connaître la volonté de l'utilisateur sur un temps assez court. Enfin, certaines tâches auront pour but d'extraire la sémantique à un niveau plus fin, visant la compréhension de chaque mot prononcé. Il s'agira de tâches comme celles d'extraction d'entités nommés ou de concepts sémantiques. Parfois, d'autres aspects de la compréhension du langage seront recherchés, comme par exemple la génération d'un résumé automatique.

Toutes ces tâches, que nous allons aborder dans cette section, ont en commun la particularité d'extraire le sens porté par le signal de parole.

2.1.1 Reconnaissance d'entités nommées

L'origine de la tâche de reconnaissance d'entités nommées (*Named Entity Recognition, NER*) remonte à la *MUC*-6 [SUNDHEIM 1995]. Décrite alors comme une tâche d'insertion de balises sémantiques normées dans un texte, elle permet d'indiquer quels mots ou groupes de mots représentent des *personnes*, *organismes*, *localisations*, *dates*, *temps*, *monnaies* ou *pourcentages*, sans alors prendre en compte leurs co-références. Ces co-références étaient visées par une autre tâche lors de la même campagne d'évaluation. On peut donner l'exemple de «la ville des amoureux», qui est une co-référence à «Paris» dans la phrase «Paris est la ville des amoureux.» et pourrait donc être étiquetée avec la même balise.

Lorsque cette tâche est utilisée pour la compréhension de la parole, elle peut être appliquée directement sur le signal de parole ou sur sa transcription [GALIBERT et al. 2011]. Le but restera l'extraction de toute information acoustique ou textuelle pertinente afin de catégoriser par des

entités nommées les chaînes de caractères qui composent la transcription.

Cette catégorisation réunit les problématiques d'une segmentation et d'une classification. La segmentation consistera à découper le segment transcrit en trouvant le début et la fin de chaque chaîne de caractères à associer à une entité nommée. La classification se traduira par l'association de cette chaîne de caractères à une des entités nommées d'un ensemble pré-défini.

Cette tâche a été représentée par Nouvel et al. [2015] comme le remplissage d'un formulaire aux champs pré-définis (*slot-filling*). Chaque champs représente ainsi une entité nommée, et les chaînes de caractères qu'il contient sont appelées sa valeur. À l'état brut, cette valeur sera plus généralement nommée «mots-support» avant d'être normalisée en «valeur».

Parmi les ensembles de données visant la tâche de reconnaissance d'entités nommées, nous pouvons citer ESTER [2004] et ETAPE [2012] de Gravier et al. et QUAERO de Grouin et al. [2011].

Plusieurs types d'annotations, présentées en détail en Section 2.1.4, ont été proposées pour la tâche de reconnaissance d'entités nommées. Ce fut notamment le cas avec Galliano et al. [2009] qui proposèrent une structuration des entités nommées par imbrication. Grouin et al. [2011] proposèrent par la suite un autre format d'annotation en deux temps, consistant à ajouter une annotation en composants sur les chaînes de mots-support déjà catégorisées par des entités nommées classiques.

Concernant l'évaluation des systèmes, il est courant d'utiliser deux types de taux d'erreur. Le premier, qu'on traduira par taux d'erreur de champs (*Slot Error Rate, SER*), fut introduit par Makhoul et al. [2007] et est encore majoritairement employé par la communauté scientifique à ce jour. Le second, le taux d'erreur d'arbre d'entités (*Entity Tree Error Rate, ETER*), fur proposé par Ben Jannet et al. [2014] afin d'évaluer la performance des systèmes de reconnaissance d'entités nommées en tenant compte de la structure hiérarchique des entités dans le texte. Le concept d'arbre d'entité suppose que les entités nommées peuvent être organisées de manière hiérarchique. Par exemple, une entité peut être comprise dans une autre entité plus large, comme c'est le cas pour les deux représentations structurées mentionnées plus loin [GALLIANO et al. 2009 ; GROUIN et al. 2011].

2.1.2 Extraction de concepts sémantiques

On peut voir les entités nommées précédemment décrites comme des catégories répondant à une norme consentie par la communauté scientifique. Dans le domaine de la Compréhension Automatique de la Parole, les concepts sémantiques ne répondent pas à une norme générale mais sont tout de même pré-définis pour une tâche précise, adaptés autant à son domaine qu'à son objectif. Les tâches de reconnaissance d'entités nommées et d'extraction de concepts sémantiques

sont similaires, la différence principale résidant en la définition de leurs balises. On peut dire que la reconnaissance d'entités nommées forme donc un cas particulier d'extraction sémantique.

De manière plus générale, un concept sémantique se réfère à une unité abstraite qui représente une idée, allant au-delà des aspects purement syntaxiques ou formels du langage pour en capturer le sens [WOODS 1975]. Dans les domaines de la linguistique et des sciences cognitives, les concepts sémantiques sont des unités de sens qui forment la base de la compréhension du langage et de sa signification.

Bien que la tâche d'extraction de concepts sémantiques puisse être aussi vue comme un remplissage de formulaire, une autre distinction peut être faite avec la reconnaissance d'entités nommées au niveau de leurs champs d'application. Tandis que la reconnaissance d'entités nommées sera plutôt utilisée pour la compréhension de documents, l'extraction de concepts sémantiques sera principalement exploitée dans le cadre d'interactions humain-machine [JABAIAN 2012 ; MESNIL, DAUPHIN et al. 2015]. Le traitement de ces interactions et plus particulièrement celui de dialogues transactionnels [TÜR et DE MORI 2011] est d'un grand intérêt quand il s'agit de viser la compréhension de la parole. La forte liaison entre la tâche traitée et le vocabulaire sémantique utilisé permet un degré de précision important quant à la capture du sens de l'interaction.

Les domaines d'application pour l'extraction de concepts sémantiques sont très variés, pouvant aller de la réservation d'hôtels ou de spectacles, comme avec MEDIA [BONNEAU-MAYNARD, AYACHE et al. 2006] et PortMEDIA [LEFÈVRE, MOSTEFA et al. 2012] à celle de restaurants comme dans M2M [SHAH et al. 2018]. On notera aussi la variation du milieu d'enregistrement, par exemple avec une capture en direct d'une conversation à un guichet, comme c'est le cas pour TARIC-SLU [MDHAFFAR, BOUGARES et al. 2024], ou via une conversation téléphonique comme pour DECODA [BECHET et al. 2012]. L'enregistrement pourra se faire avec l'aide d'un humain jouant le rôle d'une machine (*Wizard-of-Oz, WoZ*), comme ce fut le cas au niveau textuel pour MultiWOZ [BUDZIANOWSKI et al. 2018] puis pour le traitement de la parole avec, entre autres, SpokenWOZ [SI et al. 2024].

Certains de ces ensembles de données, comme c'est le cas d'ATIS [HEMPHILL et al. 1990], dont la tâche visée est un service de renseignement pour des voyages par avion, voient leur popularité progressivement diminuer [TÜR, HAKKANI-TÜR et al. 2010 ; BÉCHET et RAYMOND 2018]. Le domaine de l'extraction de concepts sémantiques depuis la parole n'est que peu fourni en données, avec des ensembles souvent de petite taille, peu généralisables, pouvant amener à une stagnation de performances.

Les métriques d'évaluation les plus courantes sont les taux d'erreur de concept (*Concept Error Rate, CER*) et de paire concept-valeur (*Concept Value Error Rate, CVER*) que nous présenterons plus en détail au Chapitre 4.

2.1.3 Autres tâches

Cette thèse aborde principalement la tâche d'extraction de concepts sémantiques, étroitement liée à celle de reconnaissance d'entités nommées. Il existe néanmoins de nombreuses autres tâches visant la compréhension de la parole, comme c'est de la de la classification d'intention, dont un travail de recherche est présenté en Annexe, mais aussi la classification de domaine, parfois appelée classification par thématique, et la tâche de résumé automatique.

Nous présentons brièvement ces trois dernières dans cette section.

Classification d'intention

La détection d'intention consiste à classifier un segment de parole en une intention globale du discours de l'utilisateur. D'une précision plus faible que les tâches de reconnaissance d'entités nommées et d'extraction de concepts sémantiques, elle est ainsi moins complexe. Contrairement à celles-ci où la génération d'une transcription est de mise pour l'évaluation des mots-support ou valeurs normalisées, la classification d'intention ne nécessite pas de transcription, seulement une classification évaluée généralement par une F-mesure [VAN RIJSBERGEN 1974].

Une difficulté réside cependant dans la formulation du discours, variant d'un segment à l'autre pour une même intention. Il peut alors s'agir de détecter des mots clés énoncés par le locuteur, ou, de manière plus complexe, de condenser l'information contenue dans une chaîne de caractères pour en extraire le sens global [TÜR et DE MORI 2011].

Cette tâche peut elle aussi être réalisée sur des dialogues humain-machine [GORIN et al. 1996] et vue comme le remplissage d'un formulaire [B. LIU et LANE 2016]. Elle mènera ensuite à d'autres mises en application, comme par exemple indiquer à un appareil domotique ou un assistant vocal quelle action réaliser [SAXON et al. 2021 ; RASTOGI et al. 2020 ; DESOT et al. 2019]. Lorsque l'application permet la classification d'un appel entrant dans un centre d'appel et sa redirection vers le département approprié, on pourra parler de routage d'appel [RABINER 2005]. Cette tâche peut nécessiter la prise en compte d'un contexte externe au segment traité, la rendant plus complexe [P. XU et SARIKAYA 2013].

Classification de domaine

La classification de domaine, aussi appelée classification par thème, réside en la segmentation puis classification d'un document en sous-ensembles thématiques et est aussi évaluée par une F-mesure. Cette tâche peut avoir pour application la simplification de recherche documentaire ou bien le résumé d'un document avec l'aide d'un second traitement sur les sous-parties segmentées, facilitant les processus de résumé automatique présentés un peu plus loin.

La complexité de la tâche est située dans les domaines ou thèmes à classifier et dans l'organisation même du document. Il sera parfois simple de délimiter les sujets individuels dans le discours, comme c'est le cas pour le domaine de discours journalistiques [GUINAUDEAU 2011 ; BOUCHEKIF 2016]. En revanche, une discussion moins organisée ou un échange mono-thématique seront plus complexes à traiter, menant à un traitement dépendant intrinsèquement de la tâche [NIEKRASZ et MOORE 2009], par exemple par la définition de thématiques d'intention [PASSONNEAU et LITMAN 1997]. Des recherches sur ces segmentations thématiques ont été menées dans le cadre du projet PASTEL [MDHAFFAR, LAURENT et al. 2018]. Il est aussi possible de réaliser cette tâche de manière non-supervisée, par regroupement thématique [SEYMORE et ROSENFELD 1997], les thèmes n'étant pas nécessairement connus par avance.

Résumé automatique

Le résumé automatique est une tâche visant à condenser l'information pertinente d'un discours ou d'un texte, triant et facilitant l'accès à la documentation conséquente mise à la disposition des utilisateurs. L'objectif est d'en garder le sens tout en limitant le support nécessaire à sa représentation. Elle est évaluée avec la métrique ROUGE (*Recall-Oriented Understudy for Gisting Evaluation*) [C. LIN 2004].

Plusieurs méthodes existent, plus ou moins complexes. Une méthode possible peut être de simplement extraire et assembler les phrases ou groupes de phrases clefs du discours [LUHN 1958]. La difficulté résidera en un classement des phrases selon leur niveau d'importance. Il est aussi possible de réaliser une totale reformulation condensée du document [KLEINBAUER et al. 2007 ; MURRAY et al. 2010] ou de faire un résumé de plusieurs documents [MASKEY et HIRSCHBERG 2003].

2.1.4 Représentations sémantiques

Afin de réaliser une tâche d'extraction de sémantique depuis la parole ou le texte, via entités nommées ou concepts sémantiques propres à la tâche, il faut choisir un format de représentation pour la sortie attendue.

La compréhension de la parole est un but subjectif complexe pouvant être abordé de diverses manières. Il existe donc de nombreuses façons de représenter la sémantique dans une phrase, de manière plus ou moins structurée. Cette section en donne un aperçu, présentant des représentations dites «à plat» ou «structurées». Les deux formats les plus couramment employés pour l'extraction sémantique par concepts ou entités nommées sont des représentations à plat : le BIO et le balisage de sous-segments. Bien que le choix de la représentation sémantique puisse influencer l'apprentissage, il sera toujours possible de changer la représentation de la sortie prédite d'un format à un autre.

Globalement, le principe restera de prédire des concepts et leur valeur, normalisée ou non, reposant sur des mots-support issus de la transcription automatique du signal de parole.

Représentations sémantiques à plat

Le format BIO (*Beginning Inside Outside*) de Ramshaw et Marcus [1995] consiste à attribuer à chaque mot une étiquette sémantique précédée d'un indice de position relative à cette étiquette. Pour une transcription automatique issue d'un signal de parole, il faut voir un mot comme une chaîne de caractères parfois abstraite, séparée d'une autre par une ponctuation ou un espace.

Comme l'illustre la Table 2.1, l'indice de position pourra être un B (*Beginning*) lorsque le mot sera le premier de la chaîne de mots-support liée à l'étiquette sémantique, ou un I (*Inside*) s'il ne la commence pas. Lorsque le mot ne sera lié à aucune étiquette du schéma sémantique pré-défini, par exemple comme c'est le cas pour un mot de liaison ou un pronom, on lui attribuera uniquement l'indice O (*Outside*) sans étiquette sémantique.

Afin d'obtenir des valeurs normalisées pour l'évaluation du système appris, les mots-support d'une même étiquette peuvent être regroupés et traités.

mot	**étiquette**	**valeur**
Le	O	
compositeur	O	
Ludwig	B-PERSON	
van	I-PERSON	Beethoven
Beethoven	I-PERSON	
est	O	
né	O	
le	O	
quinze	B-DATE	
décembre	I-DATE	15/12/1770
1770	I-DATE	
à	O	
Bonn	B-LOCATION	Bonn

TABLE 2.1 – Exemple de représentation sémantique à plat au format BIO.

Une représentation sémantique par sous-segments (*chunks*) peut aussi être utilisée pour les tâches de reconnaissance d'entités nommées et d'extraction de concepts sémantiques, bien que plus utilisée pour cette dernière [GHANNAY, CAUBRIERE, ESTÈVE et al. 2018].

Le segment sera alors annoté avec des balises ouvrantes et fermantes pour une étiquette sémantique précise, comme illustré par l'exemple en Figure 2.1. Ces étiquettes, aussi nommées attributs ou concepts, encadreront une chaîne de caractères appelée mots-support. Lorsque normalisés, ces mots-support formeront la valeur de l'étiquette. Cette normalisation peut se faire de diverses manières, souvent après apprentissage du système et génération des transcriptions annotées de balises sémantiques. Il est courant d'utiliser des règles humaines pré-définies, bien que celles-ci aient leurs inconvénients, comme nous le démontrons au Chapitre 4. Les mots hors de ces balises sont équivalents aux mots étiquetés O pour le format BIO, ne portant pas de sens en rapport avec la représentation sémantique visée.

Un avantage de ce format vis-à-vis du BIO est qu'il facilite une transcription et annotation sémantique conjointe lors d'un unique apprentissage, sans forcément nécessiter de passer par une transcription intermédiaire. Son système d'annotation est aussi plus global, n'étiquetant pas chaque mot indépendemment, mais directement un groupe de mots sans distinction de la position des mots-support. L'inconvénient principal réside dans le mélange fait entre étiquettes et transcription qui ne permet pas au système de différencier aisément les balises du texte.

Transcription <concept *valeur* > mots-support > transcription.

J'aimerais <command-tache *réservation* > réserver > euh
<nombre-chambre-reservation *2* > deux >
<chambre-type *simple*> chambres individuelles > si possible
<temps-date-debut-reservation *19/10/2024* > du dix neuf >
<temps-date-fin-reservation *22/10/2024* > au vingt deux octobre >
<localisation-ville-hotel *Paris* > à Paris > s'il-vous-plaît.

FIGURE 2.1 – Exemple de représentation sémantique à plat par étiquetage de sous-segments pour l'ensemble de données MEDIA [BONNEAU-MAYNARD, ROSSET et al. 2005].

Représentations sémantiques structurées

Les concepts sémantiques et leurs relations peuvent être décrits comme des réseaux ou graphes sémantiques. Les noeuds du graphe représentent alors les concepts, tandis que les arcs représentent les relations inter-conceptuelles [BRACHMAN 1979]. On citera le langage KL-ONE de Brachman et Schmolze [1985] et les travaux plus récents de Xie et Passonneau [2015] portés sur une structure sémantique en graphes.

Les cadres sémantiques viennent de la théorie de Fillmore [1976], développant l'idée que l'on ne puisse pas comprendre le sens d'un seul mot sans avoir accès à toutes les connaissances essentielles qui s'y rapportent, et donc à son contexte complet.

Un des principaux travaux mené avec ce type de représentation sémantique porte le nom de FrameNet [BAKER et al. 1998]. Ce projet vise à généraliser les cadres sémantiques, généralement définis pour une tâche précise. On notera aussi que des travaux ont été menés sur l'ensemble de données MEDIA [BONNEAU-MAYNARD, ROSSET et al. 2005], utilisé dans cette thèse, avec une représentation en cadres sémantiques [MEURS 2009].

Wang et al. [2006] donnent un exemple de trois cadres sémantiques pour l'ensemble de données ATIS comme présenté en Figure 2.2. On peut voir certains de ces cadres comme des classes à instancier avec les mots de la transcription, lorsque ceux-ci répondent aux pré-requis définis en amont dans les champs *filler*. Certains cadres seront instanciés par d'autres cadres, comme c'est le cas pour `ShowFlight` dans l'exemple de la Figure 2.3.

```
<frame name="ShowFlight">
    <slot name ="Flight" filler="Flight">
</frame>

<frame name="Flight">
    <slot name="DepartureCity" filler="City">
    <slot name="ArrivalCity" filler="City">
</frame>
```

FIGURE 2.2 – Exemple de cadres sémantiques dans ATIS [Y. WANG, ACERO, MAHAJAN et al. 2006].

```
<ShowFlight>
    <Flight>
        <DepartureCity filler="City"> Seattle </DepartureCity>
        <ArrivalCity filler="City"> Boston </ArrivalCity>
    </Flight>
</ShowFlight>
```

FIGURE 2.3 – Exemple d'instanciation de cadres sémantiques dans ATIS [Y. WANG, ACERO, MAHAJAN et al. 2006].

Comme évoqué précédemment, le formalisme de QUAERO propose aussi l'imbrication d'entités nommées [GROUIN et al. 2011]. Chaque concept peut donc être décomposé par composants et contenir d'autres concepts. L'ensemble de données ETAPE utilise aussi ce formalisme [GRAVIER, ADDA et al. 2012]. Cette structure peut s'apparenter à un arbre de concepts, comme proposé par Raymond [2013].

2.2 Systèmes en cascade

Les systèmes visant la compréhension de la parole via l'extraction de sémantique, entités nommées ou concepts, peuvent être composés de plusieurs modules réalisant un enchaînement de

traitements successifs. Cet assemblage de modules est souvent référé en tant que système en cascade. Un premier module peut alors réaliser le Traitement Automatique de la Parole (*Automatic Speech Recognition, ASR*) pour transcrire le signal audio sous un format textuel, tandis qu'un second module de Compréhension du Langage Naturel (*Natural Language Understanding, NLU*) réalisera la tâche d'étiquetage de cette transcription. Les deux composants sont alors optimisés séparément, l'un grâce aux informations acoustiques du signal de parole, et l'autre grâce aux informations textuelles transcrites.

L'intérêt principal de cette approche est de diviser la tâche de Compréhension Automatique de la Parole en deux sous-tâches moins complexe à l'aide d'une représentation intermédiaire.

2.2.1 Reconnaissance de la parole

La Reconnaissance Automatique de la Parole vise à extraire les informations lexicales contenues dans le signal de parole afin de produire une transcription textuelle. Pour les systèmes de reconnaissance de la parole présentés dans cette section, on parlera de traiter un flux de parole continu avec pour objectif de produire une séquence de mots.

Bien que souvent traitée par une approche statistique comme introduite par Jelinek en 1976, on pourra citer le premier modèle de reconnaissance automatique de parole discontinue à base de règles proposé par Davis et al. [1952], réalisant la transcription de chiffres prononcés par un locuteur anglais.

On notera que les systèmes *ASR* sont faits de modèles acoustiques pouvant être combinés à un modèle de langue complémentaire, qu'il soit externe ou interne au système acoustique [BENGIO, DUCHARME et al. 2000 ; HORI, WATANABE, Y. ZHANG et al. 2017]. Les modèles de langue neuronaux seront décrits à la Section 2.2.2, car utilisés comme module principal de systèmes *NLU* dans des architectures en cascade.

Systèmes *ASR* non-neuronaux

Les modèles de Markov cachés (*Hidden Markov Model, HMM*) furent jusqu'à récemment fortement présents dans le domaine du traitement de la parole. Ces automates probabilistes permettent de réaliser une modélisation acoustique du signal de parole de manière statistique [RABINER 1989].

L'unité sous-lexicale utilisée pour le domaine acoustique est le phonème. C'est la plus petite unité sonore discriminante pouvant être utilisée pour représenter une séquence de mots. Chaque phonème sera modélisé par un *HMM* distinct, dont nous donnons un exemple de modélisation en Figure 2.4. La modélisation d'une phrase, séquence de mots eux-mêmes séquences de phonèmes, résultera donc d'un chaînage de plusieurs *HMM*.

Les états e d'un *HMM* ainsi que les *HMM* eux-mêmes seront reliés entre eux par des probabilités de transition unidirectionnelles, modélisant la probabilité de passer d'un phonème à un autre. Un bouclage sur le même état est aussi possible avec une probabilité qui lui est propre. Outre ces probabilités de transition, chaque *HMM* sera aussi caractérisé par des probabilités d'émission d'observation o et une probabilité initiale qui permettra de définir par quel phonème commence la séquence. L'entraînement du modèle statistique se fera de manière itérative en ajustant ces probabilités avec l'aide d'un corpus d'apprentissage et d'un calcul de maximum de vraisemblance comme celui de l'algorithme d'Espérance-Maximisation (*Expectation Maximization, EM*) [DEMPSTER et al. 1977].

Il s'agira ensuite de trouver le meilleur chemin parmi les transitions entre *HMM*, chaque chemin représentant une séquence de mots possible. On pourra alors utiliser un algorithme comme celui de Viterbi [FORNEY 1973] ou de Baum-Welch [BAUM 1972]. Le meilleur chemin sera celui avec la plus forte probabilité.

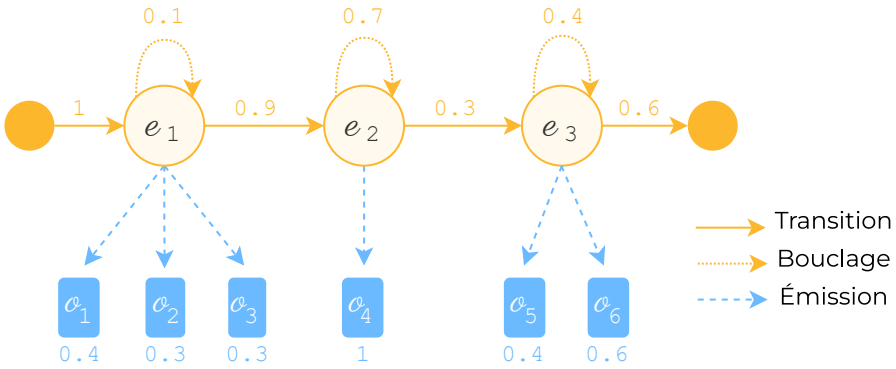

FIGURE 2.4 – Schématisation d'un *HMM*.

Les *HMM-GMM* sont des modèles *HMM* utilisant des fonctions Gaussiennes (*Gaussian Mixture Model, GMM*) afin de définir les probabilités d'émission d'observation du modèle [GAUVAIN et LEE 1994]. Une somme pondérée de Gaussiennes est réalisée afin d'en déterminer leur variance et moyenne.

Les premiers modèles de langue statistiques furent introduits par Shannon [SHANNON 1948], puis repris par Jelinek pour la construction d'un décodeur linguistique pour la reconnaissance de la parole [1976].

Un modèle de langue est initialement un modèle probabiliste visant à corriger la séquence de mots ou de caractères en sortie d'un modèle acoustique [HORI, WATANABE et HERSHEY 2017]. Lors de la prédiction de caractères, il permet de générer des mots hors vocabulaire, permettant de prédire des mots inconnus pour la tâche.

Le modèle de langue estime une probabilité d'apparition de chaque élément pour tout instant t en se fiant aux éléments précédemment prédits, évaluant les contraintes linguistiques de la séquence. Combiner les probabilités du modèle de langue à celles du modèle acoustique permet ainsi une meilleure cohérence dans la transcription ou dans l'annotation sémantique prédite. Le modèle acoustique est néanmoins appris simultanément, permettant de se fier principalement aux informations contenues dans le signal de parole.

Avec les avancées neuronales de ces dernières années, de nombreux modèles acoustiques se passent de modèle de langue [CHAN, JAITLY et al. 2016 ; L. DONG et al. 2018 ; PARK et al. 2019], produisant une séquence déjà suffisamment cohérente car modélisant directement les probabilités de leurs séquences de sortie.

Un modèle n-gramme [SHANNON 1951] est un modèle de langue permettant d'estimer les probabilités d'apparition d'un mot ou caractère dans une séquence en fonction de ses $n-1$ éléments précédents. La probabilité d'une séquence se traduit donc par le produit des probabilités de chaque élément joint à ses n-1 précédents, formant un groupe de n éléments.

Souvent compris entre 3 et 4, n ne dépasse que très rarement ces valeurs, la complexité du modèle augmentant exponentiellement [S. CHEN et GOODMAN 1999 ; BENGIO, DUCHARME et al. 2000]. Un n élevé a aussi l'inconvénient potentiel de devoir traiter des séquences de n éléments jamais vues lors de l'apprentissage. Par exemple, un n de 10 nécessiterait d'avoir vu la même séquence de 10 éléments lors de l'apprentissage pour une prise en charge optimale. Le cas contraire, sa probabilité d'apparition lors de la phase d'inférence serait de 0. Différentes techniques de lissage permettent de réduire ce problème [S. CHEN et GOODMAN 1999], généralement en attribuant des probabilités moindres mais non-nulles aux séquences jamais vues lors de l'apprentissage [KATZ 1987].

Systèmes *ASR* neuronaux

Faisant suite aux *HMM-GMM*, des modèles hybrides ont été proposés. Tout d'abord, on apprit les probabilités d'émission d'observation du *HMM* avec un perceptron multi-couches simple [BOULARD et WELLEKENS 1987 ; BOURLARD et MORGAN 1989]. La dernière couche d'un modèle neuronal simple calculait ainsi les probabilités à appliquer au *HMM*.

Par la suite, les *HMM-DNN* se sont démocratisés, utilisant des réseaux de neurones plus denses (*Dense Neural Network, DNN*), apportant de meilleurs résultats que des modèles entièrement non-neuronaux dans de nombreuses tâches de reconnaissance de la parole [SEIDE et al. 2011 ; HINTON et al. 2012 ; DAHL et al. 2012].

Il en va de même pour les réseaux de neurones à retardement (*Time-Delay Neural Network, TDNN*) qui donnèrent lieu à des *HMM-TDNN* [W. MA et VAN COMPERNOLLE 1990], ayant été démontrés utiles pour réaliser une transcription en phonèmes depuis le signal de parole [WAIBEL, HANAZAWA et al. 1989].

Avec l'apparition des modèles Transformers, ont aussi été proposés des modèles hybrides *HMM*-Transformer [Y. WANG, MOHAMED et al. 2020].

Bien que proposés dès les années 80 [LANDAUER et al. 1987 ; WATROUS et SHASTRI 1987 ; BOURLARD et MORGAN 1994], ce n'est que récemment que les modèles entièrement neuronaux prirent le dessus sur des modèles hybrides de type *HMM-DNN* dans le domaine de la Reconnaissance Automatique de la Parole. Ces approches nécessitent toujours néanmoins plusieurs étapes, dont une pré-segmentation en amont des données, ainsi qu'une extraction de leurs paramètres acoustiques, comme présentées au Chapitre 3.

On pourra notamment citer l'utilisation d'architectures récurrentes projetant directement la parole depuis une représentation cepstrale vers une séquence de caractères [GRAVES et JAITLY 2014 ; HORI, J. CHO et al. 2018].

Ces mêmes systèmes furent par la suite enrichis de couches convolutives qui permirent d'améliorer le traitement des paramètres acoustiques [AMODEI et al. 2016]. Certaines furent même uniquement constitués de couches convolutives [Y. ZHANG et al. 2016].

S'appuyant sur les technologies utilisées en Traduction Automatique, ont été proposées des architectures encodeur-décodeur avec mécanismes d'attention. La première proposition de reconnaissance de phonèmes faite par Chorowski et al. en 2014 ne surpassait alors pas les approches récurrentes. Les tentatives suivantes ont elles fait leurs preuves [CHOROWSKI, BAHDANAU, SERDYUK et al. 2015], se basant sur la reconnaissance de caractères en modifiant l'architecture du système [CHAN, JAITLY et al. 2016] ou la segmentation et contextualisation des données [BAHDANAU, CHOROWSKI et al. 2016], ou via l'utilisation conjointe d'une fonction de coût CTC [S. KIM et al. 2017] comme déjà proposé pour des architectures convolutives [HORI, WATANABE, Y. ZHANG et al. 2017].

Se plaçant alors à l'état-de-l'art du domaine de la reconnaissance de la parole avec des systèmes d'attention multi-têtes [C. CHIU et al. 2017], c'est naturellement que la recherche se tourna vers les architectures Transformer [L. DONG et al. 2018 ; A. LIU, S. YANG et al. 2019], via un apprentissage supervisé utilisant des *fbanks*, encodage du signal audio décrit au Chapitre 3. On pourra citer le système de Pham et al. [2019], surpassant toute autre architecture jusqu'alors proposée grâce à un total de 96 couches Transformer d'encodage et de décodage. Le traitement de la parole en flux continu fut aussi l'objet de l'utilisation de Transformers [MORITZ et al. 2020].

Se basant sur les avancées dans le domaine textuel avec des encodeurs tels que GloVe [PENNINGTON et al. 2014] et Word2Vec [MIKOLOV, K. CHEN et al. 2013 ; MIKOLOV, SUTSKEVER et al. 2013], les encodeurs de parole par apprentissage semi-supervisé furent proposés par Schneider et al. [2019] avec leur modèle wav2vec pour le traitement de la parole puis par Baevski et al.

avec wav2vec 2.0 [2020]. Suivirent les modèles multilingues XLSR [LAMPLE et CONNEAU 2019] et XLS-R [BABU et al. 2022], parfois multi-modaux comme c'est le cas pour Whisper [RADFORD, J. KIM et al. 2023]. Des modèles monolingues furent développés simultanément, comme les modèles LeBenchmark pour la reconnaissance du français [EVAIN, H. NGUYEN et al. 2021] menant à des résultats à l'état-de-l'art sur l'ensemble de données MEDIA [GHANNAY, CAUBRIERE, MDHAFFAR et al. 2021]. On pourra aussi citer le modèle anglais HuBERT [HSU et al. 2021], basé sur le fonctionnement du modèle de langue BERT [DEVLIN et al. 2019]. Nous décrivons les encodeurs de paroles en lien avec cette thèse dans le Chapitre 3.

Nous noterons que dans la recherche d'encodeurs de parole toujours plus performants, la quantité de données utilisées lors de l'apprentissage [PRATAP, TJANDRA et al. 2024] mais aussi le nombre de paramètres des modèles [RADFORD, J. KIM et al. 2023] est généralement en forte augmentation, tandis que la réalisation d'un apprentissage entièrement auto-supervisé reste complexe dans le domaine du traitement de la parole [PASCUAL et al. 2019].

2.2.2 Compréhension de la langue écrite

Un système en cascade pour l'extraction de concepts sémantiques est ensuite composé d'un module d'étiquetage de la transcription préalablement générée. Initialement réalisé à partir de règles probabilistes afin de formaliser des grammaires [KLATT 1977], puis utilisant des systèmes statistiques semblables aux *HMM* présentés plus tôt, le domaine *NLU* s'est tourné vers les solutions neuronales [SARIKAYA et al. 2011], avec notamment l'utilisation de modèles de langue de plus en plus imposants. Cette section présente brièvement ces différents systèmes d'étiquetage.

Systèmes *NLU* non-neuronaux

Les grammaires furent les premiers systèmes de compréhension du langage naturel [WEIZENBAUM 1966 ; DE MORI 1983 ; DOWDING et al. 1993]. Les règles qui les composent traitent les informations linguistiques afin de transformer le langage naturel en une représentation logique sur laquelle il est possible d'effectuer une analyse syntaxique et sémantique [DE MORI 2007]. Pour le traitement d'un langage fini on définit donc un nombre fini de règles pour la réalisation d'une grammaire formelle, dont nous donnons un exemple en Figure 2.5. Celles-ci guideront la syntaxe des phrases recherchées ainsi que les mots de vocabulaire à utiliser [CHOMSKY 1957 ; CHOMSKY 1965].

Il existe différents types de grammaires formelles. Les grammaires générales comprennent les grammaires contextuelles, qui comprennent les grammaires hors-contexte elles-même comprenant les grammaires régulières [CHOMSKY 1957]. Les grammaires hors-contexte sont les plus généralement utilisées pour le traitement du langage naturel. Lorsqu'il s'agira de traiter un dialogue

phrase = sujet + chambre + date + lieu.

sujet = pronom + verbe

chambre = nombre + type

date = préposition + déterminant + nombre + mois

lieu = préposition + ville

Je voudrais réserver une chambre simple pour le dix août à Avignon.

FIGURE 2.5 – Exemple de règles de grammaire.

de phrases finies et régulières, les grammaires régulières seront généralement utilisées [MOHRI et J. 2001].

Il existe aussi des grammaires probabilistes stochastiques, utilisées comme complément à un système afin d'en pondérer les hypothèses [STEPHANIE 1992]. Elles permettent ainsi de réduire les erreurs dues à des ambiguïtés. Cette pondération se fera à l'aide de poids, associés à chacune des règles qui la composent.

On notera le manque important de justesse des grammaires pour le traitement d'une phrase polysémique. Une même phrase peut être interprétée différemment, ce qui posera alors problème lors de son étiquetage [ALLEN et al. 2007].

Il est possible de combiner une grammaire régulière a un système plus imposant, comme par exemple à un automate à états finis (*Finite State Machine, FSM*). Un *FSM* est un modèle mathématique qui permet ainsi de représenter les connaissances linguistiques produites par une règle de grammaire pour un sous-ensemble fini du langage naturel et un vocabulaire fermé. Comme un *HMM*, les *FSM* vont représenter une grammaire via une suite d'états finis et leurs probabilités de transition. Chaque *FSM* représentant une règle précise de la grammaire, il en faudra autant que de règles pour en définir l'ensemble. Les chaînes formées permettront de déterminer l'appartenance d'une phrase au langage traité [RAYMOND 2005], lorsque le *FSM* pourra transitionner de l'état initial de la chaîne à son état final.

On parlera de transducteur à état fini pour les *FSM* réalisant une analyse grammaticale des symboles liés à chaque probabilité de transition. L'objectif sera de lier sémantiquement des ensembles de symboles. Cette classification permet ainsi de générer un étiquetage sémantique sur la transcription. La Figure 2.6 illustre la représentation d'une règle de grammaire pour un *FSM* classique et un transducteur.

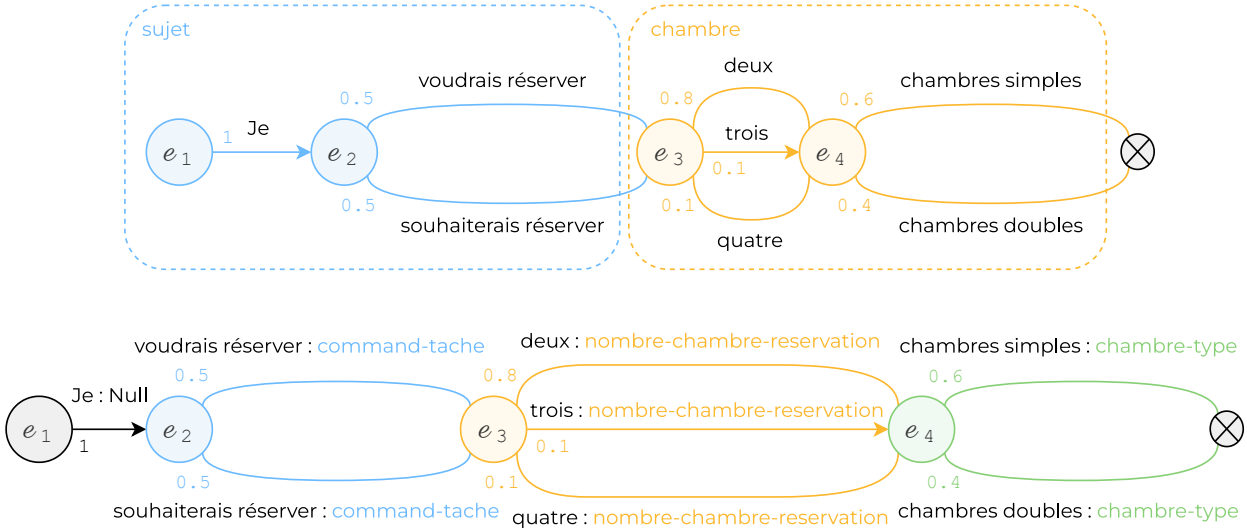

FIGURE 2.6 – Exemple de *FSM* pour une règle de grammaire régulière et d'un transducteur à état fini pour un exemple lié à la tâche sémantique de MEDIA [BONNEAU-MAYNARD, ROSSET et al. 2005].

Les machines à vecteur de support (*Support Vector Machines, SVM*), parfois appelés séparateurs à vastes marges [VAPNIK 2006 ; VAPNIK 2000] permettent de construire un classifieur grâce à l'apprentissage de valeurs réelles pour un problème non-linéaire. Ils servent ainsi de séparateur linéaire pour un problème linéairement séparable, cherchant les meilleurs hyperplans pour la classification des échantillons.

La majorité des problèmes de classification textuelle est linéairement séparable. Lorsque ce n'est pas le cas, une projection des données peut être réalisée pour les rendre exploitables par le *SVM*.

La classification peut donc consister premièrement à projeter les données dans un espace dimensionnel propice à leur séparation, via une méthode réalisant le produit scalaire des échantillons. Disposant alors d'un espace linéairement séparable, il s'agira de trouver les hyperplans optimaux afin de maximiser la marge entre les échantillons et ces hyperplans, comme illustré par la Figure 2.7.

Afin d'utiliser un *SVM* sur un segment textuel, il sera nécessaire de le transformer, par exemple avec la méthode communément utilisée du sac de mots (*bag of words*) [K. ZHANG et al. 2006]. Cette méthode permet de comptabiliser le nombre d'occurrences de chaque mot dans le segment. La dimension du vecteur ainsi que la position de cette information pour chaque mot dans celui-ci est fixée par un dictionnaire définit par avance [JOACHIMS 1998].

L'application des *SVM* à une tâche de compréhension de la parole équivaut à une succession de classifications, réalisées pour chaque mot de la séquence. Une pondération des classifications obtenues peut ensuite être réalisée [HAHN et al. 2011].

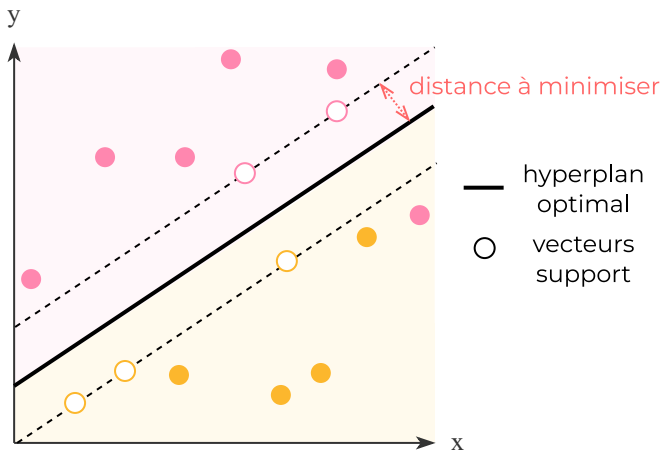

FIGURE 2.7 – Exemple de classification binaire par *SVM* avec maximisation de la marge entre échantillons et hyperplan.

Les champs aléatoires conditionnels (*Conditionnal Random Fields, CRF*) sont des modèles probabilistes de graphes non-orientés discriminants [LAFFERTY et al. 2001] de la famille des champs aléatoires de Markov. La structure du graphe permet de définir les dépendances entre chaque noeud. On ne parle ainsi plus de transition ni d'état. Ces modèles sont régis par la théorie fondamentale des champs aléatoires [HAMMERSLEY et CLIFFORD 1971] et peuvent être résolus par différents algorithmes comme celui de Viterbi cité plus tôt dans ce chapitre [MALOUF 2002].

Les *CRF* ont plusieurs avantages, en comparaison avec les modèles génératifs tels que les *FSM*, bien que plus coûteux que ces derniers. Le premier est leur faculté de contextualisation, utilisant l'ensemble d'un segment pour la génération de ses étiquettes car étant non-orienté. Cela a pour effet de réduire la dépendance stricte des mots à une étiquette, celle-ci n'étant pas attribuée en fonction d'un historique immédiat uniquement, mais en fonction de l'ensemble de la séquence. En revanche, ceci signifie qu'un *CRF* n'a pas la faculté du *FSM* à maximiser les probabilités de la séquence entière de manière conjointe. Ceci mène à un autre avantage concernant la résolution d'un biais d'étiquettes présent chez les modèles génératifs. Ceux-ci sont limités lorsqu'un état n'a que peu de transitions possibles, chaque probabilité de transition ne dépendant que de l'état précédent et suivant, ce qui n'est pas le cas dans un *CRF*.

Dans le domaine de la Compréhension Automatique de la Parole, bien que des *HMM* et autres réseaux bayésiens aient été proposés pour l'extraction de concepts sémantiques [SCHWARTZ et al. 1996 ; BONNEAU-MAYNARD et LEFÈVRE 2005 ; BUNDSCHUS et al. 2008] et la reconnaissance

d'entités nommées [McCallum et W. Li 2003], ce sont les *CRF* qui dominèrent l'état-de-l'art jusque récemment [Lefèvre 2007 ; Vukotic et al. 2015 ; Hahn et al. 2011], avant l'apparition de réseaux de neurones denses et complexes.

Systèmes *NLU* neuronaux

Afin de pouvoir fournir une transcription automatique et donc un texte à un réseau de neurones, il faut tout d'abord le représenter sous format vectoriel. Tout comme le sac de mot dont nous parlions plus tôt, il existe d'autres types de représentations.

La représentation *one-hot* consiste à modéliser une phrase de n mots avec n vecteurs. Chaque mot sera donc encodé par un vecteur binaire rempli de 0 et d'un unique 1 placé à la position lui correspondant suivant un dictionnaire défini par avance. Comme pour le sac de mot, ces vecteurs auront une dimension égale à la taille du vocabulaire de la tâche. Cette représentation a pour principal défaut d'être très volumineuse et pauvre linguistiquement et sémantiquement.

Nous citions GloVe [Pennington et al. 2014] et Word2Vec [Mikolov, K. Chen et al. 2013] plus tôt dans cette section. Les plongements de mots [Bengio, Ducharme et al. 2000] sont une autre méthode permettant entre autres de représenter les transcriptions en entrée d'un module *NLU* tout en préservant la richesse linguistique et sémantique des mots. Chaque mot sera alors encodé en un vecteur de valeurs réelles, dense et compact, qu'il sera possible de *fine-tuner* conjointement à l'apprentissage du module ou de simplement extraire d'un dictionnaire externe. Déjà utilisés pour l'apprentissage de modèles de langue [Schwenk 2007], ils furent rapidement employés pour la compréhension de la parole [Yao, Zweig et al. 2013 ; Mesnil, X. He et al. 2013].

Les premiers travaux pour la compréhension de la parole furent menées par Sarikaya et al. [2011] pour une tâche de routage d'appels. Leur exploitation fut ensuite proposée pour l'extraction d'une séquence de concepts sémantiques avec l'utilisation de modules de classification par Deng et al. [2012].

A suivi l'exploitation de modèles récurrents, tout d'abord pour des tâches purement *NLU* [Yao, Zweig et al. 2013], puis comme module *NLU* d'une approche *SLU* en cascade [Mesnil, X. He et al. 2013] via l'utilisation des récurrences de Elman [Elman 1990] et de Jordan [Jordan 1997]. La faculté des *RNN* à tirer profit d'informations pertinentes éloignées dans la séquence traitée a permis des avancés dans le domaine de la Compréhension Automatique de la Parole [P. Xu et Sarikaya 2014 ; Shi et al. 2015], avec la proposition de nouvelles récurrences [Dinarelli et Tellier 2016 ; Dinarelli, Vukotic et al. 2017]. Les couches *LSTM* ont elles-aussi été utilisées [Yao, Peng et al. 2014], avant de se tourner vers les couches *bi-LSTM* [Hakkani-Tür et al. 2016], bénéficiant d'une meilleure contextualisation [Mesnil, Dauphin et al. 2015].

D'autres systèmes furent proposés avec l'utilisation de couches convolutives pour la compréhension de la parole [P. Xu et Sarikaya 2013].

Le domaine se tourna ensuite vers les architectures encodeur-décodeur et les mécanismes d'attention [B. LIU et LANE 2016 ; S. ZHU et YU 2017 ; Y. WANG, TANG et al. 2018]. En parallèle, les modèles hybrides, mélangeant réseaux de neurones récurrents et une optimisation des résultats par *CRF*, menèrent l'état-de-l'art [KADARI et al. 2018]. Premièrement utilisés avec des couches récurrentes [SIMONNET, CAMELIN et al. 2015] comme pour une tâche de traduction automatique [BAHDANAU, K. CHO et al. 2015], les architectures encodeur-décodeur ne permettaient alors pas de rivaliser avec les modèles hybrides [VUKOTIC et al. 2015], avant que Simonnet et al. [2018] ne proposent une nouvelle approche via la simulation d'erreurs de transcriptions.

Ceux-ci se firent eux-mêmes rattraper par des modèles mêlant Transformer et *CRF* [L. ZHANG et H. WANG 2019].

Dernièrement, les modèles Transformer de Vaswani et al. [2017] devinrent l'état-de-l'art du domaine *NLU* grâce aux modèles de langue utilisant cette architecture.

Les modèles de langue neuronaux permettent de contourner les problématiques liées aux modèles de langue probabilistes en prenant en considération un lexique plus important [SCHWENK 2007]. Les mots sont alors projetés dans un espace continu afin d'en exploiter la similarité.

D'abord introduits avec des architectures simples comme des perceptrons multi-couches [BENGIO, DUCHARME et al. 2000 ; SCHWENK 2007], des modèles de langue furent proposés avec une architecture récurrente [MIKOLOV, KARAFIÁT et al. 2010 ; MIKOLOV, KOMBRINK et al. 2011 ; SUNDERMEYER et al. 2012] afin de faciliter l'utilisation des dépendances de longue distance. Peters et al. proposèrent eLMo [2018], un modèle de langue bi-directionnel avec une contextualisation sémantique, syntaxique et linguistiques des mots d'une séquence. Pour Bengio et al. [2000] puis Schwenk [2007] il s'agit alors de prédire un mot suivant les mots le précédant. Pour ce faire, les mots sont projetés dans un espace latent. L'intérêt principal des modèles de langue neuronaux est la similarité de la projection pour deux mots sémantiquement proches, permettant une généralisation plus aisée.

Avec l'arrivée des architectures Transformer, de nombreux modèles virent le jour. On pourra citer les déclinaisons des modèles BERT et mBERT de Devlin et al. [2019], pré-appris sur une tâche de modélisation de la langue puis proposés *fine-tunés* pour différentes tâches dont de l'extraction sémantique. Parmi ces variantes, RoBERTa [Y. LIU, OTT et al. 2019] est proposé comme amélioration de BERT, CamemBERT [MARTIN et al. 2019] et FlauBERT [LE et al. 2020] se focalisent sur le français et LaBSE [FENG et al. 2022], présenté au Chapitre 3, propose une version multilingue agnostique à la langue. D'autres modèles de langue furent proposés comme XLM [CONNEAU et LAMPLE 2019] et XLM-R [CONNEAU, KHANDELWAL et al. 2020], modèles cross-lingues issus de diverses approches d'apprentissage sur une très grande quantité de données.

Déjà utilisés pour des tâches de traitement de la parole [S. CHIU et B. CHEN 2021 ; HERVÉ et al. 2022 ; PELLOIN, DARY et al. 2022], de nombreux travaux utilisèrent ces modèles Transformer pour

la compréhension de la parole, y compris pour des tâches d'extraction de concepts sémantiques [KORPUSIK et al. 2019 ; GHANNAY, SERVAN et al. 2020 ; GHANNAY, CAUBRIERE, MDHAFFAR et al. 2021 ; CATTAN et al. 2022].

Les larges modèles de langue (*Large Language Model, LLM*) sont des modèles de langue neuronaux très denses, dont le nombre de paramètres se compte en milliards ou billions [BENDER et al. 2021]. Parmi ces modèles nous pouvons citer les modèles GPT [RADFORD et NARASIMHAN 2018 ; RADFORD, J. WU et al. 2019] comme GPT-3 [T. BROWN et al. 2020], PaLM [CHOWDHERY et al. 2024], LLaMA [TOUVRON et al. 2023], Alpaca [TAORI et al. 2021], Megatron [SHOEYBI et al. 2019] et FLAN [WEI et al. 2021]. Ces *LLM* sont réunis dans la Figure 2.8, démontrant par ailleurs le monopole de certains laboratoires dans ce domaine.

Les *LLM* soulèvent de nombreuses interrogations, donnant parfois lieu à des analyses poussées de certains modèles comme les modèles GPT [YENDURI et al. 2023]. C'est notamment le cas d'un point de vue environnemental [BENDER et al. 2021], déjà discuté pour des modèles de langue moins denses [LE SCAO et al. 2022], mais aussi concernant leur réelle faculté à comprendre la langue [ARCAS 2022 ; MITCHELL et KRAKAUER 2022]. Sur ce dernier point, les avis divergent.

2.3 Systèmes de bout-en-bout

Les systèmes de bout-en-bout (*end-to-end*) permettent de viser la Compréhension Automatique de la Parole (*Spoken Language Understanding, SLU*) avec un unique système. Celui-ci prendra en entrée les données acoustiques issues du signal de parole brut ou après un premier traitement. Il peut s'agir d'un traitement en MFCC ou *fbanks* [S. DAVIS et MERMELSTEIN 1980], comme présentés au Chapitre 3. Il générera les sorties attendues pour la tâche *SLU*. Pour une annotation en concepts sémantiques ou reconnaissance d'entités nommées, il s'agira alors de prédire une séquence, parfois accompagnée de la transcription automatique du signal de parole.

Cette transformation directe d'une représentation acoustique à une représentation sémantique fut longtemps considérée trop complexe [SERDYUK et al. 2018], la recherche se tournant donc vers les systèmes en cascade réalisant une chaîne de traitements successifs [RAYMOND et RICCARDI 2007]. Une de ces problématiques est liée aux faibles quantités de données disponibles pour les tâches *SLU*, malgré des propositions d'ensemble de données voyant le jour pour diverses sous-tâches [BASTIANELLI et al. 2020]. À contrario, des quantités importantes de données [GAUVAIN, LAMEL et al. 2002 ; RADFORD, J. KIM et al. 2023] peuvent être utilisées pour des tâches d'*ASR*, premier module d'une architecture en cascade. Cela fait de ces modules *ASR* des systèmes de transcription globalement très performants, ne pouvant être égalés par des systèmes de bout-en-bout *SLU* réalisant une transcription conjointe à leur tâche de compréhension de la parole.

Pourtant, les architectures de bout-en-bout offrent un grand nombre davantages. Dans un système en cascade, les erreurs de transcription du module *ASR* [Y. GONG 1995 ; GHANNAY 2017] se

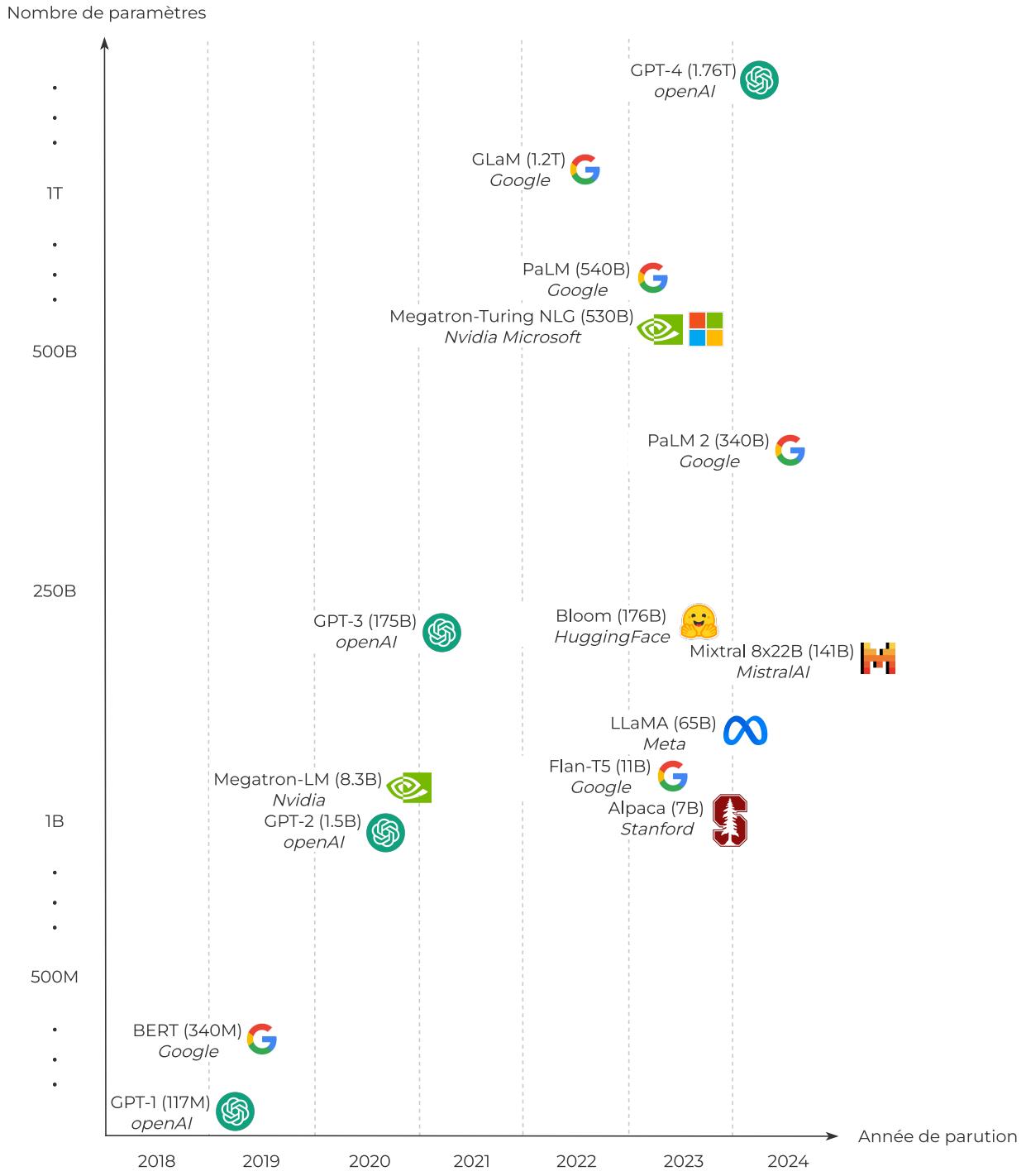

FIGURE 2.8 – Mise en avant du nombre croissant de paramètres utilisés pour l'apprentissage de *LLM* et des laboratoires dominant leur production via une sélection de *LLM* populaires et représentatifs du milieu.

propagent au module *NLU* [Y. WANG, ACERO et CHELBA 2003]. Cela signifie qu'il n'est pas possible de rectifier ces erreurs lors de l'apprentissage sémantique, mais aussi que le module *NLU* n'apprendra pas sur des transcriptions propres. Afin d'améliorer la robustesse des systèmes en cascade face à l'erreur, des simulations d'erreurs de transcription ont été réalisées [SIMONNET, GHANNAY, CAMELIN, ESTÈVE et DE MORI 2017 ; SIMONNET, GHANNAY, CAMELIN et ESTÈVE 2018]. Ces études permirent de démontrer l'impact de cette propagation d'erreurs au sein du module *NLU*, ne pouvant totalement contourner le problème.

Un second intérêt repose sur l'exploitation des informations acoustiques contenues dans le signal de parole. Un module *NLU* n'apprendra à extraire la sémantique qu'à partir d'une transcription textuelle. Hors, les informations paralinguistiques telles que la prosodie peuvent elles aussi être utilisées afin d'extraire le sens de la parole [P. PRICE et al. 1991 ; TRAN et al. 2017 ; SHRIBERG 2005].

Enfin, un dernier intérêt est lié à l'apprentissage unique nécessaire pour la réalisation d'un système *SLU* de bout-en-bout. Cela signifie que la tâche est visée dès le traitement des données acoustiques, adaptant directement leur projection pour les projeter dans un espace sémantique. Pour un système en cascade, le module *ASR* ne connaît pas la tâche *NLU*, et n'est donc pas adapté pour cette tâche.

Après que des approches de bout-en-bout se soient généralisées pour le Traitement Automatique de la Parole [HANNUN et al. 2014 ; AMODEI et al. 2016 ; Y. ZHANG et al. 2016], l'intérêt crût pour les appliquer au domaine de la Compréhension Automatique de la Parole. Bien que des systèmes mélangeant reconnaissance et compréhension de la parole furent proposés par le passé [ESTÈVE, RAYMOND et al. 2003], les architectures de bout-en-bout ne se popularisèrent réellement que plus tard pour la reconnaissance d'entités nommées [HORI et NAKAMURA 2006 ; HATMI et al. 2013], la détection d'intention [SERDYUK et al. 2018] et l'extraction de concepts sémantiques [GHANNAY, CAUBRIERE, ESTÈVE et al. 2018], le tout via l'utilisation d'approches hybrides [X. MA et HOVY 2016 ; CAUBRIERE, ROSSET et al. 2020] ou purement neuronales [GHANNAY, CAUBRIERE, MDHAFFAR et al. 2021].

Afin de concurrencer les performances des systèmes en cascade pour le traitement de la parole, une solution est de réaliser une augmentation de données via une synthèse de la parole [DESOT et al. 2020 ; LUGOSCH, MEYER et al. 2020 ; Y. HUANG et al. 2020]. Nous présenterons ici deux autres méthodes, dont le transfert d'apprentissage et l'utilisation d'encodeurs de paroles. Ces trois méthodes peuvent être utilisées simultanément.

Systèmes *SLU* via transfert d'apprentissage

Les systèmes de compréhension de la parole de bout-en-bout peuvent être obtenus en réalisant un transfert d'apprentissage [TOMASHENKO et al. 2019 ; BHOSALE et al. 2019 ; Y. HUANG

et al. 2020]. Ceci permet d'utiliser les quantités plus importantes de données disponibles pour des tâches proches en tentant de contrer le manque de données de la tâche cible. Il s'agira donc de réaliser un premier pré-apprentissage sur une tâche souvent moins complexe mais suffisamment proche de la tâche finale visée, puis de *fine-tuner* le modèle obtenu sur cette dernière. Pour ce faire, seules les dernières couches du modèle seront nécessairement ré-initialisées afin de pouvoir générer les sorties attendues pour la tâche finale, qu'elle soit une reconnaissance d'entités nommées, une extraction de concepts sémantiques, ou autre. Il sera aussi utile de ré-ajuster le taux d'apprentissage à chaque étape de transfert.

Ce principe de transfert d'apprentissage d'une tâche plus simple vers une tâche plus complexe peut s'apparenter à l'apprentissage par Curriculum (*Curriculum learning*) proposé par Bengio et al. [2009], souvent référée dans ces travaux. Cette technique consiste à ordonner les données d'apprentissage au sein d'un même corpus de la plus simple à la plus compliquée à traiter.

Ghannay et al. proposèrent ainsi de réaliser un transfert d'apprentissage depuis une tâche de Traitement Automatique de la Parole vers une tâche de reconnaissance d'entités nommées [2018]. Ceci leur permit de contourner la faible quantité de données annotées en entités nommées qui leur était disponible. Uniquement la couche de sortie Softmax fut réinitialisée entre les deux apprentissages.

De la même manière, des travaux ont mis en avant la possibilité de réaliser un transfert d'apprentissage depuis une tâche de Traitement Automatique de la Parole vers une tâche d'extraction de concepts sémantiques [PELLOIN, CAMELIN et al. 2021].

Les entités nommées pouvant être considérées comme des concepts sémantiques basiques [NOUVEL et al. 2015], des travaux se sont aussi penchés vers le transfert d'apprentissage depuis une tâche de reconnaissance d'entités nommées vers une tâche plus spécifique d'extraction de concepts sémantiques [CAUBRIERE, TOMASHENKO et al. 2019]. Bien que l'architecture proposée ne fut composée que de modules récurrents et convolutifs avec utilisation d'une *CTC*, ce transfert d'apprentissage permit alors d'atteindre l'état-de-l'art de certaines tâches, jusqu'alors obtenu via des architectures en cascade. Ces travaux prouvent aussi l'importance de l'ordonnancement des étapes de transfert. Ils démontrent qu'il est plus efficace pour l'extraction de concepts sémantiques d'apprendre une tâche de transcription de la parole avant celle de reconnaissance d'entités nommées, et non l'inverse.

Systèmes *SLU* avec encodeurs de parole

Le développement des réseaux de neurones conduisit à une large amélioration des modèles de bout-en-bout pour la compréhension de la parole via diverses tâches [LUGOSCH, RAVANELLI et al. 2019 ; DINARELLI, KAPOOR et al. 2020 ; Y. QIAN et al. 2017 ; PALOGIANNIDI et al. 2020].

C'est aussi le cas pour l'apprentissage auto-supervisé dont nous parlions au Chapitre 1, qui

impacta la grande majorité des sous-domaines du traitement de la parole via l'utilisation d'encodeurs de parole appris sur une vaste quantité de données [P. WANG et al. 2020]. Ces modèles menèrent notamment à des systèmes de bout-en-bout à l'état-de-l'art pour la compréhension de la parole [A. LIU, S. LI et al. 2021 ; DINARELLI, NAGUIB et al. 2022] dont des tâches d'extraction de concepts sémantiques [S. YANG et al. 2021 ; HAGHANI et al. 2018], bien que souvent moins performants que des systèmes en cascade [GHANNAY, CAUBRIERE, MDHAFFAR et al. 2021].

Cette méthode d'apprentissage fut démontrée très utile pour le traitement de langues peu dotées ou petits ensembles de données [HSU et al. 2021 ; BAEVSKI, ZHOU et al. 2020], ouvrant le champs des possibles concernant leur traitement. Des modèles monolingues furent aussi proposés, notamment pour le traitement et la compréhension du français [EVAIN, H. NGUYEN et al. 2021], menant une fois de plus à une amélioration des performances dans plusieurs sous-tâches [EVAIN, M. NGUYEN et al. 2021]. On notera aussi l'intention d'utiliser les modèles textuels existants afin d'améliorer les performances des modèles de compréhension de la parole [CHUNG, C. ZHU et al. 2021 ; AGRAWAL et al. 2020 ; MÜLLER et al. 2021], comme déjà réalisé pour la reconnaissance de la parole [ZHENG et al. 2021].

On peut considérer le coût en calcul important pour la réalisation d'un système de bout-en-bout via transfert d'apprentissage. Néanmoins, les systèmes utilisant un encodeur de parole sont eux aussi coûteux, nécessitant à la fois le pré-apprentissage de cet encodeur auto-supervisé sur une vaste quantité de données mais aussi son *fine-tuning* sur la tâche ciblée [STRUBELL et al. 2019 ; BENDER et al. 2021 ; PARCOLLET et RAVANELLI 2021].

Cette thèse se positionne dans un contexte de réalisation de systèmes de bout-en-bout avec encodeurs de parole. L'objectif est à la fois lié au gain de performances pour des tâches complexes de compréhension de la parole mais aussi à la considération des coûts de calculs qu'impliquent de tels systèmes.

2.4 Portabilité cross-lingue pour l'extraction sémantique

L'accessibilité aux nouvelles technologies et aux services qui leurs sont liés est incontestablement cruciale afin de donner accès à l'information de façon universelle et équitable. Pour les technologies liées à la parole, ce besoin l'est d'autant plus pour des personnes ne pouvant écrire ou lire, ou dans des situations qui ne s'y prêteraient pas. Il est donc important de ne pas limiter ces technologies à certaines langues et domaines précis. L'objectif final est de pouvoir proposer des systèmes performants pour un maximum de langues et d'applications possibles. Pourtant, cet objectif se heurte à des restrictions en terme de quantités de données disponibles, nécessaires pour le développement de ces systèmes de traitement de la parole. Une solution à cela réside dans l'utilisation d'ensembles de données issus de langues et domaines différents de la tâche ciblée.

On parlera alors de modèles multilingues, cross-lingues, et de la réalisation d'un transfert de connaissances via une portabilité de ces systèmes. Ces différentes méthodes sont décrites dans cette section. Nous y ferons aussi le rapprochement avec le développement de systèmes cross-modaux, notamment pour des tâches de traduction de la parole.

Cette thèse se focalise sur l'utilisation de ces derniers dans un contexte de portabilité cross-lingue et multilingue pour la compréhension de la parole d'ensembles de données peu dotés.

2.4.1 Cross-linguisme et multilinguisme

Un modèle cross-lingue vise la projection dans un espace vectoriel proche de mots sémantiquement identiques bien qu'issus de langues différentes. Le modèle sera donc entraîné sur de multiples langues avec pour objectif une minimisation de l'écart de représentation des mots ayant la même signification [Conneau et Lample 2019 ; Conneau, Khandelwal et al. 2020]. On peut alors parler d'un objectif de génération d'un interlingua, autrement dit une représentation commune à toutes les langues traitées, une langue universelle bien que vectorielle.

Un modèle multilingue peut se définir par sa capacité à traiter des données issues de multiples langues, sans considération de l'espace vectoriel dans lequel le signal de parole sera projeté. Nous pouvons donc considérer un système cross-lingue comme un système multilingue, mais pas inversement, bien qu'il soit souvent probable que le modèle fasse lui-même le lien entre des mots porteurs de même sens. Ce n'est pour autant pas l'objectif principal d'un modèle uniquement multilingue.

Ces deux approches permettent l'utilisation d'un unique modèle contre celle de nombreux modèles monolingues. Elles rendent aussi possible l'amélioration du traitement d'ensembles de données considérablement petits, grâce à l'apport d'ensembles de données plus importants bien que différents en langue et éventuellement domaine.

Des modèles multilingues furent proposés pour le traitement du langage écrit [Devlin et al. 2019 ; Conneau, Khandelwal et al. 2020 ; Xue et al. 2020], menant à leur utilisation dans un contexte de compréhension du langage [Conneau, Lample et al. 2018 ; Hu et al. 2020 ; Ruder et al. 2021]. Ils ouvrirent la voie au traitement de langues peu dotées via l'utilisation de grandes quantités de données pour des langues plus communément traitées dans le domaine.

De la même manière, des systèmes de traitement de la parole se tournaient déjà vers la génération de représentations cross-lingues et multilingues, permettant d'améliorer les performances dans certaines langues cibles grâce à leur prédiction de phonèmes universels [Schultz 1998 ; H. Lin et al. 2009 ; Z. Wang et al. 2002]. On notera l'intérêt du multilinguisme déjà présent pour des systèmes de type *GMM* [Lu et al. 2014] mais aussi le traitement de langues peu dotées comme l'afrikaans [Imseng et al. 2014], bénéficiant de sa corrélation avec certains dialectes allemands [Heeringa 2008].

Plus récemment, grâce à la démonstration du cross-linguisme appliqué à la reconnaissance de la parole [FÉR et al. 2017 ; CONNEAU, BAEVSKI et al. 2020], les modèles issus d'auto-supervision [BAEVSKI, SCHNEIDER et al. 2020] furent eux-aussi appris de manière multilingue [KAWAKAMI et al. 2020] et cross-lingue [CONNEAU, BAEVSKI et al. 2020]. Ces modèles permettent ainsi d'utiliser au mieux les larges ensemble de données déjà disponibles dans de nombreuses langues.

Suivant cet élan, des systèmes de compréhension de la parole cross-lingues se sont développés, faisant suite aux premiers essais multilingues de systèmes à base de règles et de grammaires [KOMATANI et al. 2001 ; MENG et SIU 2002] et autres systèmes stochastiques [HAHN et al. 2011].

Le premier système de bout-en-bout multilingue pour la compréhension de la parole fut proposé selon notre connaissance par Müller et al. [2021]. Celui-ci se focalisait alors sur la détection d'intention dans des ensembles de données français, anglais et espagnol avec l'aide de modèles Transformers et d'une cross-modalité réalisée avec un modèle BERT [DEVLIN et al. 2019].

Des modèles cross-lingues de bout-en-bout pour des tâches de compréhension de la parole ont aussi vu le jour [X. ZHANG et L. HE 2021], se basant sur des encodeurs acoustiques comme l'XLSR [CONNEAU, BAEVSKI et al. 2020], déjà multilingue. Comme pour l'apprentissage de wav2vec 2.0 présenté au Chapitre 3, l'apprentissage contrastif a aussi été proposé afin d'éloigner et rapprocher au mieux les représentations vectorielles de différentes langues suivant leur sens [QIN et al. 2022], ne se basant pas uniquement sur la structure multilingue du modèle pour réaliser le cross-linguisme attendu.

2.4.2 Portabilité entre les langues

La portabilité d'un système de traitement de la parole peut se définir par sa capacité à être pré-appris puis adapté à une tâche, un domaine, ou une langue différente, ce pour un moindre coût. On parlera alors de transfert d'apprentissage (*transfer learning*) depuis un premier modèle appris sur un certain ensemble de données qui sera *fine-tuné* sur une autre tâche. On appellera *zero-shot* une inférence réalisée elle aussi suite au pré-apprentissage d'une tâche plus générique que celle ciblée [JOHNSON et al. 2017].

Comme pour le multilinguisme et le cross-linguisme, la portabilité des langues a été investiguée premièrement pour des tâches de reconnaissance de la parole, se focalisant sur la portabilité des modèles acoustiques [SCHULTZ et BLACK 2006] et celle des modèles de langue [AKBACAK et al. 2005 ; SARIKAYA 2008 ; LEFÈVRE, GAUVAIN et al. 2005]. Cette portabilité fut proposée pour un sous-ensemble précis d'un réseau de neurones par Huang et al. [2013] afin d'améliorer les performances d'une tâche monolingue, tandis que d'autres études tentèrent différentes approches, avec l'utilisation de modèles complètement cross-lingues [SWIETOJANSKI et al. 2012 ; MIAO et METZE 2013] ou multilingues [GHOSHAL et al. 2013 ; VU et al. 2014].

Dans le domaine de la compréhension de la parole, la portabilité des langues s'est tout d'abord développée pour des systèmes en cascade [GAO et al. 2005 ; SERVAN et al. 2010 ; JABAIAN et al. 2013].

La portabilité de la langue pour les systèmes de bout-en-bout suivit [TOMASHENKO et al. 2019 ; LUGOSCH, RAVANELLI et al. 2019], avec des propositions de pré-apprentissage multilingues et cross-lingues [SCHUSTER et al. 2018 ; DO et GASPERS 2019] ayant pour objectif principal d'améliorer les performances d'un système *fine-tuné* de manière monolingue [BHOSALE et al. 2019 ; R. PRICE 2020], souvent pour une langue peu dotée [W. CHEN et al. 2018 ; JIA et al. 2020].

L'intérêt principal de cette portabilité de la langue réside dans la prise en charge de langues et dialectes pour lesquels très peu de données sont disponibles [SAMSON JUAN 2015], d'autant plus dans un contexte d'extraction sémantique. Utiliser des langues plus ou moins proches pour réaliser un pré-apprentissage permet ainsi d'augmenter cette quantité de données. Cumulée à un apprentissage cross-lingue, cette approche peut donner de bons résultats dans un contexte d'inférence *zero-shot*, sans *fine-tuning* supplémentaire sur la tâche cible. Contrairement aux modèles proposés actuellement pour le traitement de la parole qui se basent sur d'importantes quantités de données pour performer efficacement, réaliser une portabilité de la sorte permet d'obtenir des résultats pertinents pour une quantité d'heures de parole annotée très faible.

Le second avantage de cette approche réside dans son coût d'apprentissage [WAIBEL, SCHULTZ et al. 2004 ; SCHULTZ et BLACK 2006] et son coût humain, nécessitant de plus petits ensembles de données annotées, et pouvant se baser sur l'apprentissage de grands ensembles de données non-annotées.

2.4.3 Cross-modalité et Traduction Automatique

La traduction automatique (*Machine Translation, MT*) est une tâche consistant à traduire le langage naturel écrit ou parlé depuis une langue source vers une langue cible. La tâche d'extraction de concepts sémantiques peut être vue comme une tâche de traduction depuis une langue source naturelle (un signal de parole) vers une langue cible conceptualisée (sa transcription annotée sémantiquement). C'est de ce point de vue que part l'idée de pouvoir utiliser certains modules réalisés pour une tâche de traduction dans un but d'améliorer les performances de systèmes de compréhension de la parole.

Afin de tirer parti des modèles multilingues et cross-lingues appris de manière auto-supervisée pour le traitement du langage écrit [DEVLIN et al. 2019], il est possible de réaliser une cross-modalité dans le but d'améliorer les performances de systèmes destinés au traitement de la parole et à sa compréhension. L'objectif sera de pousser une représentation vectorielle acoustique à s'approcher d'une représentation vectorielle textuelle correspondante [AKKUS et al. 2023]. Souvent apprise sur bien plus de données [S. WU et al. 2019 ; AGRAWAL et al. 2020], la représentation

textuelle aura bénéficié d'un enrichissement sémantique important pour la résolution de tâches de compréhension voire de traduction de la parole. On souhaitera donc obtenir un espace latent cross-modal (*Cross-Modal Latent Space, CMLS*) [KUDUGUNTA et al. 2019] permettant l'enrichissement sémantique ou linguistique de la représentation acoustique.

Une combinaison de transfert d'apprentissage, augmentation des données et cross-modalité textuelle peut être mise en place afin d'améliorer les performances d'un système de compréhension de la parole [Y. HUANG et al. 2020]. Avec mSLAM [BAPNA et al. 2022], il est aussi question de tirer partie du multilinguisme et cross-linguisme de modèles contextuels [CHUNG, Y. ZHANG et al. 2021 ; JOSHI et al. 2020] en appliquant une fonction de coût *CTC* [GRAVES, FERNÁNDEZ et al. 2006] sur chaque modalité. Conjointement, des travaux ont été menés afin d'utiliser au mieux, dans des systèmes de compréhension de la parole de bout-en-bout, des enregistrements audio non-étiquetés destinés à une tâche de reconnaissance de la parole [DENISOV et VU 2023].

Dans cette thèse, nous cherchons à tirer partie de modèles sémantiques multilingues performants afin de viser la compréhension de la parole. Ces derniers peuvent être initialement appris pour le traitement de tâches de traduction automatique. Le fait de se pencher vers ces modèles de traduction implique qu'un certain cross-linguisme a déjà été ciblé, facilitant la portabilité entre certaines langues. L'engouement pour le transfert d'apprentissage entre langues peut s'expliquer de part les récentes avancées du domaine de la traduction automatique et l'aptitude de ses modèles à s'adapter à de multiples autres tâches [M. LUONG et al. 2016 ; ERIGUCHI et al. 2017 ; D. DONG et al. 2015].

Un bref historique des systèmes de traduction automatique avant les années 2000 a été proposé par Dorr et al. [1999]. Ceux-ci étaient initialement conçus pour traduire de manière littérale mot à mot, sans prise en compte de règles linguistiques, avant d'évoluer vers les systèmes de traduction que nous connaissons aujourd'hui. Tout comme pour les autres tâches de traitement de la parole, il fut question d'utiliser des grammaires [KAPLAN et BRESNAN 1982 ; HAYASHI et al. 2010] et des systèmes empiriques [LANGLAIS et al. 2008] et statistiques [P. BROWN, COCKE et al. 1990 ; P. BROWN, DELLA PIETRA et al. 1993]. Des architectures en cascade étaient généralement employées [MATUSOV et al. 2005 ; NEY 1999], couplant encodeur de parole et module de traduction multilingue [Y. LIU, GU et al. 2020 ; NLLB-TEAM 2022], mais une tendance s'est développée pour des approches sans transcription intermédiaire [DUONG et al. 2016 ; ANASTASOPOULOS, CHIANG et DUONG 2016], menant à des architectures de bout-en-bout [BÉRARD et al. 2018]. Les approches neuronales ont par ailleurs bénéficié de structures récurrentes et architectures encodeur-décodeur avec mécanismes d'attention [K. CHO, MERRIENBOER, GULCEHRE et al. 2014 ; K. CHO, MERRIENBOER, BAHDANAU et al. 2014 ; BAHDANAU, K. CHO et al. 2015 ; ANASTASOPOULOS et CHIANG 2018].

Parallèlement au développement des systèmes de traduction depuis la parole, plusieurs en-

codeurs cross-lingues textuels [SCHWENK et DOUZE 2017 ; ARTETXE et SCHWENK 2019] furent proposés avec l'avènement des Transformers [VASWANI et al. 2017]. Ces encodeurs sont considérés enrichis sémantiquement, pouvant générer une représentation vectorielle agnostique à la langue [FENG et al. 2022].

Tout comme la cross-modalité fut appliquée à la compréhension de la parole, utiliser les connaissances cross-lingues apprises par ces systèmes permet de pallier le besoin important de données nécessaires pour apprendre des systèmes de traduction depuis la parole. Le défi principal de cette tâche réside principalement dans le fait d'obtenir des enregistrements audio traduits dans de multiples langues. Cependant, une autre difficulté survient lors de l'utilisation de modèles textuels. Ceux-ci sont généralement conçus pour produire un vecteur par phrase [CER et al. 2018 ; REIMERS et GUREVYCH 2019 ; CONNEAU, KIELA et al. 2017 ; ARTETXE et SCHWENK 2019]. Or, en traitement de la parole, c'est plus communément un vecteur par trame audio qui est générée. Pour autant, de récents travaux ont pu tirer d'importants bénéfices d'une telle cross-modalité pour la tâche de traduction de la parole. Certains eurent recours à un sous-échantillonage avec mécanismes d'attention, permettant de former une unique représentation vectorielle par segment de parole [KHURANA et al. 2022]. D'autres méthodes existent, comme par exemple lors de l'utilisation d'un module de décodage. Dans ce cas-ci, l'encodeur réalisera une réduction de dimensionnalité, créant un goulet d'étranglement qui formera la représentation vectorielle à comparer à celle issue d'un traitement textuel [NLLB-TEAM 2022]. Dans tous ces cas de figure, le module d'encodage textuel peut alors être *fine-tuné* ou non, donnant ce qu'on appelle un apprentissage *teacher-student*. Le *teacher*, dont les paramètres sont figés, servira à apprendre le *student*.

2.5 Conclusion

Ce chapitre présente les principales tâches de compréhension de la parole ainsi que l'évolution des systèmes essentiellement liés à la détection d'intentions, l'extraction de concepts sémantiques et la reconnaissance d'entités nommées. Il met en avant la volonté de la communauté de recherche à se tourner vers des modèles de bout en bout, bien que ceux-ci ne soient à ce jour pas nécessairement plus performants que les systèmes en cascade. Le chapitre définit l'utilité des modèles de langue et divers encodeurs pour la réalisation de tels systèmes.

Enfin, nous faisons le lien entre traduction automatique et avancées multilingue et cross-lingue dans le domaine de la compréhension de la parole, notamment via des systèmes utilisant une cross-modalité textuelle. Ce chapitre servira ainsi de base contextuelle aux tâches traitées durant cette thèse, celle-ci portant sur l'utilisation de systèmes de bout-en-bout réalisant une portabilité cross-lingue.

L'approfondissement du fonctionnement de certains encodeurs, parfois initialement proposés pour une tâche de Traduction Automatique de la Parole, est réalisé au Chapitre suivant.

Chapitre 3

Encodage de la parole

Sommaire

Afin de viser la compréhension de la parole, il est nécessaire de traiter le signal de parole. Pour cela différentes méthodes existent, toutes ayant pour objectif de fournir une représentation vectorielle condensée et pertinente pour la tâche visée. On appelle ces représentations des plongements de mots (*embeddings*) lorsqu'elles seront issues d'un encodeur neuronal de parole.

Initialement, le signal de parole est capturé par un microphone dont les fréquences et amplitudes sont enregistrées dans un fichier audio brut grâce à un échantillonage temporel. Cette thèse traite d'enregistrements au format WAV (*waveform audio file format*) qui limitent les pertes qualitatives car ne réalisant pas de compression.

Un encodeur de parole utilise directement ces données acoustiques. Il peut cependant être nécessaire de réaliser une première paramétrisation du signal à partir d'approches plus classiques qui ne nécessitent aucun apprentissage. Parmi les méthodes de paramétrisation non-neuronales, nous pouvons citer les traitements fréquentiels tels que les spectrogrammes ou *MFCC* discutés en section 3.1.

3.1 Paramétrisation de la parole

Le signal de parole est considéré comme la variation temporelle de l'amplitude et de la fréquence d'une onde. Cette variation vient du changement de pression de l'air suite à l'émission d'un bruit. Bien qu'il soit possible de fournir à des réseaux de neurones complexes les ondes temporelles brutes d'un enregistrement audio, Davis et Mermelstein [1980] ont démontré que ce type d'information pouvait être aisément capturé lors d'une paramétrisation fréquentielle du signal. La transformée de Fourier permet de réaliser la bascule d'un espace de représentation temporel à fréquentiel, comme illustré par la Figure 3.1.

Dans un espace de représentation temporel, le signal audio est décrit comme une suite chronologique d'amplitudes. Le signal de parole, étant non-périodique, est cependant composé de suites d'ondes de fréquence équivalente que nous pouvons mettre en évidence avec l'aide d'une transformée de Fourier. Le résultat de cette transformée est appelé un spectre. En Compréhension Automatique de la Parole, ce traitement se fait sur un nombre fini d'échantillons. Il est donc de mise d'utiliser une transformée de Fourier discrète (TFD).

Afin de mettre en oeuvre un algorithme réalisant cette TFD, il est nécessaire de découper le signal audio en trames de parole (*frames*) d'un temps fixe. Celui-ci est généralement compris entre 10 et 30 millisecondes. Cette échelle vient du fait que la parole est considérée comme un signal stationnaire sur un temps très court. Ainsi, en approximativement 20 millisecondes, nous estimons traiter une trame dont la variation d'amplitude d'onde est minimale. De ce fait, bien que la vitesse de prononciation du locuteur influence la quantité d'informations contenues dans une trame, celle-ci

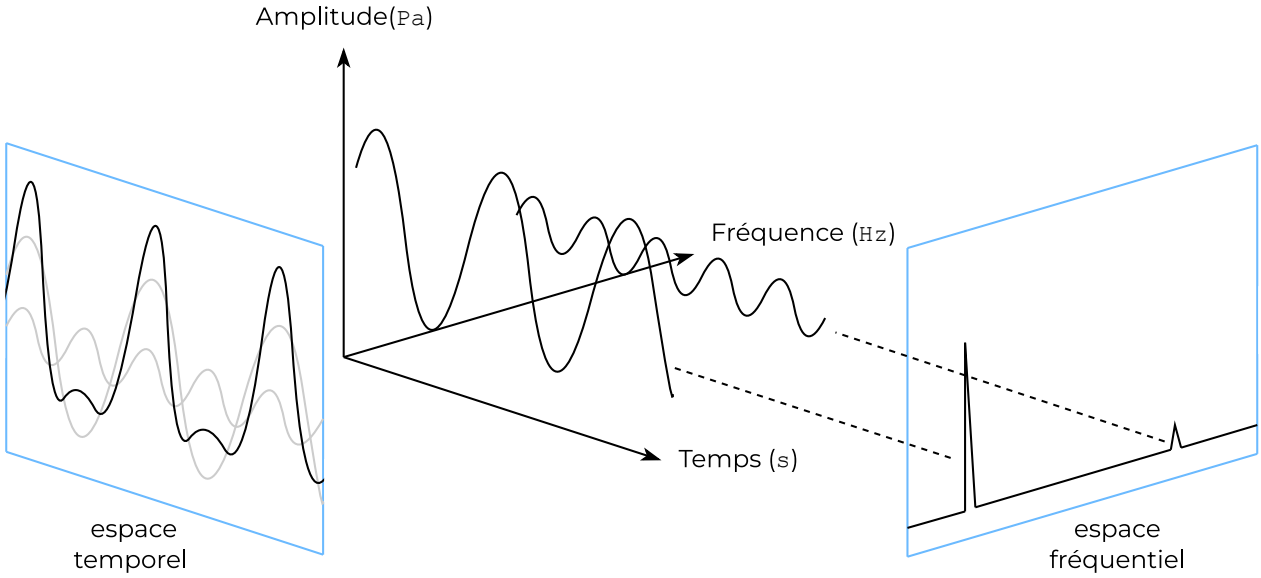

FIGURE 3.1 – Changement de l'espace de représentation d'une onde depuis un espace temporel à un espace fréquentiel par transformée de Fourier.

sera théoriquement suffisamment petite pour la capture d'un unique phonème. Ce sont ces trames qui pourront être directement fournies à nos modèles neuronaux encodant la parole.

Pour éviter un découpage arbitraire qui nuirait au traitement du signal de part sa non stationnarité réelle, une fenêtre glissante est utilisée pour réaliser un chevauchement entre les trames. Il s'agit de définir un pas de déplacement de la fenêtre, plus petit que la longueur de trame souhaitée, souvent de moitié. Il est courant d'utiliser un pas de 10 millisecondes pour une trame de 20 millisecondes. Ce principe est illustré par la Figure 3.2.

Une fonction de filtrage des éléments en bordure de fenêtre peut être appliquée pour optimiser le glissement. Avec de tels filtrages, les valeurs en bordure de trame impacteront moins intensément le traitement de celle-ci. Nous pouvons citer le filtrage de Hamming mais aussi ceux de Hann, Blackman et Bartlett.

Notons que dans le domaine de l'Apprentissage Profond, il est commun d'utiliser l'algorithme de transformée de Fourier rapide (*Fast Fourier Transform, FFT*) [BRIGHAM et MORROW 1967] afin de réaliser de manière efficace diverses paramétrisations du signal audio. Les représentations cepstrales résultantes sont utilisées pour la génération de spectrogrammes mais aussi de vecteurs de banques de filtres (*fbanks*), *MFCC* (*Mel Frequency Cepstrum Coefficients*) [S. DAVIS et MERMELSTEIN 1980], PLP (*Perceptual Linear Prediction*) [HERMANSKY 1990] ou LPCC (*Linear Prediction Cepstral Coefficients*) [MARKEL et GRAY 1976].

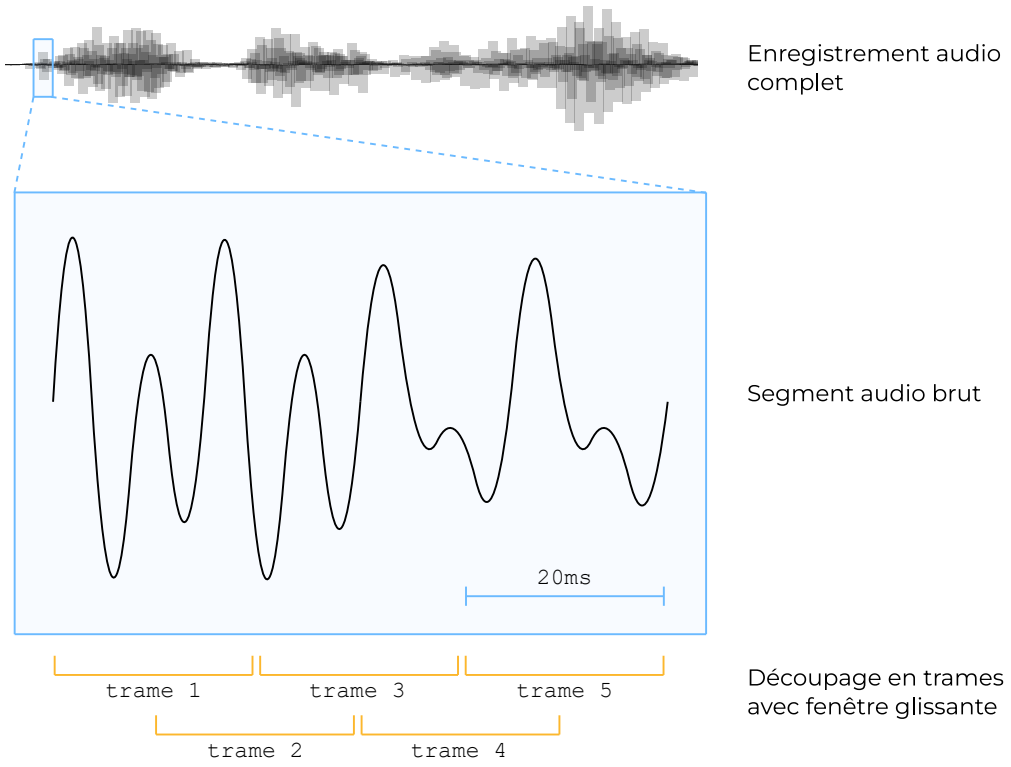

FIGURE 3.2 – Schématisation d'un découpage du signal audio en trames de parole avec fenêtre glissante.

3.1.1 Spectrogrammes

Un spectrogramme est généré par application de l'algorithme de *FFT* sur l'ensemble des fenêtres d'un signal de parole. Il permet de visualiser le signal de manière temporelle, sur l'axe des abscisses, et fréquentielle, sur l'axe des ordonnées. L'amplitude y est représentée en niveaux de gris, parfois sous forme de température, et indiquée en décibels. Elle permet de se focaliser sur les sons audibles par l'être humain, l'analyse visuelle d'un spectrogramme passant par la lecture de son amplitude. La Figure 3.3 donne un aperçu de cette représentation et met en avant ce principe.

Un spectrogramme Mel se différencie d'un spectrogramme classique par le passage de ses fréquences sous l'échelle de Mel. Une transformation en spectrogramme Mel est présentée en Figure 3.3.

L'échelle de Mel est utilisée pour améliorer la représentation fréquentielle du signal audio, du fait de notre perception non-linéraire des fréquences acoustiques. Davis et Mermelstein [1980] ont démontré notre sensibilité accrue aux plus basses fréquences, contrairement à une faible sensibilité aux plus hautes fréquences. Cette échelle fut initialement proposée par Stevens et al. [1937] afin d'approximer notre perception auditive et rendre plus linéraire la paramétrisation de la parole.

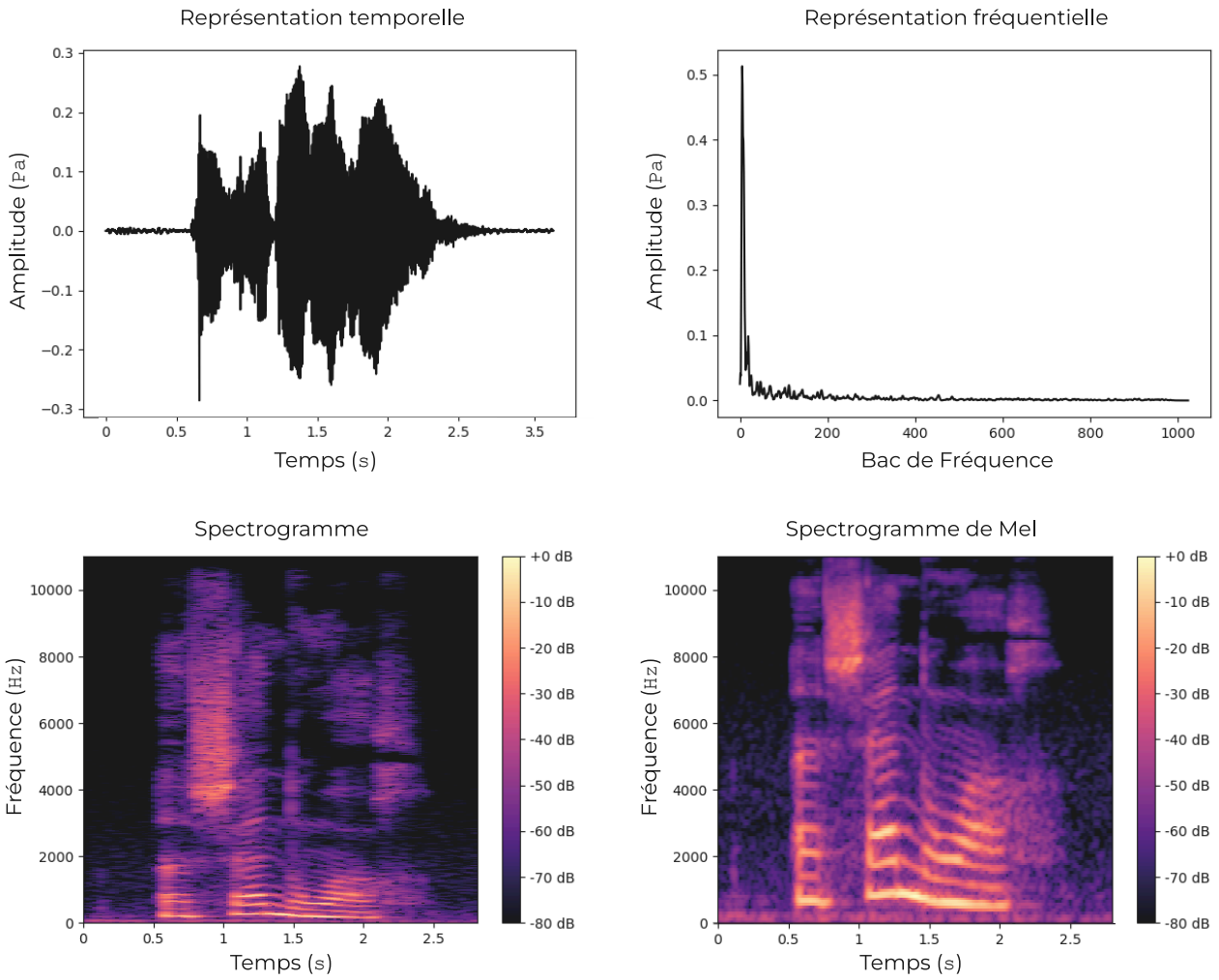

FIGURE 3.3 – Exemple de signal audio brut et sa transformation en spectrogramme puis spectrogramme de Mel, pour la vocalisation de «Un chat noir».

On la définit comme suit :

$$F_{mel}(f) = 2595 \times log_{10}\left(1 + \frac{f}{700}\right) \tag{3.1}$$

Avec f la fréquence en Hertz initialement utilisée dans un spectrogramme classique.

3.1.2 *MFCC*

Les *MFCC* (*Mel Frequency Cepstrum Coefficients*) furent introduits par Davis et Mermelstein [1980] afin de condenser l'information pertinente d'un signal de parole en un vecteur de coefficients. Ces coefficients peuvent servir d'entrée directe à un module de décodage de paramètres acoustiques.

La première étape d'une paramétrisation en *MFCC* consiste à appliquer l'algorithme de *FFT* sur les fenêtres de notre segment audio.

Le résultat de ces transformées de Fourier discrètes est ensuite filtré grâce à l'échelle de Mel. Les filtres présentés par Narayana et Kopparapu [2014] sont nommés banques de filtres Mel (*Mel filter-banks*), une banque de filtres servant à borner les fréquences traitées par une fréquence minimale et maximale. Une fréquence centrale f d'un filtre i pour une banque de filtres linéaire se calcule de la sorte :

$$f(i) = i \, \frac{f_{max} - f_{min}}{n+1} \tag{3.2}$$

Avec f_{max} et f_{min} respectivement les bornes de fréquence maximale et minimale et n le nombre de filtres définis. Afin d'obtenir des filtres à l'échelle de Mel en Hertz, on appliquera :

$$\begin{aligned} f^{mel}(i) &= F_{mel}^{-1}\left(i \, \frac{f_{max}^{mel} - f_{min}^{mel}}{n+1}\right) \\ F_{mel}^{-1}(f) &= 700(10^{\frac{f}{2595}} - 1) \end{aligned} \tag{3.3}$$

Avec comme bornes f_{max}^{mel} et f_{min}^{mel} :

$$\begin{aligned} f_{max}^{mel} &= F_{mel}(f_{max}) \\ f_{min}^{mel} &= F_{mel}(f_{min}) \end{aligned} \tag{3.4}$$

La Figure 3.4 donne un exemple de représentation de banques de filtres linéaire et à l'échelle de Mel, qui permettront de filter les fréquences des cepstres obtenus précédemment.

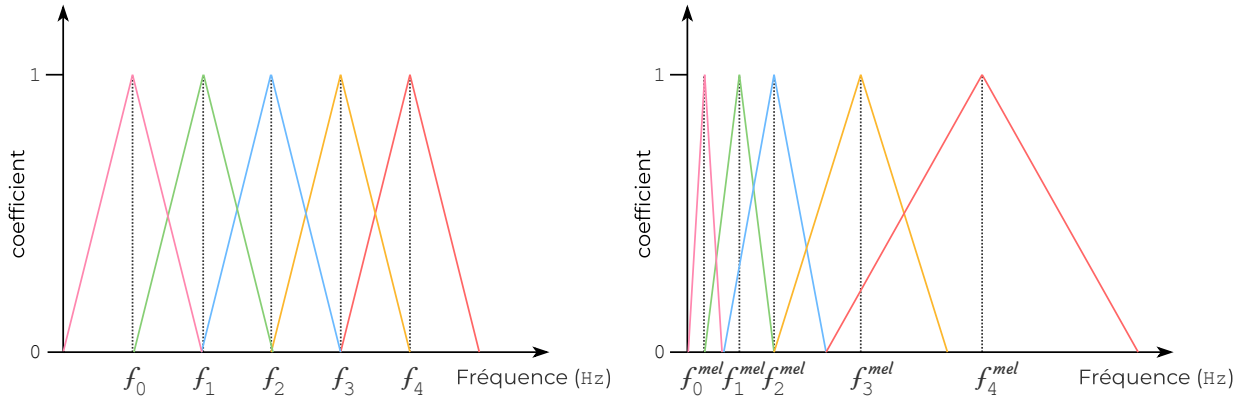

FIGURE 3.4 – Exemple de banques de filtres, linéaire à gauche et avec échelle de Mel à droite.

L'étape suivante consiste à prendre le logarithme des fréquences filtrées. La représentation vectorielle résultante nommée «vecteur de banque de filtre» (*fbank*) peut être utilisée directement en entrée de certains modèles neuronaux. Ces *fbanks* sont utilisés comme des *MFCC*, étant une paramétrisation du signal de parole proche.

Enfin, une seconde transformée de Fourier est réalisée. Cette fois on parlera de transformée de Fourier inverse, souvent notée DCT^{-1} pour *Discrete Cosine Transform*. Nous en obtiendrons une représentation compressée du signal de parole nommée cepstre, sur laquelle on pourra appliquer de nouveau un filtre afin de ne conserver que les coefficients cepstraux dont la valeur est généralement comprise entre 2 et 13. Le résultats de cette série d'étapes sera nommé *MFCC*.

Les *MFCC* ont longtemps été utilisés comme méthode la plus efficace pour paramétriser le signal de parole. Cependant, les *fbanks* ont vu leur popularité augmenter car ne nécessitant pas de transformée de Fourier inverse, qui a tendance à décorréler les coefficients entre eux. Les réseaux de neurones profonds ont pourtant refait basculer la balance avec entre autres l'étude de Abdel-Hamid et al. [2012] montrant l'importance minime d'une telle corrélation.

Avec l'avènement de l'apprentissage auto-supervisé, d'autres approches ont vu le jour, permettant un encodage direct du signal de parole à travers un apprentissage neuronale. La paramétrisation devient ainsi adaptable à la tâche, par optimisation des paramètres d'encodeurs de parole neuronaux très denses. Cette densité en fait son principal désavantage face aux solutions de paramétrisation non-neuronales, contre un gain de performances cependant conséquent.

3.2 Encodeurs de parole monolingues

Bien qu'utilisés depuis plus de vingt ans sous d'autres appellations, les premiers encodeurs de parole neuronaux à avoir réellement conquis le domaine de la Parole sont assez récents. En 2019, Schneider et al. présentaient wav2vec, un modèle semi-supervisé ayant pour but de traiter directement le signal acoustique brut afin d'en extraire des représentations vectorielles optimales pour diverses tâches de traitement de la parole. La semi-supervision signifie deux étapes d'apprentissage : une première étape d'auto-supervision sur un grand nombre de données non-annotées, telle que décrite au Chapitre 1, suivie d'une phase de *fine-tuning* du modèle sur une faible quantité de données annotées, quelques heures tout au plus.

Initialement, wav2vec utilisait uniquement deux réseaux convolutifs pour réaliser une prédiction du futur signal de parole. Avec l'arrivée des modèles Transformers, wav2vec fut amélioré et sa tâche modifiée, donnant lieu à wav2vec 2.0, base maintenant de nombreux encodeurs de parole monolingues, multilingues et sémantiques. Cette section décrira les encodeurs monolingues, tandis que nous aborderons ces deux autres catégories dans les sections suivantes.

3.2.1 wav2vec 2.0

Baevski et al. ont introduit wav2vec 2.0 [2020], évolution du wav2vec originel. Cet encodeur de parole performa dès son début sur des tâches de traitement de la parole, démontrant la faisabilité

et utilité d'un pré-apprentissage auto-supervisé dans ce domaine, bien que cette méthode d'apprentissage n'ait été populaire jusqu'alors que pour le Traitement Automatique du Langage Naturel écrit. C'est justement en se basant sur la tâche de masquage du modèle textuel BERT de Devlin et al. [2019] que ce nouveau wav2vec tira son épingle du jeu.

Comme son prédécesseur wav2vec, wav2vec 2.0 est composé d'un premier bloc d'encodage du signal convolutif, transformant celui-ci en représentations latentes. Il se base ensuite sur les modèles Transformers, notamment ses blocs d'encodage, et utilise un module similaire pour obtenir des représentations dites contextuelles. Enfin, il utilise un module de discrétisation des représentations latentes du signal pour les transformer en unités de parole discrètes, le tout afin d'optimiser une fonction de coût contrastive (*contrastive loss*). Ces quatre éléments sont mis en évidence par la Figure 3.5, illustrant le fonctionnement global de wav2vec 2.0, puis détaillés plus bas.

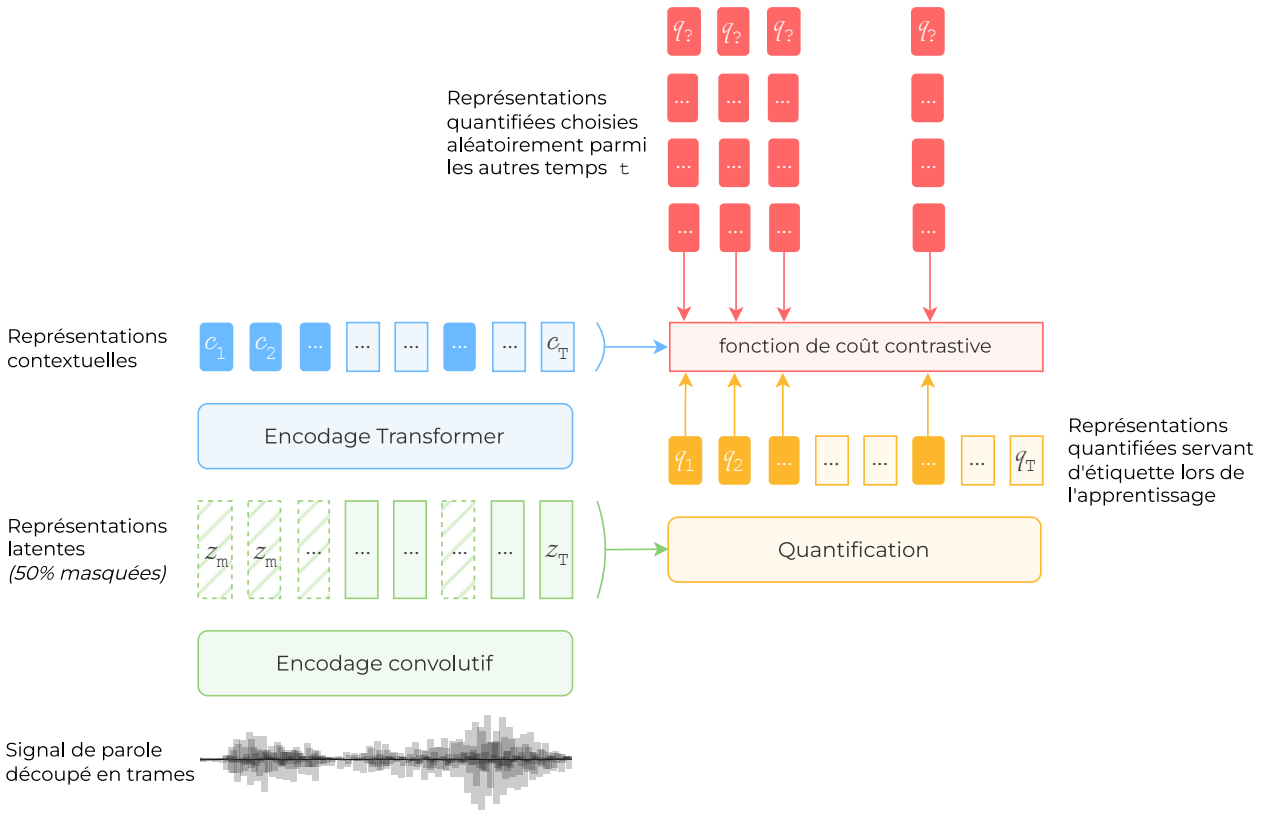

FIGURE 3.5 – Schématisation du fonctionnement de wav2vec 2.0.

Le bloc d'encodage convolutif décrit par la Figure 3.6 permet de réaliser une première compression des données audio brutes fournies en entrée du modèle. Une transformation des données

audio est nécessaire pour permettre leur apprentissage. Les entrées sont préalablement normalisées avant d'être transférées à sept blocs convolutifs, tous de 512 neurones, dont la taille du noyau et du pas de déplacement diminuent progressivement jusqu'à la couche supérieure. Les sorties de ce module, compressions des trames fenêtrées de 20 millisecondes, seront notées z_t pour un instant t.

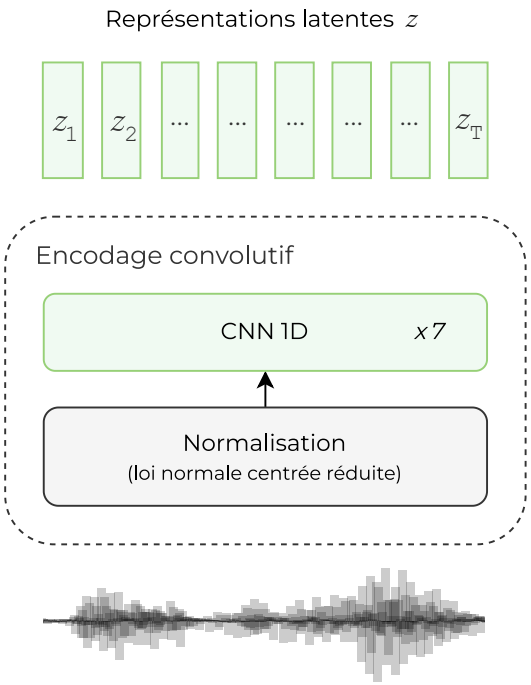

FIGURE 3.6 – Schématisation du fonctionnement du module d'encodage du signal en représentations vectorielles latentes dans wav2vec 2.0.

Le module suivant contextualise ces représentations latentes grâce à un encodage de type Transformer, comme l'illustre la Figure 3.7. Dans wav2vec 2.0 *base*, on comptera 12 blocs d'encodage contre 24 pour wav2vec 2.0 *large*. Afin d'utiliser les représentations latentes préalablement générées, il est nécessaire de premièrement ajuster leur dimension. On passe alors de représentations vectorielles de 512 à 768 éléments pour la version *base* et $1\ 024$ pour la version *large*. Contrairement à un encodage Transformer classique, une couche convolutive apprend à générer la représentation positionnelle relative de la trame t, qui sera additionnée à z'_t la projection linéaire préalablement réalisée. Finalement, le module de contextualisation génère les représentations contextuelles c.

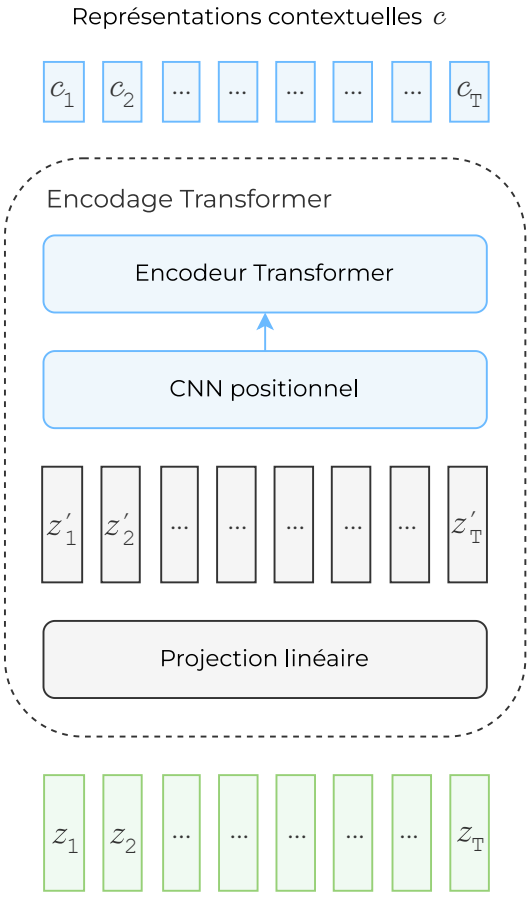

FIGURE 3.7 – Schématisation du fonctionnement du module de contextualisation de type encodeur de Transformer dans wav2vec 2.0.

Le module de discrétisation sert à palier la complexité lexicale liée au traitement de la parole lors de l'utilisation de modèles de type Transformer. Le signal de parole n'étant pas discret, cette étape consiste à transformer les représentations vectorielles latentes z en unités de parole discrètes qui serviront d'étiquettes afin de simuler un apprentissage supervisé. Ces étiquettes seront initialement composées de deux vecteurs (*codewords*) issus de deux dictionnaires différents (*codegroups* ou *codebooks*) ayant chacun un vocabulaire de 320 *codewords*. Afin de décider quel *codeword* de chaque *codebook* associer à un échantillon z_t, le module de discrétisation applique une Gumbel Softmax sur les deux *codebooks*. Les *codewords* sélectionnés pour z_t seront ensuite concaténés. Cette concaténation passera par une projection linéaire afin de générer les unités de parole discrétisées q. Ce sont elles qui serviront d'étiquettes lors de l'apprentissage de wav2vec 2.0. La Figure 3.8 illustre schématiquement le fonctionnement de ce module.

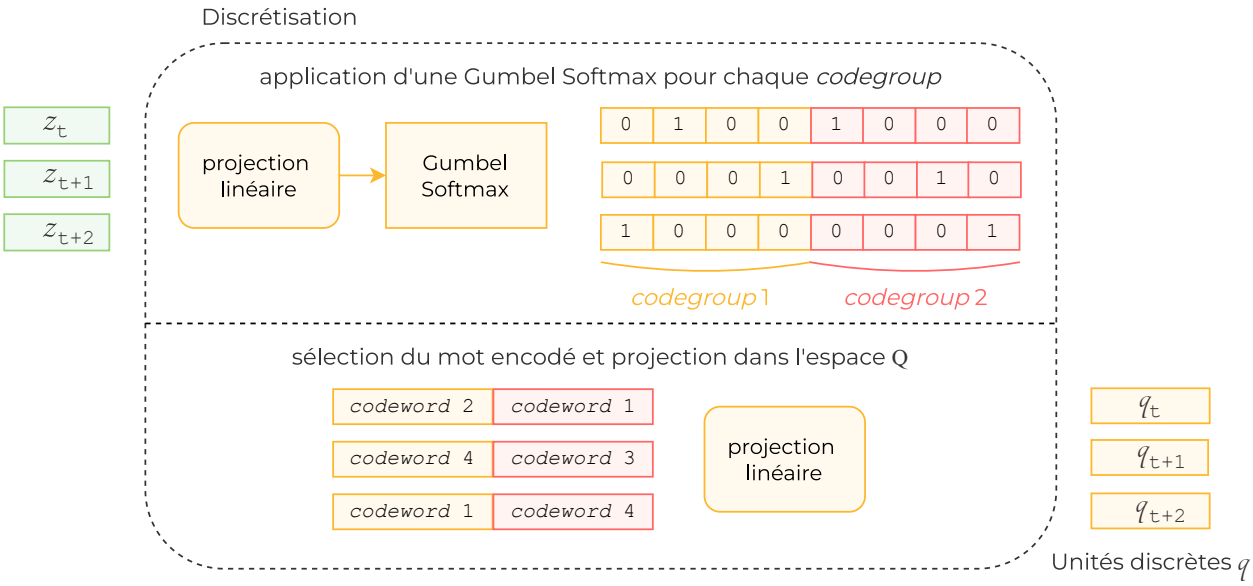

FIGURE 3.8 – Schématisation du fonctionnement de la discrétisation dans wav2vec 2.0, avec pour exemple des *codebooks* ayant chacun un vocabulaire de 4 *codewords*.

Durant l'apprentissage auto-supervisé de wav2vec 2.0, la tâche visée est la réduction d'une fonction de coût contrastive. La Figure 3.5 présentant l'ensemble des étapes d'apprentissage de wav2vec 2.0 met en avant le masquage réalisé pour l'optimisation de cette fonction de coût.

Environ la moitié des représentations latentes z sont aléatoirement masquées, remplacées par une représentation z_m commune. Ces représentations masquées sont fournies au module de contextualisation qui aura pour tâche de produire, à chaque instant t, un vecteur contextuel c_t identique à celui qu'il aurait dû produire si la représentation latente initiale n'avait pas été masquée. Ce vecteur c_t est projeté dans la même dimension que les discrétisations q. Pour chaque instant t, 100 vecteurs q sont générés depuis d'autres représentations latentes aléatoires. Dans la Figure 3.5, ces vecteurs sont notés $q_?$. Le modèle compare ainsi c_t à ces 100 représentations ainsi qu'à la vraie discrétisation q_t issue de z_t avant son masquage. wav2vec 2.0 tente donc de minimiser la distance entre c_t et q_t, avec l'aide d'une similarité cosinus et de la maximiser pour tout autre représentation $q_?$.

Une seconde fonction de coût est utilisée afin de favoriser la diversité des *codewords* utilisés, en maximisant l'entropie de la Gumbel Softmax utilisée dans le module de discrétisation.

Après ce pré-apprentissage conséquent, le modèle wav2vec 2.0 peut être *fine-tuné* sur une tâche précise avec un nombre limité de données étiquetées. Il suffira alors d'ajouter au dessus du module de contextualisation un nouveau module décodant les vecteurs c, communément appelés plongements de mots mais nommés représentations vectorielles dans cette thèse. Le module de discrétisation et les fonctions de coût associés au pré-apprentissage sont eux mis de côté.

3.2.2 LeBenchmark

Suite au succès de l'apprentissage auto-supervisé pour le traitement de la parole en anglais, Evain et al. [2021] proposèrent LeBenchmark, un modèle français basé sur wav2vec 2.0.

Celui-ci est pré-appris sur un ensemble de données français conséquent réunissant des enregistrements audios divers, dont de la parole naturelle (CFPP2000 [Branca-Rosoff et al. 2009], ESLO2 [Eshkol-Taravella et al. 2011], MPF [Gadet 2017], TCOF [André et Canut 2010], PortMEDIA [Lefèvre, Mostefa et al. 2012], EPAC [Estève, Bazillon et al. 2010]) avec différents accents, de la lecture (Multilingual LibriSpeech [Pratap, Q. Xu et al. 2020], African Accented French [1]), et du discours préparé émotionnellement (GEMEP [Bänziger et al. 2011], CaFE [Gournay et al. 2018], Att-HACK [Le Moine et Obin 2020]). Appris sur $1\,096$ heures de parole, l'encodeur de parole sera nommé LeBenchmark 1k, contre LeBenchmark 3k pour $2\,933$ heures de parole, et suivit de la dénomination *base* ou *large* suivant le nombre de modules d'encodage utilisés, comme c'est le cas pour wav2vec 2.0. Après publication, des modèles LeBenchmark 7k et 14k furent produits, suivant ce même principe.

C'est ce modèle auto-supervisé, sans les tâches de *fine-tuning* proposées par les auteurs, que nous utilisons pour certaines expérimentations monolingues françaises avec l'ensemble de données MEDIA [Bonneau-Maynard, Ayache et al. 2006] pour cette thèse.

Le modèle est fourni *fine-tuné* pour la réalisation de quatre tâches : Reconnaissance Automatique de la Parole, Compréhension Automatique de la Parole, Traduction Automatique de la Parole et Reconnaissance d'Émotion. Pour tous ces *fine-tunings*, un module décodera les représentations vectorielles issues d'un des modèles LeBenchmark pré-appris de manière auto-supervisée. Ont été testées des architectures de bout-en-bout plus ou moins profondes, mais aussi certaines utilisant un *HMM-DNN* hybride, dont les modules ont été préalablement décrits au Chapitre 2.

Pour la tâche de Reconnaissance Automatique de la Parole, les données françaises Common Voice [Ardila et al. 2020] et ETAPE [Gravier, Adda et al. 2012] ont été utilisées, totalisant environ 500 heures de données d'apprentissage transcrites dont seulement 20 heures pour ETAPE. Pour viser la compréhension de la parole, LeBenchmark a été *fine-tuné* sur un peu plus de 10 heures de données étiquetées en concepts sémantiques, provenant de la base de données MEDIA [Bonneau-Maynard, Ayache et al. 2006]. Ce sont les ensembles de données CoVoST-2 [C. Wang, A. Wu et al. 2021] et TEDx [Salesky et al. 2021], avec au total plus de 200 heures d'anglais, 38 heures d'espagnol et 25 heures de portugais, qui ont été utilisés pour *fine-tuner* le modèle sur la tâche de Traduction Automatique de la Parole. Enfin, RECOLA [Ringeval et al. 2013] et AlloSat [Macary et al. 2020], cumulant environ 40 heures de signal audio annoté, ont été utilisés pour la tâche de Reconnaissance d'Émotion.

1. https://www.openslr.org/57/

Quelques années après LeBenchmark, LeBenchmark 2.0 fut proposé par Parcollet et al. [2024], étant une version enrichie en modèles pré-appris et tâches de *fine-tuning* telles que l'Analyse Syntaxique et la Détection de Locuteur. C'est dans cette publication que les auteurs mettent en évidence les modèles 7k et 14k, mais aussi des modèles appris uniquement sur des enregistrements audios de voix connotées masculines ou féminines.

3.2.3 VoxPopuli

Les modèles VoxPopuli basés sur l'architecture *base* et *large* de wav2vec 2.0 ont été présentés par Wang et al. [2021] conjointement à l'ensemble de données VoxPopuli. Celui-ci réunit des données très majoritairement non-étiquetées dans de multiples langues, permettant le pré-apprentissage auto-supervisé de différents encodeurs de parole monolingues.

Parmi les 23 langues apprises, nous pouvons citer l'italien avec plus de $21\ 000$ heures d'audio non-étiquetées. C'est le modèle pré-appris découlant d'un apprentissage sur le sous-ensemble italien de VoxPopuli que nous utilisons dans cette thèse afin de traiter la tâche italienne de PortMEDIA [LEFÈVRE, MOSTEFA et al. 2012] lors d'apprentissages monolingues.

De la même manière que les modèles LeBenchmark, les modèles VoxPopuli pré-appris de manière auto-supervisée ont été par la suite *fine-tunés* sur des tâches spécifiques de traitement de la parole, comme la Reconnaissance Automatique de la Parole et la Traduction Automatique de la Parole, afin de démontrer leur efficacité.

3.3 Encodeurs de parole multilingues

Un modèle de type wav2vec peut être appris sur plusieurs langues à la fois, donnant ce que l'on appelle un encodeur de parole multilingue. Pour ce faire, un pré-apprentissage auto-supervisé classique peut être réalisé. Il est pour autant possible d'utiliser d'autres modules neuronaux simultanément lors de cet apprentissage afin d'améliorer l'optimisation du module d'encodage de la parole. Cette section présentera un encodeur de parole multilingue classique, XLS-R, ainsi qu'un encodeur de parole issu d'une nouvelle approche de supervision, Whisper.

3.3.1 XLS-R

C'est Conneau et al. qui présentèrent initialement le wav2vec 2.0 XLSR [2020], pour *Cross-Lingual Speech Representations*. Tout comme les encodeurs de parole LeBenchmark et VoxPopuli, ce modèle résulte d'un pré-apprentissage auto-supervisé sans ajout de module ou tâche externe. Le modèle XLSR-53 consiste en un apprentissage simultané sur 53 langues, issues des ensembles de données Common Voice [ARDILA et al. 2020], BABEL [GALES et al. 2014] et Multilingual LibriSpeech [PRATAP, Q. XU et al. 2020], pour un total de $56\ 000$ heures de parole. Les auteurs

ont démontré la pertinence d'un tel apprentissage, proposant un modèle plus performant que de précédents modèles monolingues pour les mêmes tâches, essentiellement pour le traitement de langues peu dotées. On notera aussi la pertinence d'un tel apprentissage pour des langues jamais traitées par le modèle. À contrario, l'apprentissage multilingue est démontré comme portant préjudice aux langues à ressources importantes, comme c'est le cas pour l'anglais, l'espagnol, le français et l'italien. Les auteurs parlent d'un transfert positif depuis les langues à grandes ressources vers les langues à faibles ressource, et de transfert négatif inversement. Il est induit qu'un modèle plus dense permettrait de pallier une partie de ce problème d'interférence.

La majeure différence avec un wav2vec 2.0 monolingue réside dans le partage de son vocabulaire de discrétisation. Les auteurs se sont basés sur l'apprentissage multilingue des modèles BERT de Devlin et al. [2019] et XLM de Lample et Conneau [2019]. Les mots encodés du module de discrétisation sont les mêmes pour toutes les langues. De ce fait, le modèle apprend à répartir ce vocabulaire parmi les langues, mais aussi à réutiliser de mêmes mots encodés pour plusieurs d'entre elles. Ce phénomène est représenté par un exemple en Figure 3.9. On peut alors parler de ponts entre les langues suite à la mise en commun d'une partie de leur espace de représentation d'unités de parole. De la même manière, les auteurs démontrent la création de regroupements (*clusters*) de langues proches pour les mots encodés, suite à ce partage de vocabulaire.

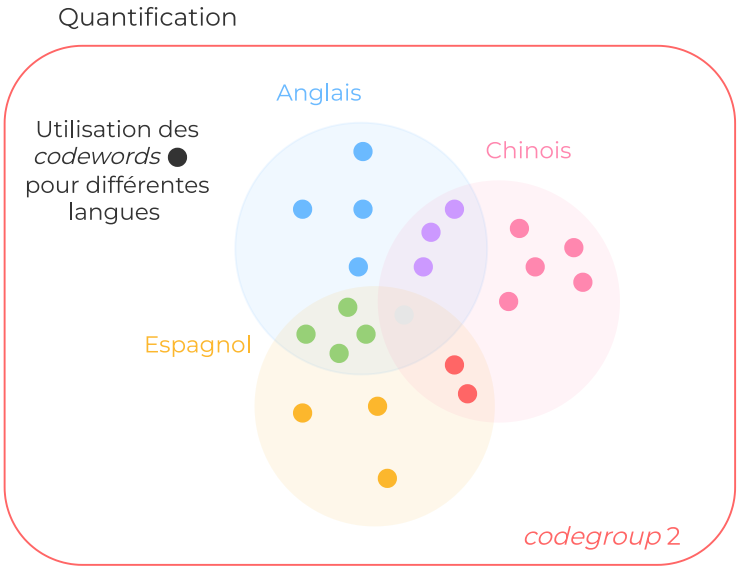

FIGURE 3.9 – Schématisation du partage de vocabulaire entre différentes langues dans le module de discrétisation de l'encodeur XLSR lors d'un pré-apprentissage auto-supervisé.

Conneau et al. démontrent ensuite l'intérêt de *fine-tuner* l'encodeur de parole de manière multilingue, en réalisant ce *fine-tuning* pour une tâche de Reconnaissance Automatique de la Parole.

En se basant sur l'encodeur de parole XLSR, Babu et al. proposèrent le modèle XLS-R [2022], appris sur 128 langues pour un peu moins de $500\ 000$ heures de parole. Ce modèle multilingue atteint alors les performances de l'état-de-l'art sur de nombreuses tâches comme c'est le cas en Traduction Automatique de la Parole, Reconnaissance Automatique de la Parole et Identification de la Langue. Les données utilisées lors du pré-apprentissage auto-supervisé sont issues des bases de données Common Voice [ARDILA et al. 2020], BABEL [GALES et al. 2014], VoxPopuli [C. WANG, RIVIERE et al. 2021], VoxLingua107 [VALK et ALUMÄE 2021] et Multilingual LibriSpeech [PRATAP, Q. XU et al. 2020].

C'est sur le modèle XLS-R que se base SAMU-XLSR, encodeur de parole sémantique présenté plus loin dans ce chapitre.

3.3.2 Whisper

L'inspiration pour Whisper naquit de Narayanan et al. [2018], Likhomanenko et al. [2020] et Chan et al. [2021] qui démontrèrent l'utilité d'un apprentissage supervisé multilingue et multi-domaine pour le traitement de la parole. Bien que la méthode de pré-apprentissage auto-supervisé ait été très bénéfique aux encodeurs de parole de type wav2vec, l'absence d'optimisation simultanée d'un module de décodage de qualité et la nécessité d'un *fine-tuning* secondaire du modèle pour une tâche précise limitent son champs d'utilisation et sa robustesse face à l'inconnu. L'encodeur et décodeur de parole Whisper fut proposé par Radford et al. [2023] afin de pallier ces problèmes. Les auteurs visent ainsi l'obtention d'un wav2vec 2.0 clefs en main sans nécessiter son *fine-tuning* pour chaque déploiement souhaité, le modèle ayant pour atout d'être pré-adapté à de nombreuses tâches, domaines et langues pour le Traitement Automatique de la Parole à travers ce que les auteurs ont nommé une faible-supervision (*weak-supervision*).

Sa multi-modalité se caractérise par sa capacité à tirer parti d'un apprentissage de Reconnaissance Automatique de la Parole pour réaliser d'autres tâches proches telles que la détection d'activité vocale ou identification de la langue. Bien que ces composants soient usuellement optimisés séparément avec différents *fine-tuning*, Whisper optimise l'ensemble de ses tâches lors de son unique apprentissage. Pour ce faire, le modèle réalise en parallèle l'entièreté de la pipeline de traitement de la parole pour ces sous-tâches, dont les résultats aideront les tâches principales de Reconnaissance et Traduction Automatique de la Parole.

Un potentiel inconvénient de cet apprentissage réside dans l'interférence pouvant être créée par le grand nombre de langues et tâches traitées en simultané, réduisant les performances sur les langues et tâches à importantes ressources, comme c'est le cas pour le modèle XLS-R. Afin de prouver l'absence de cette problématique avec Whisper, les auteurs ont comparé ses performances avec celles d'un apprentissage monolingue anglais. C'est sa densité qui permet à Whisper de se distinguer de plus petits modèles tels que XLS-R et de ne pas générer d'interférence de la

sorte, en venant même à surpasser le modèle appris uniquement sur l'anglais.

Afin de produire un encodeur et décodeur de parole multilingue, Whisper est appris sur un vaste ensemble de données de $680\ 000$ heures de parole transcrite. Parmi ces heures, $117\ 000$ couvrent 96 langues, avec $125\ 000$ heures dédiées à la tâche de traduction vers l'anglais. L'ensemble de données de Whisper compte parmi les plus grands ensembles étiquetés pour le traitement de la parole. En plus de la diversité des environnements d'enregistrement de la parole, mais aussi de celle des locuteurs et langues, les différents degrés de qualité du signal audio ont permis un apprentissage robuste, qualitatif et diversifié de Whisper. Cependant, de multiples traitements ont dû être réalisés afin de fournir une base de données propre pour son apprentissage, bien qu'issue directement d'internet. Ces traitements automatiques sont suivis d'une inspection manuelle.

Un premier filtrage consiste à identifier les transcriptions issues de modèles de Traitement Automatique de la Parole et de les retirer de la base de données. Ce filtrage automatique sert à améliorer la qualité moyenne des transcriptions apprises par Whisper. Ce choix a été fait suite à l'étude de Ghorbani et al. [2022], démontrant l'inefficacité d'un apprentissage sur un mélange de transcriptions manuelles et automatiques.

Un second filtrage est réalisé pour vérifier l'alignement linguistique entre signal de parole et transcription. La seule tâche de traduction traitée par Whisper étant une traduction vers l'anglais, les données correspondantes sont les seules traductions gardées pour l'apprentissage du modèle.

Les échantillons sans activité vocale sont eux préservés dans la base de données afin de réaliser une sous-tâche de détection d'activité vocale.

L'encodeur de Whisper prend en entrée le logarithme de la magnitude des spectrogrammes de Mel. Cette entrée est réduite à l'intervalle $[-1, 1]$.

Comme pour un Transformer classique, la représentation positionnelle de la séquence en entrée du bloc de décodage est optimisée durant l'apprentissage. Le décodeur prend en entrée la séquence de sortie précédemment prédite, enrichie de nombreux autres tokens, issus des sous-tâches de Whisper. Ces tokens sont générés simultanément avant d'être concaténés entre eux. La Figure 3.10 présente un exemple de séquence enrichie. Un token indiquant le commencement (*BOS*) et la fin (*EOS*) du segment audio est ajouté à la séquence. Les temps de début et de fin des trames traitées sont indiqués eux aussi si demandés par un token booléen (*timestamp*). Ces tokens de segmentation permettent de traiter plus efficacement les dépendances à long terme de la parole. Afin d'éviter une mauvaise segmentation du début de la séquence, le premier token de segmentation audio est impérativement compris entre 0 et 1 seconde. Un autre token indique la prédiction de la langue du signal audio fourni en entrée du modèle. Les tokens uniques d'identification de la langue sont au nombre de 99. Si aucune parole n'est présente dans le signal audio, alors ce token indiquera l'absence de signal vocal. Un dernier token indiquera la tâche souhaitée, étant soit transcription soit traduction textuelle vers l'anglais.

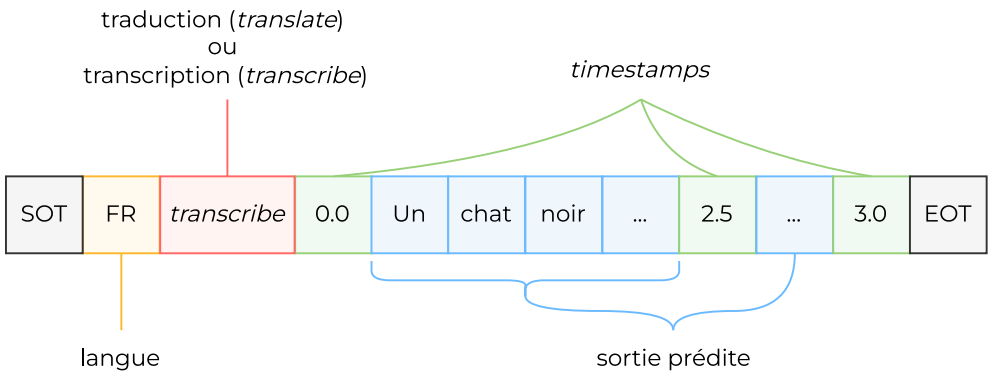

FIGURE 3.10 – Exemple de séquence fournie au module de décodage de Whisper.

Une fois la séquence formée, le décodage de celle-ci commence afin de réaliser la tâche principale souhaitée. La prédiction se fait en texte brut sans standardisation ou normalisation. Une tokenisation des sorties est cependant réalisée au format BPE [SENNRICH et al. 2016, RADFORD, J. WU et al. 2019] pour la tâche de Reconnaissance Automatique de la Parole. Un algorithme de recherche par faisceau est utilisé afin de choisir les BPE les plus probables.

3.4 Encodeurs de parole sémantiques

On définit ici un encodeur de parole sémantique comme un encodeur multilingue appris de sorte à générer une représentation compressée de l'information sémantique présente dans le signal de parole. Bien que ces encodeurs de parole ne soient de prime abord pas pensés pour une tâche de Compréhension Automatique de la Parole, leur agnosticisme à la langue et leur capacité de condensation sémantique les pré-disposent à ce domaine d'application. Nous présentons dans cette section les modèles SONAR et SAMU-XLSR, nous étant tout particulièrement intéressés sur ce dernier durant de cette thèse.

3.4.1 SAMU-XLSR

SAMU-XLSR (*Semantically-Aligned Multimodal Utterance-level Cross-Lingual Speech Representation*) fut proposé par Khurana et al. [2022] pour une tâche de recherche d'information vocale puis de traduction de la parole. Cet encodeur de parole se base sur un *fine-tuning* du XLS-R de Babu et al. [2022] avec l'aide des représentations sémantiques issues de l'encodeur textuel LaBSE (*Language-agnostic BERT Sentence Embedding*) de Feng et al. [2022], le rendant multilingue. Cette approche vise à améliorer la capture de la sémantique contenue dans un signal audio, comme LaBSE le fait sur des données textuelles.

LaBSE est un modèle d'extraction sémantique agnostique à la langue, appris sur plus de 109

langues. Pour réaliser son apprentissage, les auteurs ont utilisé les données de CommonCrawl [SCHWENK, WENZEK et al. 2021] et Wikipedia[2], soit un total de 6 milliards de données textuelles appairées pour une tâche de traduction. Ces dernières, issues directement d'internet, ont été triées manuellement. Les annotateurs ont évalué leur qualité puis un seuil a été utilisé afin de supprimer les traductions jugées trop mauvaises. Pour chaque langues, les 100 millions meilleures paires ont été conservées. Faisant suite aux travaux de Yang et al. [2021], cet ensemble de données sera utilisé pour le pré-apprentissage de l'encodeur utilisé par LaBSE mais aussi pour son dual *fine-tuning*. La Figure 3.11 illustre l'architecture du modèle.

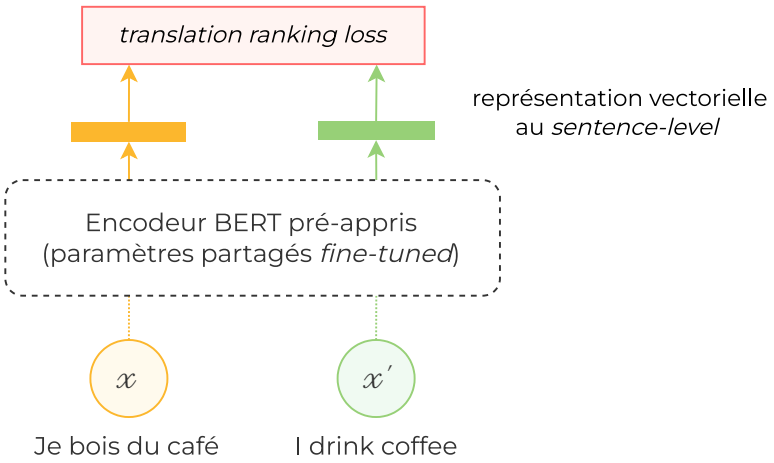

FIGURE 3.11 – Schématisation de l'architecture de LaBSE.

LaBSE se base sur l'architecture Transformer de Vaswani et al. [2017], réutilisant le modèle BERT *base* de Devlin et al. [2019] avec 12 têtes d'encodage. Ce bloc d'encodage, conçut pour les 109 langues de LaBSE, va subir un premier apprentissage pour une tâche de masquage de la langue (*MLM*) identique à celle du modèle BERT, combinée à une tâche de traduction de la langue (*TLM*) comme introduite par Lample et Conneau [2019]. L'encodeur textuel est ensuite utilisé pour générer une représentation vectorielle d'une phrase dans une langue l_1 mais aussi de sa traduction dans une langue l_2. Les auteurs parlent d'un apprentissage dual de l'encodeur. Le modèle optimise ainsi une fonction de coût maximisant la similarité des deux représentations vectorielles générées par cet encodeur commun (*translation ranking loss*).

Une des particularité de LaBSE réside en sa faculté à traiter efficacement des langues jamais vues durant l'apprentissage. Ce phénomène est dû à la grande couverture linguistique de son corpus d'apprentissage, lui donnant la capacité de généraliser ses connaissances entre langues proches. Cette grande couverture rend LaBSE agnostique à la langue. Le modèle fut donc choisi pour l'apprentissage de SAMU-XLSR afin de concentrer le *fine-tuning* de leur modèle wav2vec

2. https://fr.wikipedia.org/

sur l'extraction directe de la sémantique contenue dans la parole, contrairement aux wav2vec présentés ultérieurement qui se focalisant généralement sur des tâches de transcription de la parole.

L'approche SAMU vise à rapprocher les représentations vectorielles de l'encodeur de parole en cours de *fine-tuning* à celles de LaBSE dans un même espace, le tout grâce à une fonction de coût de similarité cosinus. On peut représenter cette tâche comme l'optimisation des paramètres d'un encodeur de parole pour viser la prédiction d'une représentation sémantique agnostique à la langue. La Figure 3.12 schématique l'architecture de SAMU-XLSR et illustre l'utilisation de sa fonction de coût.

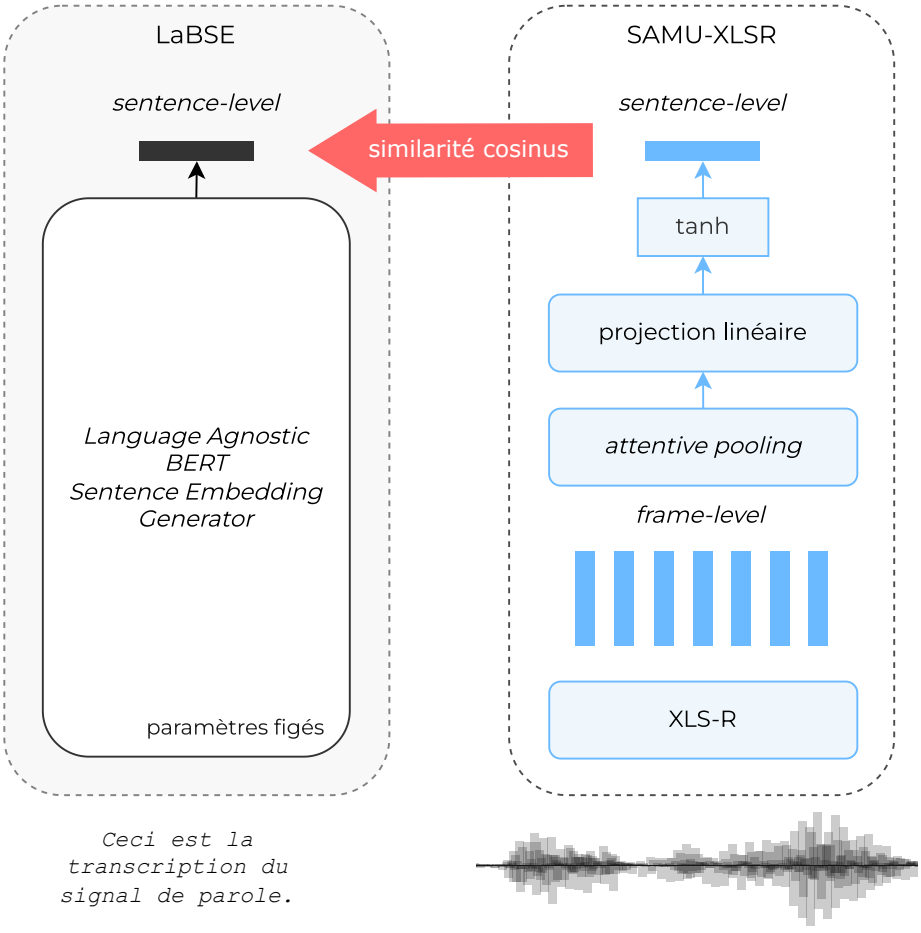

FIGURE 3.12 – Schématisation de l'architecture de SAMU-XLSR.

Les représentations vectorielles de LaBSE sont générées au niveau d'une phrase, cette phrase étant alignée au segment complet du signal de parole fourni en entrée du XLS-R. En revanche, le modèle XLS-R génère une représentation vectorielle pour chaque trame acoustique de ce segment. Un module supplémentaire apprend donc à assembler les représentations vectorielles des

trames issues du modèle XLS-R (*frame-level*) en une unique représentation vectorielle de l'ensemble du segment de parole (*sentence-level*). Ce module est composé d'une fonction de sous-échantillonage avec mécanismes d'attention (*attentive pooling*) qui permet d'obtenir une première représentation *sentence-level* en limitant la perte d'information due à cette compression. S'ensuit une projection linéaire permettant l'ajustement de cette représentation pour l'approcher au plus de celle de LaBSE, puis une fonction de tangente hyperbolique est appliquée. En parallèle, les représentations textuelles *sentence-level* de LaBSE sont simplement extraites du modèle dont les paramètres restent figés.

SAMU-XLSR est appris sur des enregistrements audios et leur transcription, le signal de parole étant transmis à XLS-R tandis que sa transcription est fournie à LaBSE. LaBSE étant aligné sémantiquement à travers différentes langues, il est donc possible de lui fournir toute traduction textuelle du signal de parole, bien qu'en pratique cela ne permette pas une meilleure optimisation de l'encodeur de parole. En effet, être agnostique à la langue lui permet de ne pas faire la différence entre des phrases sémantiquement identiques bien qu'étant écrites dans des langues différentes. La Figure 3.13 illustre ce principe par un exemple schématisé.

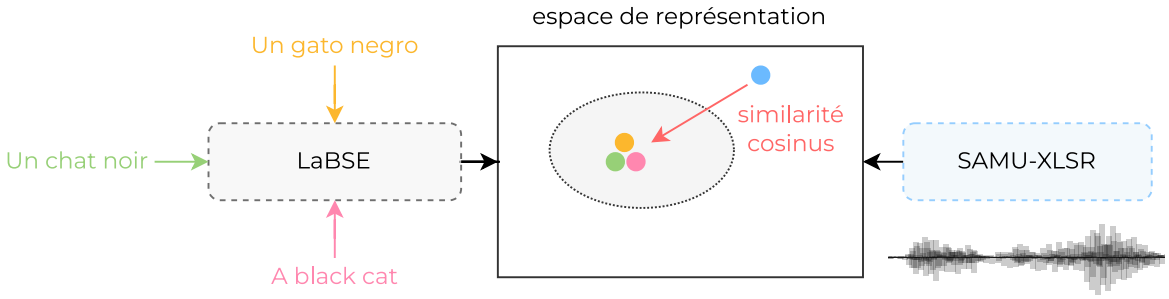

FIGURE 3.13 – Exemple d'apprentissage, intrinsèquement agnostique à la langue, de SAMU-XLSR pour un audio anglais et une transcription anglaise, dont la représentation issue de LaBSE équivaut à celle d'une traduction française et espagnole.

L'apprentissage présenté dans l'article [KHURANA et al. 2022] se déroule sur des données multilingues issues de la septième version de l'ensemble de données Common Voice [ARDILA et al. 2020], totalisant $6\ 800$ heures de parole dans 25 langues. Cet ensemble étant fortement déséquilibré, les auteurs ont réalisé une augmentation de données pour les langues peu représentées, ainsi qu'une réduction de données pour celles trop représentées. L'augmentation de données se fait par duplication d'échantillons, tandis que la réduction de données est réalisée en supprimant des échantillons aléatoirement. Nous utilisons dans cette thèse un modèle SAMU-XLSR appris sur 53 langues.

3.4.2 SONAR

L'encodeur de parole SONAR (*Sentence-level multimOdal and laNguage Agnostic Representations*) de Duquenne et al. [2023] est lui aussi multi-modal et multilingue, bien que différent de ceux présentés précédemment. Ce modèle est optimisé sur des alignements de signaux de parole et transcriptions, ces dernières étant fournies à un encodeur-décodeur multilingue textuel qui sera *fine-tuné* lors de l'apprentissage de SONAR.

Peu de temps avant l'apparition de SONAR et de SAMU-XLSR de Khurana et al. [2022], Duquenne et al. proposaient un encodeur de parole [2021] semblable à SAMU-XLSR. Sa différence résidait en leur choix d'encodeurs ainsi que leur méthode de compression des représentations de la parole au *sentence-level*. La fonction de coût minimisée était ici aussi une similarité cosinus et la tâche visée une Traduction Automatique de la Parole pour 4 langues issues de Librivox et CommonCrawl [2021], pour plus de $20\ 000$ heures de parole.

LASER de Artetxe et Schwenk [2019] fut choisi comme encodeur textuel car prouvé pertinent dans le domaine de la fouille de texte (*text mining*) par Schwenk et al. [2021] et ayant la particularité d'être agnostique à la langue comme LaBSE présenté plus haut, étant appris sur plus de 80 d'entre elles. Cet encodeur textuel se base sur l'architecture encodeur-décodeur avec couches récurrentes de Cho et al. [2014]. L'encodeur de parole multilingue XLSR de Conneau et al. [2020] était utilisé, seul son module d'encodage de type Transformer étant *fine-tuné*. Afin d'obtenir une représentation de la parole au *sentence-level* équivalente à celle de LASER, les auteurs choisirent d'utiliser pour chaque segment de parole la première sortie de son encodeur, nommée BOS (*begin-of-sentence*). Ce choix s'est appuyé sur les études de Devlin et al. [2019] et Reimers et Gurevych [2019] qui démontrèrent la pertinence d'utiliser de tels tokens de début de phrase dans le domaine textuel.

Concernant SONAR, ce n'est ni LaBSE de Feng et al. [2022] ni LASER3 de Heffernan et al. [2022] qui furent choisis pour encoder les échantillons textuels, mais NLLB 1B de la NLLB Team [2022]. Cet encodeur textuel basé sur l'architecture Transformer de Vaswani et al. [2017] et pré-appris sur 202 langues fut prouvé par les auteurs comme étant plus performant sur certaines tâches. Comme LaBSE, ce modèle est optimisé pour une tâche de traduction textuelle, et est donc surmonté d'un décodeur. La principale modification apportée au modèle NLLB pour SONAR est la présence d'une projection linéaire réalisée entre les deux blocs, visant à réduire la dimension de la sortie de l'encodeur via un goulet d'étranglement (*bottleneck*), afin d'obtenir une représentation vectorielle au *sentence-level*. Aucune attention (*cross-attention*) n'est réalisée entre le module d'encodage et celui de décodage.

C'est l'encodeur de parole W2V-BERT de Chung et al. [2021] qui est utilisé dans SONAR. Ce modèle, dont l'architecture est illustré par la Figure 3.14, tire sa particularité du mélange du wav2vec 2.0 de Babu et al. [2022] et du BERT de Devlin et al. [2019], utilisant des blocs Conformers

introduits par Gulati et al. [2020], combinaison de Transformers et convolutions. Le W2V-BERT est décrit par ses auteurs comme proche des modèles HuBERT [HSU et al. 2021], vq-wav2vec [BAEVSKI, SCHNEIDER et al. 2020] et DiscreteBERT [BAEVSKI, AULI et al. 2019], bien que n'étant appris que sur une unique tâche de bout-en-bout et corrigeant le problème de discrétisation du signal de HuBERT. Cette correction passe par un apprentissage pour deux fonctions de coût : une fonction contrastive de similarité cosinus comme réalisée par wav2vec 2.0, et une tâche de masquage identique à celle de BERT (*MLM*). La première prendra en entrée les unités de parole q issues du module de discrétisation, tandis que les identifiants i du vocabulaire de discrétisation des éléments masqués seront fournis à la seconde. Son apprentissage est réalisé sur environ $60\ 000$ heures de signal audio non-étiquetées, issues de l'ensemble de données Libri-Light [KAHN et al. 2020].

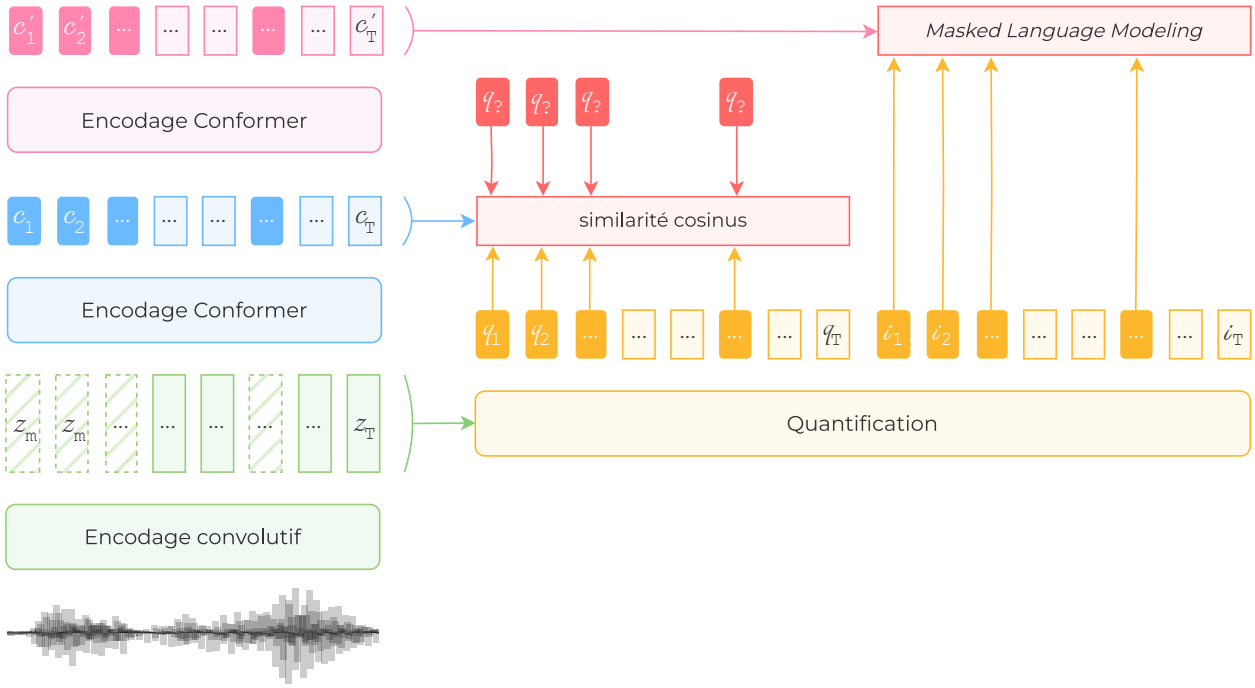

FIGURE 3.14 – Schématisation de l'architecture de W2V-BERT.

La première étape de l'apprentissage de SONAR consiste à *fine-tuner* le décodeur du modèle multilingue NLLB sur une tâche de Traduction Automatique comprenant 200 langues. Les auteurs appellent cette étape un *random interpolation decoding*. Cette tâche est combinée à des objectifs de réduction de bruits par auto-encodage (*denoising auto-encoding*) [Y. LIU, GU et al. 2020] et de similarité cross-lingue (*cross-lingual similarity*) pour la représentation vectorielle au *sentence-level*.

La seconde étape de l'apprentissage de SONAR, présentée par la Figure 3.15, est elle-aussi réalisée via la minimisation d'une combinaison de fonctions de coût pour un *fine-tuning* de l'en-

semble du modèle sur 37 langues. On compte parmi elles une Erreur Quadratique Moyenne (*MSE*) réalisée entre les représentations textuelles de NLLB et les représentations de parole du W2V-BERT, ainsi que toutes celles de l'étape précédente. La fonction de coût (*MSE*) permet d'améliorer l'alignement des représentations au *sentence-level* entre les différentes langues, menant à un meilleur agnosticisme de la langue pour SONAR.

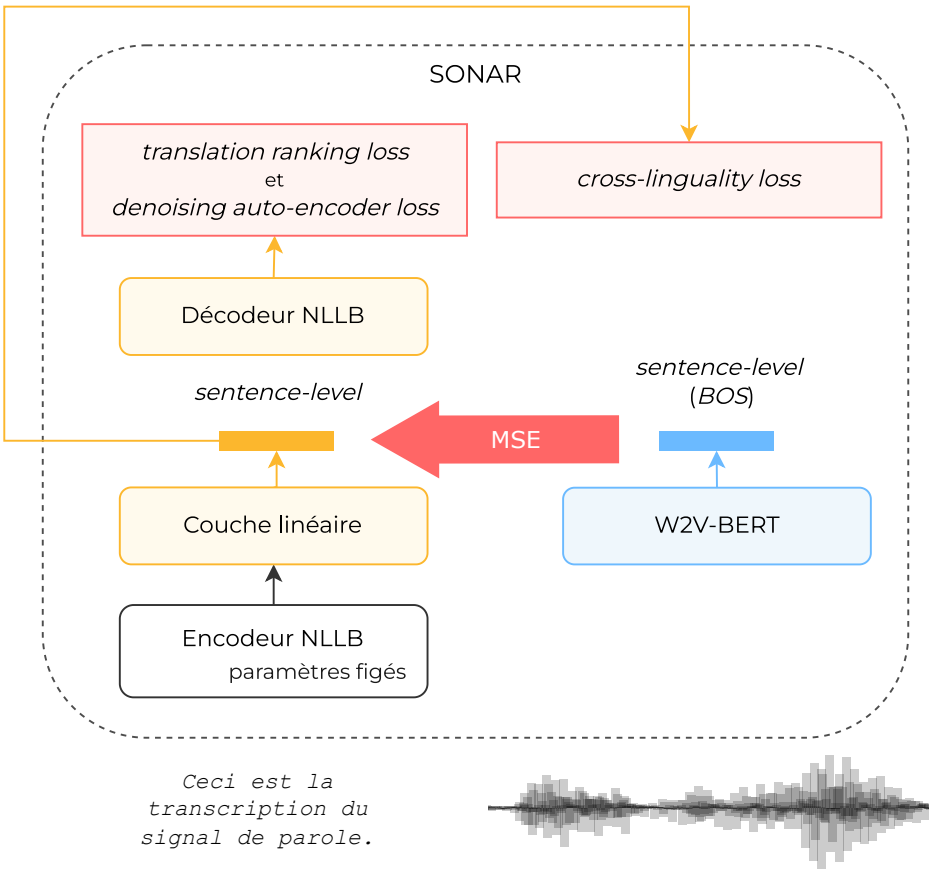

FIGURE 3.15 – Schématisation de l'architecture de SONAR.

3.5 Conclusion

Ce chapitre présente les méthodes d'encodage de la parole par paramétrisation acoustique ainsi que par approche neuronale, utilisées entre-autres dans le domaine de la Compréhension Automatique de la Parole. Bien que cette thèse n'utilise principalement que des méthodes neuronales d'encodage de la parole pour un gain significatif de performances, nous tenons à attirer l'attention vers leur coût computationnel important.

Après avoir décrit le fonctionnement du modèle wav2vec 2.0, base de nombreux encodeurs

de parole, ce chapitre a décrit le fonctionnement et les phases d'apprentissage de plusieurs encodeurs de parole monolingues, multilingues et sémantiques. Ceux-ci sont à l'état-de-l'art dans le domaine de la Compréhension Automatique de la Parole, bien que parfois initialement proposés pour des tâches plus variées telles que la Traduction Automatique de la Parole ou la Reconnaissance Automatique de la Parole.

Cette thèse se focalisera sur l'utilisation et l'analyse de plusieurs de ces encodeurs de parole neuronaux.

Le modèle LeBenchmark sera plus particulièrement utilisé pour nos apprentissages monolingues français au Chapitre 4.

Au Chapitre 5, SAMU-XLSR sera principalement utilisé pour nos expérimentations multilingues et cross-lingues pour le français, l'italien et le tunisien, étant comparé à XLS-R pour démontrer l'impact de son enrichissement sémantique.

Ces chapitres fourniront une analyse poussée des encodeurs de parole les plus pertinents pour cette thèse, tant dans leur encodage linguistique que sémantique.

Deuxième partie

Contributions

Chapitre 4

Données, métriques d'évaluation et recette

Sommaire

Publications liées à ce chapitre

- **LREC 2022** – The Spoken Language Understanding MEDIA Benchmark Dataset in the era of Deep Learning : Data updates, training and evaluation tools [LAPERRIÈRE, PELLOIN, CAUBRIERE et al. 2022b]
- **JEP 2022** – Le benchmark MEDIA revisité : données, outils et évaluation dans un contexte d'apprentissage profond [LAPERRIÈRE, PELLOIN, CAUBRIERE et al. 2022a]

Ce chapitre présente les ensembles de données principalement utilisés lors de cette thèse pour une tâche de transcription avec extraction sémantique depuis la parole dans le cadre de dialogues humain-machine. Après avoir décrit les ensembles de données MEDIA [BONNEAU-MAYNARD, ROSSET et al. 2005] et PortMEDIA [LEFÈVRE, MOSTEFA et al. 2012] et leur formalisme d'annotation et discuté de leurs métriques d'évaluation, nous parlerons de nos contributions concernant la mise en avant de MEDIA dans la recherche scientifique actuelle à travers une recette réalisée sur la boîte à outils SpeechBrain [RAVANELLI, PARCOLLET et al. 2021].

D'autres données telles que l'ensemble tunisien TARIC-SLU [MDHAFFAR, BOUGARES et al. 2024] ont été utilisées lors de nos expérimentations de portabilité cross-lingue. Nous les décrirons plus brièvement lorsque nous parlerons de celles-ci dans les chapitres à venir.

Comme déjà abordé dans le Chapitre 1, il est d'usage de découper l'ensemble de données d'apprentissage en trois corpora distincts :

— **Apprentissage** (*train*) : Il représente généralement 70 à 80% de l'ensemble des données. Servant d'exemple lors de la phase d'apprentissage, il permet au modèle de modifier ses paramètres dans le but de pouvoir prédire des annotations pour un corpus similaire.

— **Développement** (*dev*) : Il représente généralement 10 à 15% de l'ensemble des données. Il est utilisé en inférence à la fin d'une époque d'apprentissage afin d'évaluer les performances du modèle et d'en choisir la meilleure version. Cette évaluation permettra d'ajuster ses hyper-paramètres et de poursuivre ou arrêter l'apprentissage du modèle si celui-ci sur-apprend le corpus de *train* et perd donc de sa capacité à généraliser ses connaissances sur d'autres données. Ce corpus n'influence pas directement la modification des paramètres du modèle.

— **Test** (*test*) : Il représente généralement 10 à 15% de l'ensemble des données. Il est utilisé en inférence après l'apprentissage afin d'évaluer les performances finales du modèle. Ce corpus n'est jamais utilisé lors de l'apprentissage, ni pour modifier les paramètres ni pour ajuster les hyper-paramètres ou faire des choix humains concernant le système. Les résultats de son évaluation reflètent donc l'exploitation possible du modèle en conditions réelles, avec notamment sa capacité de généralisation face à l'inconnu.

La distribution d'heures de ces corpora sera donnée pour chaque ensemble de données présenté dans ce chapitre.

4.1 MEDIA et PortMEDIA

MEDIA [BONNEAU-MAYNARD, ROSSET et al. 2005] (Méthodologie d'Évaluation automatique de la compréhension hors et en contexte du DIAlogue) et PortMEDIA [LEFÈVRE, MOSTEFA et al. 2012 ; JABAIAN 2012] sont des ensembles de données dédiés à la compréhension de la parole.

L'ensemble de données françaises MEDIA fut créé dans le cadre du projet Technolangue MEDIA-EVALDA établi par le gouvernement français de 2002 à 2006 et est distribué gratuitement pour la recherche académique depuis 2020 par ELRA (*European Language Resources Association*) [1].

PortMEDIA fut développé en 2009 lors du projet éponyme PortMEDIA lui-aussi issu de fonds nationaux français dans le cadre de l'appel à projet Contenu et Interactions, mais aussi de fonds européens avec le projet LUNA [SERVAN et al. 2010]. Il est distribué de la même manière que MEDIA par ELRA depuis 2020 [2]. Il s'agit en réalité de deux ensembles de données visant une portabilité multilingue et multi-domaine de MEDIA et l'amélioration de sa robustesse face aux erreurs de reconnaissance de la parole. Ces deux ensembles de données complètent MEDIA. PM-DOM, nommé ici PortMEDIA-fr, propose un ajout de dialogues français dans un domaine sémantique proche de celui de MEDIA. PM-LANG, nommé ici PortMEDIA-it, propose une version italienne d'un sous-ensemble de MEDIA.

Ces ensembles de données visent à mettre en place une infrastructure de production et de diffusion de ressources linguistiques et d'évaluation des technologies de la langue écrite et orale. Ils sont pour cela annotés pour réaliser l'extraction de l'information sémantique d'enregistrements téléphoniques dans un contexte de dialogues humain-machine scénarisés pour une tâche de réservation de chambre d'hôtel ou de spectacle. Les utilisateurs communiquent au serveur téléphonique leurs préférences en terme de situation géographique, périodes, prix et équipements souhaités pour ces réservations tout en demandant des renseignements touristiques. Les scénarios établis par avance consistent en des listes d'informations à demander au serveur, la formulation restant dépendante de l'utilisateur lui-même. Ceux-ci sont plus ou moins complexes, partant de réservations simples et allant jusqu'à des réservations très précises et multiples avec un grand nombre de requêtes, voire des hésitations et modifications au cours du même dialogue.

La machine interlocutrice de l'utilisateur humain dans les conversations téléphoniques de MEDIA et PortMEDIA est un magicien d'Oz (*Wizard of Oz, WoZ*) [KELLEY 1984]. Il s'agit d'une simulation d'un dialogue humain-machine, le magicien d'Oz étant en réalité un humain se faisant passer pour une machine. Celui-ci dispose de scénarios définis par avance afin d'uniformiser au mieux son discours mais aussi d'une réelle plate-forme d'information touristique où il fournira les requêtes de l'utilisateur afin de lui répondre en conséquence. Ses réponses peuvent être claires ou

1. http://catalog.elra.info/en-us/repository/browse/ELRA-E0024/
2. http://www.elra.info/en/projects/archived-projects/port-media/

non suivant son choix, et peuvent simuler des erreurs de reconnaissance vocale. Sa tâche consistera à choisir les questions ou réponses adaptées à la requête de l'utilisateur. L'intérêt principal de cette méthode réside dans le fait qu'il n'est ainsi pas nécessaire d'avoir un système de dialogue automatique construit spécifiquement pour la tâche pour les tours de la machine.

4.1.1 Formalisme d'annotation

MEDIA et PortMEDIA utilisent un formalisme d'annotation complexe, avec une représentation sémantique de haut niveau. L'étude de Béchet et Raymond [BÉCHET et RAYMOND 2019] a mis en évidence le fait que MEDIA soit actuellement considéré comme l'un des ensembles de données pour la tâche de Compréhension Automatique de la Parole le plus riche et complexe à traiter, par rapport à d'autres ensembles bien connus tels que ATIS [DAHL et al. 2012] et M2M [SHAH et al. 2018]. Un exemple de phrase annotée est donné au Chapitre 2 en Figure 2.1. Les différentes spécificités de l'annotation de MEDIA sont détaillées dans cette section.

La collecte et l'annotation des différents corpora ont été prises en charge par ELRA. Seuls les tours de l'utilisateur ont été annotés de manière manuelle, avec une pré-annotation automatique pour PortMEDIA. Afin de réaliser cette annotation, tous les enregistrements audio y compris ceux du compère magicien d'Oz ont d'abord été transcrits manuellement pour MEDIA et automatiquement pour PortMEDIA, bien qu'une correction manuelle ait été apportée par la suite. Les annotateurs disposaient ensuite d'un outil d'annotation sémantique nommé Semantizer. Comme le compère utilise une plate-forme recensant une liste d'hôtels avec leurs équipements et localisation, la plupart des groupes de mots à annoter sémantiquement dans MEDIA étaient connus de l'annotateur. Les dates, nombres, et autres données plus ou moins numériques étaient aussi aisément annotables. D'autres mots spécifiques à la tâche d'extraction sémantique étaient rassemblés dans un lexique sémantique afin de faciliter leur association avec un concept.

L'ontologie de ces ensembles de données est complexe, les étiquettes étant triées en divers niveaux de précision organisés hiérarchiquement. En plus de l'étiquette initiale, nommée attribut, s'ajoutent des spécificateurs (*specifier*) pouvant être suivis de modificateurs (*modifier*). Ces derniers sont ajoutés à l'étiquette suivant le contexte de la phrase afin de préciser l'annotation sémantique. Le formalisme d'annotation de MEDIA est très riche. Son dictionnaire sémantique comprend 82 attributs et spécificateurs et 21 modificateurs. Des phénomènes linguistiques complexes comme les co-références sont également gérés. La combinaison des attributs et des spécificateurs permet un lexique de 83 concepts. Les étiquettes, spécificateurs et modificateurs de MEDIA, et la base de ceux de PortMEDIA-fr et PortMEDIA-it sont recensés des Figures 4 à 16 placées en Annexes. Les concepts : «objetBD», «evenement», «connectAttr», «commandDial», «reponse», «connectProp», «commandTache», «objet», «unknown» et «null» ; ne sont pas représentés dans ces figures car utilisés sans spécificateurs ni modificateurs.

Il existe deux niveaux d'annotation pour MEDIA : *full* et *relax*. Les statistiques liées au nombre de concepts présents dans chacun des corpus de MEDIA, PortMEDIA-fr et PortMEDIA-it sont données Table 4.1. Les dix concepts les plus courants de chaque ensemble de données et leur nombre d'occurrences sont indiqués dans la Table 4.2. La communauté scientifique utilise plus généralement la version *relax* qui ne prend pas en compte l'utilisation de modificateurs, soit un total de 83 concepts possibles en tenant compte des règles ontologiques, à la place des près de 500 de la version *full*. Pour autant, nous avons pris le parti dès le début de cette thèse de nous concentrer sur la tâche *full* de MEDIA. Cela nous permet de traiter l'ensemble de données PortMEDIA-it simultanément, celui-ci n'étant disponible que dans une version *full*, mais aussi de viser une tâche bien plus complexe d'extraction de concepts sémantiques. Le principal inconvénient de ce choix réside dans le fait que nos résultats soient plus difficilement comparables avec l'état-de-l'art présent ou passé de MEDIA [HAHN et al. 2011 ; VUKOTIC et al. 2015 ; DINARELLI et TELLIER 2016].

Pour PortMEDIA-it, les mêmes concepts utilisés dans la version *full* de MEDIA furent utilisés, bien que tous ne soient pas présents dans les corpora. PortMEDIA-it utilise aussi dix-huit concepts de l'ontologie originelle n'apparaissant jamais dans les corpora de MEDIA.

Le domaine de PortMEDIA-fr étant proche de celui de MEDIA, la plupart des concepts sémantiques ont été préservés. PortMEDIA-fr contient dix nouveaux concepts afin de prendre en compte le contexte du festival : «etat-piece-complet», «nb-billet», «nb-reservation», «nom-lieu», «nom-piece», «numero-reference», «piece-nom-auteur», «type-artiste», «type-billet» et «type-spectacle». Il a été cependant nécessaire de garder une certaine similarité dans l'annotation de ces deux ensembles de données afin de réaliser une portabilité sémantique entre les deux domaines.

Corpus	**Occurrences**			**Lexique**			
	MEDIA	**PM-fr**	**PM-it**	**MEDIA Full**	**MEDIA Relax**	**PM-fr**	**PM-it**
train	$31,7$ k	$9,9$ k	$9,8$ k	144	73	36	125
dev	$3,3$ k	$2,6$ k	$3,5$ k	104	63	31	107
test	$8,8$ k	$5,2$ k	$6,7$ k	125	71	31	117

TABLE 4.1 – Nombre d'occurrences et taille du lexique de concepts en version *full* et *relax* pour MEDIA [BONNEAU-MAYNARD, ROSSET et al. 2005] et dans leur unique version pour PortMEDIA-fr (PM-fr) et PortMEDIA-it (PM-it) [LEFÈVRE, MOSTEFA et al. 2012].

En plus des étiquettes sémantiques, des modes sont associés à chaque concept. Ces modes ne sont généralement pas utilisés et nous ne les utilisons pas non-plus au cours de cette thèse. Le mode peut être : positif par défaut ($+$) , négatif ($-$), interrogatif (?) ou optionnel (~). Un mode positif signifiera que la requête de l'utilisateur est affirmative. Par opposition, un mode négatif signifiera une requête négative, par exemple si un client ne veut pas d'hôtel d'un certain standing. Un mode

MEDIA		PortMEDIA-fr		PortMEDIA-it	
9,8 k	reponse	3,7 k	reponse	5,1 k	reponse
3,5 k	commandTache	2,4 k	commandTache	1,6 k	commandTache
2,7 k	lienRef-coRef	1,6 k	objet	1,0 k	nombre-chambre
2,6 k	objet	1,3 k	nb-billets	0,9 k	chambre-type
2,1 k	connectProp	1,2 k	temps-date	0,7 k	objetBD
1,9 k	chambre-type	0,9 k	nombre	0,6 k	nombre
1,8 k	objetBD	0,8 k	commandDial	0,5 k	localisation-ville
1,8 k	nombre-chambre	0,8 k	nom-piece	0,5 k	objet
1,7 k	paiement-monnaie	0,7 k	temps-heure	0,5 k	localisation-lieurelatif-general
1,3 k	hotel-services	0,7 k	type-billet	0,4 k	temps-date-debut

TABLE 4.2 – Nombre d'occurrences des dix concepts les plus présents dans la version *full* de MEDIA [BONNEAU-MAYNARD, ROSSET et al. 2005] et l'unique version de PortMEDIA-fr et PortMEDIA-it [LEFÈVRE, MOSTEFA et al. 2012].

interrogatif indiquera une question tandis qu'un mode optionnel sera utilisé lors d'une requête facultative de l'utilisateur. Cette précision permet de désambiguïser les segments transcrits.

Une valeur normalisée des mots-support encadrés par les concepts sémantiques est aussi annotée, comme indiquée dans la Figure 2.1. Une date ou un chiffre écrit en toutes lettres sera par exemple normalisé au format numérique. C'est cette valeur normalisée qui est régulièrement utilisée lors de l'évaluation en *CVER* dont nous parlerons plus loin en Section 4.2.3. Pour autant, il n'existe à ce jour pas de système suffisamment performant pour réaliser une telle normalisation des mots-supports générés par un modèle de compréhension de la parole. Nous détaillerons donc dans cette même section nos choix concernant l'utilisation de cette valeur.

4.1.2 MEDIA

L'ensemble de données originel MEDIA contient au total $1\,258$ dialogues téléphoniques d'environ 250 locuteurs différents pour 28h 58m de parole utilisateur et 71h 51m de dialogue total.

Les Tables 4.3 et 4.5 donnent la répartition des heures de parole mais aussi le nombre de dialogues, segments et tours de parole de chaque corpus pour l'ensemble des utilisateurs pour les corpora originels de MEDIA tandis que la Table 4.4 donne les statistiques concernant les mots et mots tronqués dans la transcription. On définira un dialogue comme la totalité de l'enregistrement audio pour un même appel téléphonique. Le tour de parole correspond à la prise de parole sans interruption d'un utilisateur entre deux interventions du magicien d'Oz. Un segment de parole sera issu d'une segmentation de ces tours de parole, réalisée en supprimant les blancs de parole ou

en suivant l'annotation préalable des chutes d'intonation marquant la fin d'une phrase, et sera utilisé comme échantillon lors de l'apprentissage du modèle. Enfin, un mot sera considéré tronqué lorsqu'il ne sera pas complètement ou correctement prononcé. La partie du mot inintelligible est annotée entre parenthèses.

Corpus	Nb. segments	Nb. tours de parole	Nb. dialogues
train	13,7 k	13,0 k	727
dev	1,4 k	1,3 k	79
test	3,8 k	3,5 k	208
pas utilisé	4,0 k	3,8 k	244

TABLE 4.3 – Statistiques sur la composition de l'ensemble de données MEDIA originel [BONNEAU-MAYNARD, ROSSET et al. 2005] en ne tenant compte que des énoncés de l'utilisateur.

Corpus	Occurrences		Lexique	
	Mots	Mots Tronqués	Mots	Mots Tronqués
train	92,6 k	820	2,3 k	372
dev	10,5 k	134	0,8 k	89
test	26,0 k	227	1,4 k	146
pas utilisé	28,0 k	159	1,3 k	107

TABLE 4.4 – Nombre d'occurrences et taille du lexique de mots et mots tronqués dans MEDIA [LAPERRIÈRE, PELLOIN, CAUBRIERE et al. 2022b ; LAPERRIÈRE, PELLOIN, CAUBRIERE et al. 2022a] en ne tenant compte que des énoncés de l'utilisateur.

Corpus	Temps segments			Temps globaux		
	Nb. Heures	Durée Moyenne	Durée Médiane	Nb. Heures	Durée Moyenne	Durée Médiane
train	16h 56m	4,69s	3,12s	42h 10m	209s	194s
dev	01h 40m	4,77s	2,79s	03h 37m	165s	158s
test	04h 47m	4,89s	3,34s	11h 34m	200s	190s
pas utilisé	05h 35m	5,30s	3,86s	14h 30m	214s	196s

TABLE 4.5 – Statistiques sur la durée des segments de l'utilisateur et sur les temps globaux des enregistrements audio (utilisateur, *WoZ* et blancs de parole compris) dans l'ensemble de données MEDIA originel [BONNEAU-MAYNARD, ROSSET et al. 2005].

Ces tables montrent qu'une partie conséquente des données MEDIA n'a pas été utilisée lors de la campagne officielle en 2005, car finalisée après la fin de celle-ci. Par conséquent, même si ces données sont présentes dans l'archive distribuée par ELRA, elles ne sont pas répertoriées officiellement, étant cachées parmi les sous-répertoires qui structurent cette archive. À notre connaissance, ces données n'ont jamais été utilisées dans des travaux de recherche, du moins jusqu'à leur mise en avant dans la recette MEDIA de SpeechBrain réalisée durant cette thèse [LAPERRIÈRE, PELLOIN, CAUBRIERE et al. 2022b ; LAPERRIÈRE, PELLOIN, CAUBRIERE et al. 2022a].

4.1.3 PortMEDIA-fr

L'ensemble de données PortMEDIA-fr contient 697 dialogues téléphoniques pour un total de 12h 33m de parole utilisateur et plus de 40h de dialogue. Notons la différence en terme de nombre de dialogues indiqué dans le papier originel [LEFÈVRE, MOSTEFA et al. 2012] car trois d'entre eux ne sont pas utilisables dû à un manquement dans leur transcription ou la troncation de leur enregistrement audio.

PortMEDIA-fr est un nouvel ensemble de données pour une tâche proche de celle de MEDIA. Il s'agira ici de nouveaux scénarios de réservations et informations concernant le festival d'Avignon de 2010. Les utilisateurs peuvent ainsi acheter ou se renseigner sur les billets d'un spectacle, représentation théâtrale ou autre.

Les statistiques de cet ensemble de données sont données dans les Tables 4.6, 4.7 et 4.8.

Corpus	**Nb. segments**	**Nb. dialogues**
train	$5,9$ k	397
dev	$1,4$ k	100
test	$2,8$ k	200

TABLE 4.6 – Statistiques sur la composition de l'ensemble de données PortMEDIA-fr [LEFÈVRE, MOSTEFA et al. 2012] en ne tenant compte que des énoncés de l'utilisateur.

Corpus	**Occurrences**	**Lexique**
train	$35,6$ k	$1,3$ k
dev	$9,4$ k	$0,7$ k
test	$18,4$ k	$1,0$ k

TABLE 4.7 – Nombre d'occurrences et taille du lexique de mots dans PortMEDIA-fr [LEFÈVRE, MOSTEFA et al. 2012] en ne tenant compte que des énoncés de l'utilisateur.

Corpus	Temps segments		
	Nb. Heures	Durée Moyenne	Durée Médiane
train	07h 15m	4,46s	3,36s
dev	01h 45m	4,50s	3,42s
test	03h 33m	4,59s	3,48s

TABLE 4.8 – Statistiques sur la durée des segments de l'utilisateur dans PortMEDIA-fr [LEFÈVRE, MOSTEFA et al. 2012].

4.1.4 PortMEDIA-it

L'ensemble de données PortMEDIA-it contient 604 dialogues téléphoniques de 130 locuteurs italiens pour un total de 14h 41m de parole utilisateur et environ de 50h de dialogue.

PortMEDIA-it est une version italienne issue des scénarios d'un sous-ensemble de MEDIA. Il ne s'agit pas d'une traduction de MEDIA mais bien d'une traduction des scénarios donnés aux locuteurs italiens. Ces scénarios listent les informations à demander au serveur téléphonique, que l'utilisateur devra utiliser dans ses requêtes avec sa propre formulation. Un exemple de phrase générée automatiquement avec l'aide d'un outil externe français avant la traduction des scénarios est aussi traduit en italien afin d'aider l'utilisateur s'il le souhaite. L'ordre des requêtes peut aussi varier, tout comme la répétition de celles-ci si la réponse du magicien d'Oz n'est pas concluante. Il n'y a donc pas d'alignement entre les dialogues français et italiens, parfois trop différents pour des scénarios identiques.

Les statistiques de cet ensemble de données sont données dans les Tables 4.9, 4.10 et 4.11.

Corpus	Nb. segments	Nb. dialogues
train	5,6 k	300
dev	1,9 k	104
test	3,7 k	200

TABLE 4.9 – Statistiques sur la composition de l'ensemble de données PortMEDIA-it [LEFÈVRE, MOSTEFA et al. 2012] en ne tenant compte que des énoncés de l'utilisateur.

Corpus	Occurrences	Lexique
train	21,7 k	1,4 k
dev	7,7 k	0,9 k
test	14,7 k	1,2 k

TABLE 4.10 – Nombre d'occurrences et taille du lexique de mots dans PortMEDIA-it [LEFÈVRE, MOSTEFA et al. 2012] en ne tenant compte que des énoncés de l'utilisateur.

Corpus	Temps segments		
	Nb. Heures	**Durée Moyenne**	**Durée Médiane**
train	07h 18m	4,78s	3,84s
dev	02h 32m	4,91s	4,01s
test	04h 51m	4,85s	3,92s

TABLE 4.11 – Statistiques sur la durée des segments de l'utilisateur dans PortMEDIA-it [LEFÈVRE, MOSTEFA et al. 2012].

4.2 Métriques d'évaluation

Afin d'évaluer les performances des modèles appris et la qualité des hypothèses qu'ils génèrent en rapport avec des critères propres à la tâche, il est nécessaire de choisir des métriques d'évaluation adaptées. Ces métriques seront appliquées tout au long de l'entraînement sur le corpus de développement afin de donner un aperçu de l'évolution des performances du système, mais aussi lors de l'inférence réalisée sur le corpus de test afin de comparer des systèmes entre eux. Ce sont ces métriques qui détermineront si un système est à l'état-de-l'art vis-à–vis des publications parues jusqu'à présent.

Généralement, les tâches d'extraction sémantique sont évaluées avec l'aide de taux d'erreur. Plus bas est ce taux d'erreur, meilleur est le système. Pour les ensembles de données de MEDIA, il est d'usage d'utiliser un taux d'erreur de concepts (*Concept Error Rate, CER* ou *CoER*) et un taux d'erreur de paires concept-valeur (*Concept Value Error Rate, CVER*). La métrique d'évaluation utilisée pendant la campagne d'évaluation en 2005 [BONNEAU-MAYNARD, AYACHE et al. 2006] est un taux d'erreur de compréhension (*Understanding Error Rate, UER*). Pour autant, nous n'avons compté qu'une étude poursuivant l'évaluation en *UER* avec l'ensemble des annotations et modes de MEDIA [LEHUEN et LEMEUNIER 2010]. Ces métriques sémantiques peuvent être accompagnées de taux d'erreur de transcription, lorsque celle-ci est aussi attendue en sortie du modèle. Il peut s'agir de taux d'erreur de caractères (*Character Error Rate, CER* ou *ChER*) ou de taux d'erreur de mots (*Word Error Rate, WER*). Le *WER* ainsi que les deux taux d'erreur liés aux concepts sémantiques (*CER* et *CVER*) seront présentés dans cette section. Notons que cette thèse utilisera la notation *CER* pour parler de taux d'erreur de concepts et n'abordera pas les résultats obtenus en taux d'erreur de caractères car moins significatif que le *WER*.

Peuvent aussi être utilisées des métriques de classification telles que la Précision, le Rappel et la F-mesure. Ces trois-ci seront aussi présentées à la fin de cette section.

4.2.1 Taux d'erreur de mots (WER)

Le *WER* est une métrique utilisée principalement pour la Reconnaissance Automatique de la Parole. Pour autant, lorsqu'une transcription est attendue en complément d'une tâche de Compréhension Automatique de la Parole, ce taux d'erreur peut être utilisé pour l'évaluer, mettant de côté tout concept ajouté à la transcription. Il permet d'indiquer le taux de mots incorrectement reconnus dans l'hypothèse générée par le modèle en la comparant à la référence fournie dans l'ensemble de données.

Pour ce faire, un alignement de l'hypothèse et de sa référence est réalisé. L'alignement pour le calcul du *WER* est dérivé de la distance de Levenshtein [LEVENSHTEIN 1965], appliquant une comparaison au niveau d'unités de mots plutôt que de caractères. Celle-ci permettra d'indiquer le nombre d'unités correctement placées et reconnues et d'obtenir le nombre de suppressions (D), insertions (I) et substitutions (S) pour n unités initialement dans la référence.

— D correspondra donc au nombre de mots omis dans l'hypothèse.
— I indiquera le nombre de mots ajoutés dans l'hypothèse.
— S donnera le nombre de mots remplacés par un autre, parfois simplement mal orthographiés, dans l'hypothèse.

Un ratio sera par la suite calculé pour donner un taux d'erreur de mots :

$$ErrorRate = \frac{D + I + S}{n} \tag{4.1}$$

Tandis que ce taux ne peux descendre en dessous de 0, il sera possible de dépasser 100. C'est par exemple le cas lorsque beaucoup d'insertions sont faites dans l'hypothèse. Pour autant, il est commun de considérer son résultat comme un pourcentage, un taux dépassant 100 étant de toute manière considéré très mauvais.

4.2.2 Taux d'erreur de concepts (CER)

Le *CER* a été introduit par Raymond et Riccardi [2007] afin de simplifier l'évaluation de MEDIA et est devenu la métrique de référence dans les travaux sur ces ensembles de données [HAHN et al. 2011 ; DINARELLI, KAPOOR et al. 2020 ; GHANNAY, CAUBRIERE, ESTÈVE et al. 2018]. Cette métrique a la particularité d'évaluer l'ordre de la séquence de concepts ainsi que leur nombre d'occurrence, contrairement à des métriques de classification pouvant elles-aussi être utilisées pour des tâches d'extraction sémantique.

Ce taux d'erreur fonctionne de la même sorte que présenté en Équation (4.1) mais ne prend en compte que les concepts sémantiques hors transcription et mots-support. L'alignement est donc réalisé sur ces concepts, considérés chacun comme une unité.

4.2.3 Taux d'erreur de concepts et valeurs (CVER)

Le *CVER* fut premièrement utilisé par Hahn et al. [2011] puis nommé ainsi par Simonnet et al. [2017 ; 2018]. En complément du *CER*, cette métrique est utilisée dans de nombreux travaux [CAUBRIERE, TOMASHENKO et al. 2019 ; GHANNAY, CAUBRIERE, MDHAFFAR et al. 2021 ; PELLOIN, CAMELIN et al. 2021].

Le *CVER* fonctionne lui-aussi comme présenté dans l'Équation (4.1). Contrairement au *CER*, il prend en considération l'ensemble de la paire concept-valeur, c'est-à-dire le concept et les mots-support qui lui correspondent, normalisés sous forme de valeur. On omet alors dans le calcul du *CVER* toute transcription externe aux balises sémantiques. La paire concept-valeur sera une unité considérée correcte lorsque le concept et la valeur seront tous deux exactement identiques à la référence. Une paire incorrecte, manquante ou supplémentaire signifie une unique erreur de substitution, suppression ou insertion.

Normalisation des valeurs

Les mots-support sont généralement normalisés afin d'obtenir une valeur utilisée dans le calcul du *CVER*, comme réalisé en Table 2.1 pour le format BIO. Cette normalisation permet idéalement de générer une valeur propre sans prise en compte de fautes d'orthographe, fautes syntaxiques ou différences dans la formulation des mots-support. Elle permet notamment de mieux normaliser les dates et nombres, en les mettant par exemple au format numérique. C'est Hahn et al. [2011] qui démontra le gain de performances obtenu en normalisant les valeurs grâce à des règles manuellement établies en fonction du corpus d'apprentissage. Pelloin et al. [2021] proposa par la suite une normalisation automatique directe des valeurs après prédiction grâce à un encodeur-décodeur de bout-en-bout avec mécanismes d'attention. De très bons résultats étaient obtenus, mais comme pour d'autres études [HAHN et al. 2011], les règles humaines établies restaient plus performantes.

Évaluation stricte

Nous avons réalisé une analyse de ces règles humaines permettant la normalisation de mots-support en valeurs. Le *CVER* avec valeurs normalisées par règles tel qu'utilisé dans les travaux de la dernière décennie présente des défauts majeurs. Appliqué sur les références, le *CVER* devrait être égal à 0%. Or, en appliquant les même règles de normalisation que dans les travaux cités plus haut, nous obtenons $4,7$% de *CVER* sur le corpus de développement, et $5,7$% sur le corpus de test de MEDIA. Toutes ces erreurs sont des substitutions, c'est-à-dire de mauvaises normalisations de mots-support.

Afin de donner des résultats justes et indépendants d'une normalisation de valeur qui pourrait de ce fait être biaisée ou inefficace, nous avons donc évalué tous les systèmes présentés dans les chapitres suivants avec un *CVER* non-normalisé. Ce dernier sera donc plus strict, et ne pourra

pas être comparé aux *CVER* des travaux à l'état-de-l'art utilisant une normalisation, contrairement aux résultats en *WER* et *CER*. Dès lors qu'un unique caractère des mots-support encadrés par un concept sera incorrect, l'ensemble de la paire concept-valeur sera jugée incorrecte.

Il semble clair que la nouvelle étape du traitement des tâches MEDIA soit l'actualisation de la métrique *CVER* et la redéfinition de ses règles de normalisation. Nous insistons sur le fait qu'un système sans expertise humaine est nécessaire et doit être réfléchi par les chercheurs du domaine.

4.2.4 Précision, Rappel et F-mesure

On peut utiliser la Précision, le Rappel ou la F-mesure [VAN RIJSBERGEN 1974] afin d'obtenir une métrique sans prise en charge de la séquentialité des unités. En compréhension de la parole, on pourra chercher à comptabiliser les concepts prédits et leur nombre d'occurrence sans tenir compte de leur ordre de prédiction dans l'hypothèse et sa référence.

Le Rappel et la Précision sont habituellement calculés pour chaque classe avant d'en réaliser la moyenne pondérée. Ces indicateurs ne sont pour autant pas suffisants à l'évaluation d'un système, souvent accompagnés de ou remplacés par une F-mesure. Cette métrique utilise les deux précédentes afin d'obtenir un score plus représentatif des performances du système en en réalisant une moyenne harmonique. Contrairement aux taux d'erreur, ces trois métriques indiqueront de meilleurs performances lorsqu'elles approcheront 100%.

Afin de mieux comprendre les calculs qui suivent, il est important de connaître les notions de taux, expliquées ci-dessous du point de vue de la prédiction de concepts :

- **Vrai Positif** (*True Positive, TP*) : un concept prédit (Positif) par le système et bien présent (Vrai) dans la référence ;
- **Vrai Négatif** (*True Negative, TN*) : un concept non-prédit (Négatif) par le système et bien absent (Vrai) de la référence ;
- **Faux Positif** (*False Positive, FP*) : un concept prédit (Positif) par le système mais absent (Faux) de la référence ;
- **Faux Négatif** (*False Negative, FN*) : un concept non-prédit (Négatif) par le système mais présent (Faux) dans la référence ;

La Précision représente le taux de concepts correctement prédits par le système sur la totalité des concepts qu'il aura émis. Autrement dit, on comptabilisera le nombre de Vrais Positifs (VP) sur le nombre de Vrais Positifs (VP) et Faux Positifs (FP).

$$Precision = \frac{VP}{VP + FP} \tag{4.2}$$

Le Rappel donnera le taux de concepts correctement prédits sur le nombre de concepts attendus dans la référence. On parlera ici de compter le nombre de Vrais Positifs (VP) en les divisant

par le nombre de Vrais Positifs (VP) et Faux Négatifs (FN).

$$Rappel = \frac{VP}{VP + FN} \tag{4.3}$$

Le calcul de la F-mesure se fait ainsi :

$$F\text{-}mesure = 2 * \frac{Precision * Rappel}{Precision + Rappel} \tag{4.4}$$

Soit, en utilisant les taux abordés précédemment :

$$F\text{-}mesure = \frac{2 * VP}{2 * VP + FP + FN} \tag{4.5}$$

Toutes ces métriques peuvent être exprimées en pourcentages ou utilisées comme des taux.

4.2.5 Mesures de confiance

Les mesures de confiance permettent d'évaluer les métriques utilisées. Elles déterminent la fiabilité du résultat obtenu via un intervalle de confiance ou une marge d'erreur. Cette évaluation est une information importante, le résultat de l'apprentissage d'un système de compréhension de la parole ayant toujours une part d'aléatoire.

Un intervalle de confiance est par définition un intervalle dans lequel se situe une valeur que nous cherchons à estimer. Dans notre cas, il s'agit d'encadrer un taux d'erreur (*WER*, *CER* ou *CVER*) par un intervalle qui indique la variabilité possible du résultat. L'intervalle de confiance à 95% utilisé dans cette thèse suit la loi de Student. Un intervalle à 95% symbolise son seuil en signifiant qu'il a 95% de chance de contenir la valeur à estimer. Il est calculé comme suit pour ER un taux d'erreur et n le nombre d'unités évaluées dans la référence, avant d'être soustrait et additionné au taux d'erreur en question afin d'obtenir les bornes de l'intervalle :

$$I = 1,96 * \sqrt{ER * \frac{1 - ER}{n}} \tag{4.6}$$

Nous indiquons aussi pour nos expériences des marges d'erreur, obtenues en lançant 5 fois le même apprentissage avec une initialisation aléatoire différente des paramètres du système évalué. Par soucis de limitation des ressources de calcul utilisées, nous ne donnons cette marge d'erreur que pour les systèmes que nous jugeons pertinents.

4.3 Recette MEDIA SpeechBrain

Bien que parmi les plus riches et complexes à traiter, l'ensemble de données françaises MEDIA est rarement utilisé au-delà de la communauté scientifique française. Pourtant, MEDIA fait partie des très rares données disponibles pour une tâche d'extraction sémantique depuis la parole. Pour faciliter son utilisation et rendre MEDIA plus accessible, nous avons donc réalisé une

recette [3] liant préparation des données et apprentissage de bout-en-bout [LAPERRIÈRE, PELLOIN, CAUBRIERE et al. 2022b ; LAPERRIÈRE, PELLOIN, CAUBRIERE et al. 2022a] grâce à la boite à outils SpeechBrain [RAVANELLI, PARCOLLET et al. 2021]. Le choix de cet outil dédié au traitement de tâches d'Intelligence Artificielle conversationnelle s'est fait de part sa communauté grandissante à l'international mais aussi car il est tout-en-un et open-source. L'avantage d'intégrer notre recette MEDIA dans SpeechBrain est la garantie de garder un code source persistant et maintenu au gré des évolutions des prochaines années.

Des corrections ont été apportées aux annotations manuelles de MEDIA et de nombreuses données collectées durant la création du corpus original mais jamais utilisées ont été regroupées pour former un nouveau corpus de test que nous nommons *test2*. Ce sont ces données qui sont présentées dans les tables de ce chapitre sous l'intitulé «pas utilisé». Nous avons dû générer la notation *relax* des concepts sémantiques présents dans *test2*, car les données étaient uniquement annotées en notation *full*. Tous ces corpora peuvent être préparés via le code fourni dans la recette SpeechBrain. Celui-ci générera des fichiers au format CSV (*Comma-Separated Values*), considérablement plus aisés à prendre en mains que les fichiers XML (*Extensible Markup Language*) fournis initialement par ELRA.

Cette recette nous a aussi permis d'intégrer les métriques d'évaluation *CER* et *CVER* à SpeechBrain, pouvant être utilisées pour toute tâche d'extraction sémantique hors MEDIA dans la boîte à outils. Le *CVER* qui y est intégré est un *CVER* strict, comme décrit en Section 4.2.3, ayant déjà alors soulevé le problème d'évaluation par règles humaines pour MEDIA.

Traitement des données

Le langage naturel étant sujet à interprétation, des erreurs ont pu être faites lors de la transcription et annotation manuelle de MEDIA et induire de fausses erreurs d'évaluation. C'est pourquoi nous avons proposé via la recette SpeechBrain une correction des données. Outre l'orthographe, nous avons retiré les redondances d'espaces, corrigé des connexions d'apostrophes et traits d'union ainsi que la casse de certains noms propres. Les transcriptions des données MEDIA considèrent des détails de prononciation, comme des mots tronqués ou trop proches, indiqués par des parenthèses ou astérisques. Ces annotations peuvent ne pas avoir été traitées de la même manière dans les différentes études, rendant les résultats expérimentaux publiés difficilement comparables. Notre recette propose donc une uniformisation systématique de ces mots tronqués.

L'information changeante du canal audio (droite ou gauche) d'enregistrement de la voix utilisateur a aussi été partagée. Nous avons traité des problèmes de segmentation audio comme la présence de tonalité de fin d'appel dans les segments. Il s'agissait cependant principalement de réaliser une segmentation plus précise des données grâce à des balises déjà présentes dans les

3. https://github.com/speechbrain/speechbrain/tree/develop/recipes/MEDIA

fichiers XML mais rarement utilisées par la communauté. Ces balises de synchronisation temporelle nous ont permis de réduire les blancs de parole présents dans les segments. La Table 4.12 donne la répartition des heures de parole de chaque corpus pour les utilisateurs de MEDIA après cette nouvelle segmentation.

Enfin, l'identifiant de certains utilisateurs a été corrigé afin d'en respecter le format initial.

Corpus	**Temps segments**		
	Nb. Heures	**Durée Moyenne**	**Durée Médiane**
train	10h 52m	$2,85$s	$1,69$s
dev	01h 13m	$3,23$s	$1,91$s
test	03h 01m	$2,88$s	$1,70$s

TABLE 4.12 – Statistiques sur la durée des segments de l'utilisateur dans l'ensemble de données MEDIA après nouvelle segmentation [LAPERRIÈRE, PELLOIN, CAUBRIERE et al. 2022b ; LAPERRIÈRE, PELLOIN, CAUBRIERE et al. 2022a].

Au lancement de la recette, les données MEDIA sont préparées automatiquement. Pour la recette, la majorité des caractères spéciaux ont été retirés afin de suivre la norme SpeechBrain concernant le traitement des données. Pour autant, nous avons gardé les chevrons servant à encadrer les concepts et leurs mots-support. L'apostrophe a été gardée et rattachée au mot précédent pour limiter la taille du vocabulaire, hormis pour «c'est» car très commun. Seuls les traits d'union des nombres ont été retirés pour la même raison car ne participant pas suffisamment à comprendre le dialogue. Des astérisques ont été ajoutés aux mots tronqués, changeant «bon(jour)», une prononciation ambiguë de «bonjour», par «bon*». L'indication est donc gardée sans créer un nouveau mot dans le lexique.

Apprentissage

La recette permet l'apprentissage et l'évaluation des données MEDIA pour sa tâche de Reconnaissance Automatique de la Parole [4] et sa tâche de Compréhension Automatique de la Parole [5].

Une architecture de bout-en-bout permet de réaliser le *fine-tuning* du modèle LeBenchmark 3k large [EVAIN, H. NGUYEN et al. 2021] déjà présenté au Chapitre 3. L'encodeur de parole est ensuite suivi de 3 couches denses de 512 neurones activées par LeakyReLU puis d'une couche linéaire de même dimension et enfin d'une Softmax. Une CTC [GRAVES, FERNÁNDEZ et al. 2006] est utilisée en fonction de coût. Les paramètres de ces couches sont initialisés aléatoirement. Nous

4. https://github.com/speechbrain/speechbrain/tree/develop/recipes/MEDIA/ASR/CTC
5. https://github.com/speechbrain/speechbrain/tree/develop/recipes/MEDIA/SLU/CTC

utilisons l'optimiseur Adam pour l'encodeur de parole avec un taux d'apprentissage de $0,0001$, et AdaDelta pour les autres couches avec un taux de 1 et momentum de $0,95$, le tout sur 30 époques. Ces hyper-paramètres et bien d'autres peuvent être optimisés par l'utilisateur de la recette, notre objectif ayant été de diffuser la tâche MEDIA à travers une recette viable et non un système à l'état-de-l'art.

Le réseau neuronal reçoit en entrée le signal audio de fichiers WAV échantillonnés à 16 kHz et génère une séquence de caractères issus du lexique de la tâche. Un décodeur glouton à la sortie de la couche Softmax sélectionne les caractères finaux que nous avons évalués.

La recette propose l'utilisation d'un encodeur de parole préalablement *fine-tuné* de manière supervisée pour une tâche de reconnaissance de la parole sur $425,5$ heures de transcriptions françaises supplémentaires issues de l'ensemble de données français CommonVoice [6] (version 6.1). Les Tables 4.13 et 4.14 présentent les résultats de ces deux architectures pour l'ensemble des corpora de MEDIA :

— LeBenchmark : avec un *fine-tuning* direct de l'encodeur sur MEDIA pour sa tâche *SLU*.
— LeBenchmark-CommonVoice : avec un premier *fine-tuning* de l'encodeur sur CommonVoice pour une tâche *ASR* avant son *fine-tuning* sur MEDIA pour sa tâche *SLU*.

Version	**Modèle**	***dev***		***test***	
		CER	*CVER*	*CER*	*CVER*
full	LeBenchmark	$28,9$	$41,2$	$26,1$	$37,5$
	LeBenchmark-CommonVoice	$24,0$	$34,4$	$20,3$	$30,8$
relax	LeBenchmark	$23,3$	$37,1$	$21,8$	$34,1$
	LeBenchmark-CommonVoice	$18,1$	$30,4$	$16,3$	$27,7$

TABLE 4.13 – Résultats en *CER* et *CVER* sur les corpora *dev* et *test* de MEDIA en version *full* et *relax* avec les modèles LeBenchmark et LeBenchmark-CommonVoice.

Version	***test2***	
	CER	*CVER*
full	$21,1$	$30,9$
relax	$16,4$	$27,1$

TABLE 4.14 – Résultats *CER* et *CVER* sur le corpus *test2* de MEDIA en version *full* et *relax* avec le modèle LeBenchmark-CommonVoice.

6. https://commonvoice.mozilla.org/fr/datasets

4.4 Conclusion

Ce chapitre présente les données principalement utilisées dans cette thèse : MEDIA [BONNEAU-MAYNARD, ROSSET et al. 2005] et PortMEDIA-it [LEFÈVRE, MOSTEFA et al. 2012] ; mais aussi un autre ensemble de données fortement lié bien que moins utilisé dans nos expérimentations : PortMEDIA-fr [LEFÈVRE, MOSTEFA et al. 2012]. Nous y détaillons précisément la base de leur ontologie, liée principalement aux données initiales de MEDIA, mais aussi diverses statistiques concernant leur annotation en concepts sémantiques et leur transcription.

Nous insistons sur notre volonté à traiter la tâche MEDIA dans sa version *full* plus complexe bien que la communauté scientifique tend à se focaliser sur sa version *relax*. C'est aussi la seule version disponible pour PortMEDIA-it, ensemble de données que nous utiliserons grandement dans nos recherches de portabilité cross-lingue et multilingue.

Par la suite, nous mettons en lumière une problématique liée à l'évaluation de ces tâches d'extraction sémantique. Le *CVER* y est discuté et une solution temporaire est apportée en proposant de ne pas utiliser les règles humaines établies par le passé pour la normalisation des transcriptions générées. À la place, un *CVER* strict sera utilisé pour l'ensemble des expérimentations de cette thèse, comme détaillé en Section 4.2.3.

Enfin, après avoir discuté de l'évaluation par intervalle de confiance et marge d'erreur de nos futurs résultats, nous avons présenté la recette MEDIA [LAPERRIÈRE, PELLOIN, CAUBRIERE et al. 2022b ; LAPERRIÈRE, PELLOIN, CAUBRIERE et al. 2022a] intégrée à la boîte à outils SpeechBrain [RAVANELLI, PARCOLLET et al. 2021]. Cette recette a pour objectif la popularisation de la tâche riche et complexe de MEDIA pour la compréhension de la parole tout en garantissant un code source persistant et dûment maintenu.

Un autre ensemble de données grandement utilisé dans nos travaux de recherche sera présenté en détails en Section 5.1 du chapitre suivant. Cet ensemble de données nommé TARIC-SLU [MDHAFFAR, BOUGARES et al. 2024] fut proposé très récemment pour une tâche d'extraction sémantique depuis des dialogues humain-machine tunisiens.

Nous considérons cette langue sémitique afin d'étudier le transfert de connaissances sémantiques et linguistiques entre des langues distantes, le français et l'italien étant deux langues latines très proches ne nous le permettant pas. Nous souhaitons par ailleurs étudier dans cette thèse le traitement de dialectes oraux peu dotés, étant d'autant plus sous-représentés dans le domaine de la compréhension de la parole. De plus, l'utilisation de l'ensemble de données TARIC-SLU nous permettra d'évaluer les capacités de nos modules d'encodage de la parole utilisés dans les chapitres suivants à traiter des langues jamais vues lors de leur pré-apprentissage multilingue.

Chapitre 5

Enrichissement sémantique d'un encodeur de parole

Sommaire

Publications liées à ce chapitre

- **JSALT 2022** – Multi-lingual speech to speech translation for under-resourced languages [LARCHER et al. 2022]
- **SLT 2023** – On the use of semantically-aligned speech representations for Spoken Language Understanding [LAPERRIÈRE, PELLOIN, ROUVIER et al. 2023]
- **SASB 2023** – Specialized semantic enrichment of speech representations [LAPERRIÈRE, H. NGUYEN et al. 2023b]
- **Interspeech 2023** – Semantic enrichment towards efficient speech representations [LAPERRIÈRE, H. NGUYEN et al. 2023a]
- **IWSLT 2023** – ON-TRAC consortium systems for the IWSLT 2023 dialectal and low-resource speech translation tasks [LAURENT et al. 2023]
- **Interspeech 2024** – A dual task learning approach to fine-tune a multilingual semantic speech encoder for Spoken Language Understanding [LAPERRIÈRE, GHANNAY et al. 2024]

Les encodeurs de parole auto-supervisés sont depuis quelques années de plus en plus présents dans le domaine de la Compréhension Automatique de la Parole et de la Traduction Automatique. Comme énoncé au Chapitre 2, les tâches de compréhension de la parole, notamment celle d'extraction sémantique, peuvent être perçues comme des tâches de traduction d'une langue source naturelle vers une langue cible purement sémantique. C'est pour cette raison que cette thèse s'est naturellement intéressée aux encodeurs de parole à l'état-de-l'art dans le domaine de la Traduction Automatique afin de résoudre une tâche d'extraction sémantique depuis la parole.

Ces dernières années ont vu naître des encodeurs de parole multi-modaux enrichis sémantiquement de par l'utilisation d'encodeurs textuels multilingues performants. C'est le cas de SAMU-XLSR [KHURANA et al. 2022], présenté au Chapitre 3, qui réalise l'alignement sémantique de ses représentations de la parole aux représentations textuelles de haut niveau issues de LaBSE [FENG et al. 2022], les rendant intrinsèquement cross-lingues et agnostiques à la langue.

Ce chapitre présente notre étude de cet encodeur de parole enrichi sémantiquement face à l'encodeur de parole XLS-R [BABU et al. 2022] duquel il est issu, pour une tâche complexe de compréhension de la parole. Nous y détaillons nos démarches d'analyse de ces modèles et les architectures neuronales retenues pour nos expérimentations et analysons leurs résultats.

Dans la poursuite de cet objectif d'enrichissement sémantique d'un encodeur de parole, nous proposons un *fine-tuning* de SAMU-XLSR référé dans cette thèse en tant que «spécialisation». Cette spécialisation reprend avec exactitude le processus de pré-apprentissage multilingue de SAMU-XLSR avec l'utilisation des représentations textuelles de LaBSE. Elle sera réalisée sur les transcriptions brutes et les enregistrements audio des ensembles de données que nous utilisons pour notre tâche d'extraction sémantique. En entraînant SAMU-XLSR sur le domaine lexical et

l'environnement audio ciblés lors de nos expérimentations *SLU*, nous espérons spécialiser efficacement l'enrichissement sémantique de cet encodeur de parole. Nous souhaitons ainsi tirer partie du premier pré-apprentissage cross-lingue de SAMU-XLSR et l'affiner pour notre tâche cible avant d'utiliser l'encodeur de parole pour l'extraction de ses concepts sémantiques.

Nous présenterons ensuite nos expérimentations concernant l'utilisation de la représentation vectorielle au *sentence-level* de SAMU-XLSR et une nouvelle méthode de spécialisation de cet encodeur fusionnée à notre apprentissage *SLU*.

Notons que nous nous focalisons sur l'apprentissage de systèmes de bout-en-bout pour leurs avantages face aux systèmes en cascade, énoncés au Chapitre 2, notamment pour le traitement de tâches complexes d'extraction sémantique.

L'ensemble de ces expériences a été réalisé sur l'ensemble de données françaises MEDIA [BONNEAU-MAYNARD, ROSSET et al. 2005] et italiennes PortMEDIA-it [LEFÈVRE, MOSTEFA et al. 2012] décrits au Chapitre 4. Elles ont été poursuivies sur l'ensemble de données tunisiennes TARIC-SLU [MDHAFFAR, BOUGARES et al. 2024], présenté en Section 5.1, et l'ensemble de données tamasheq Tamasheq-French Parallel Corpus [ZANON BOITO et al. 2022].

5.1 Enrichissement sémantique

Ces travaux font suite à une collaboration entre de nombreux chercheurs et étudiants lors de l'atelier JSALT 2022 (*Frederick Jelinek Memorial Summer Workshop on Speech and Language Technology*), dont M. Sameer Khurana et Dr. Antoine Laurent tous deux auteurs de SAMU-XLSR [2022]. Nos travaux furent financés par l'Union Européenne sous les fonds Horizon 2020 du programme d'innovation Marie Skodowska-Curie pour le projet ESPERANTO (subvention numéro 101007666). Notre groupe de recherche s'était alors penché sur la réalisation d'un système multimodal pour une tâche principale de Traduction Automatique textuelle et orale multilingue traitant entre autres de langues peu dotées telles que le tamasheq [LARCHER et al. 2022].

Nous nous sommes orientés vers une utilisation de SAMU-XLSR commune à toutes ses modalités, souhaitant créer un espace sémantique partagé entre les différentes tâches. Nous avons réalisé un certain nombre d'analyses sur sa capacité à encoder la sémantique, dont une étude d'extraction de ses représentations vectorielles couche-par-couche [LAPERRIÈRE, PELLOIN, ROUVIER et al. 2023], présentée en Section 5.1.3. Cette section décrira les architectures retenues pour l'utilisation de cet encodeur de parole enrichi sémantiquement ainsi que les résultats expérimentaux obtenus pour des apprentissages *SLU*.

5.1.1 Architectures neuronales

Cette section présente les architectures neuronales utilisées pour la résolution des tâches d'extraction sémantique de MEDIA, PortMEDIA-it et TARIC-SLU dans le cadre de l'analyse de l'enrichissement sémantique de SAMU-XLSR réalisé lors de son premier apprentissage.

Lors de l'évaluation des capacités de SAMU-XLSR et XLS-R appliquée à MEDIA et PortMEDIA-fr [LAPERRIÈRE, PELLOIN, ROUVIER et al. 2023], une rapide étude comparative a été menée afin de choisir les modules à placer au-dessus de ces encodeurs de parole.

Nous nous sommes attardés sur la comparaison de diverses couches récurrentes telles que les *GRU*, *Li-GRU*, *LSTM*, et *RNN* classiques présentées au Chapitre 1, qu'elles soient unidirectionnelles ou bidirectionnelles, mais aussi sur l'utilisation ou non de couches denses après ces récurrences. Les couches *bi-LSTM* étant souvent les plus appropriées pour le traitement séquentiel de la parole [MOUMEN et PARCOLLET 2023], ce sont elles qui donnèrent assez naturellement les meilleurs résultats. Nous nous sommes aussi intéressés au nombre de couches récurrentes à utiliser (allant de 0 à 4) ainsi qu'à leur nombre de neurones (512, $1\,024$ ou $2\,048$). Ces expérimentations ont été par la suite pour certaines ré-itérées pour le traitement des données TARIC-SLU dont les résultats sont présentés en Tables 5.1 et 5.2.

La Figure 5.1 présente l'architecture neuronale *SLU* utilisée pour nos premières expérimentations sur les données MEDIA et PortMEDIA-it pour l'étude d'un encodeur de parole enrichi sémantiquement [LAPERRIÈRE, PELLOIN, ROUVIER et al. 2023]. L'encodeur inscrit correspond à SAMU-XLSR ou XLS-R, dont les paramètres peuvent être *fine-tunés* durant l'apprentissage ou laissés figés. Lorsqu'ils seront *fine-tunés*, nous souhaiterons évaluer les capacités maximales de ces encodeurs à fournir une représentation propice à l'extraction de sémantique. Lorsqu'ils seront figés, nous évaluerons celles acquises lors de leur pré-apprentissage multilingue sur des langues proches ou identiques mais un domaine et environnement audio différent de celui visé. Les paramètres de toutes les couches supérieures seront appris depuis une initialisation aléatoire.

Le modèle prend en entrée un signal audio échantillonné sous 16 kH au format WAV et produit une transcription annotée sémantiquement comme : Je <*commandTache*> voudrais réserver > <*nombre-chambre*> une > <*chambre-type*> chambre simple > sur <*localisation-ville*> Paris >. L'encodeur de parole fournira une représentation vectorielle pour chaque trame audio de 20 millisecondes. Pour SAMU-XLSR, nous prenons donc les représentations intermédiaires au niveau trame avant la réalisation du sous-échantillonage les réunissant en un unique vecteur par segment.

Ces représentations de la parole sont ensuite fournies à trois couches *bi-LSTM* de $1\,024$ neurones permettant de contextualiser d'avantage les trames entre elles. Leurs sorties passent ensuite par trois couches denses de même dimension avant d'être traitées par une fonction Softmax et évaluées avec une fonction de coût CTC [GRAVES, FERNÁNDEZ et al. 2006].

Toutes ces couches sont activées par une LeakyReLU. Les couches denses sont optimisées avec l'aide d'AdaDelta [ZEILER 2012] dont le taux d'apprentissage est fixé à $1,0$. Adam est utilisé pour l'optimisation des couches récurrentes mais aussi de l'encodeur de parole lorsque celui-ci est *fine-tuné* durant l'apprentissage. Son taux d'apprentissage est initialement de $0,0001$.

Afin d'évaluer le modèle avec les différents corpora de développement, nous considérons la métrique de *CER* présentée au Chapitre 4.

Souhaitant premièrement étudier les différences d'encodage entre SAMU-XLSR et XLS-R pour notre tâche d'extraction sémantique, l'architecture *SLU* s'est vue évoluer au fil de nos travaux. Cette évolution est cependant légère, étant principalement concentrée sur le nombre de couches denses utilisées à la suite des couches récurrentes, passant de trois à une [LAPERRIÈRE, H. NGUYEN et al. 2023b ; LAPERRIÈRE, H. NGUYEN et al. 2023a], comme illustré par la Figure 5.2. C'est cette nouvelle architecture qui est utilisée pour nos expérimentations détaillées dans cette section. L'apprentissage de ce modèle se fait sur $387,8$ M de paramètres lorsque l'encodeur de parole est *fine-tuné*, contre $71,5$ M lorsque celui-ci est figé.

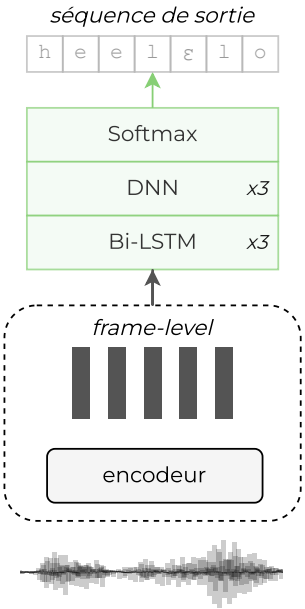

FIGURE 5.1 – Première architecture neuronale *SLU* pour l'étude de l'enrichissement sémantique de SAMU-XLSR face à l'encodeur de parole XLS-R originel [LAPERRIÈRE, PELLOIN, ROUVIER et al. 2023].

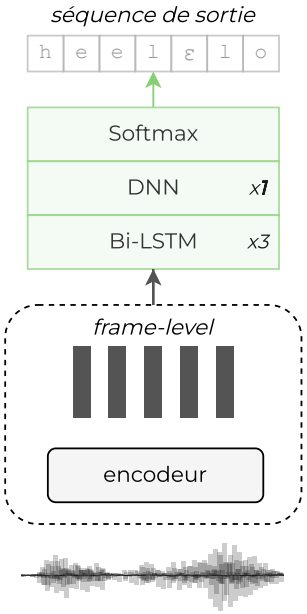

FIGURE 5.2 – Seconde architecture neuronale *SLU* pour l'étude de l'enrichissement sémantique de SAMU-XLSR face à l'encodeur de parole XLS-R originel, aussi utilisée lors de futures expérimentations sur la spécialisation de son enrichissement sémantique [LAPERRIÈRE, H. NGUYEN et al. 2023b ; LAPERRIÈRE, H. NGUYEN et al. 2023a ; LAPERRIÈRE, GHANNAY et al. 2024].

Nos collaborations pour le traitement de l'ensemble de données TARIC-SLU avec ses auteurs nous ont poussés à analyser de nouveau les hyper-paramètres liés à ces diverses couches, spécifiquement pour ces données. Les résultats de ces expérimentations sont présentés dans les Tables 5.1 et 5.2. Sont indiqués les scores de *WER*, *CER* et *CVER* pour les encodeurs de pa-

role XLS-R ou SAMU-XLSR figés ou *fine-tunés*. Suivent ensuite les couches détaillées dans la seconde colonne de ces tables. Leur nombre de neurones est indiqué entre parenthèses. Est ajoutée systématiquement une dernière couche Dense de $1\,024$ neurones.

Nous pouvons interpréter ces résultats en confirmant l'utilisation pertinente de couches *bi-LSTM* en comparaison à des couches Denses, ce pour des encodeurs de parole figés et *fine-tunés*. Le seul doute persistant repose sur le *fine-tuning* de XLS-R, donnant de légèrement moins bons résultats lors de l'utilisation de telles couches. Afin de pouvoir comparer les deux encodeurs de parole, nous avons fait le choix de préserver la structure la plus apte à donner les meilleurs résultats, c'est-à-dire trois couches *bi-LSTM* de $1\,024$ neurones avant la couche Dense de sortie.

		WER	*CER*	*CVER*
figé	1 Dense (1 024)	$97,8$	$85,2$	$98,8$
	3 Dense (512)	$103,9$	$82,9$	$101,2$
	3 Dense (1 024)	$99,8$	$84,9$	$101,4$
	3 *bi-LSTM* (1 024)	**56,4**	**47,4**	**70,8**
fine-tuné	1 Dense (1 024)	**36,3**	$33,0$	**50,2**
	3 Dense (512)	$36,4$	$33,6$	$51,1$
	3 Dense (1 024)	$37,5$	**32,0**	$51,2$
	3 *bi-LSTM* (1 024)	$38,0$	$32,9$	$50,9$

TABLE 5.1 – Résultats du corpus de *test* de la première version de TARIC-SLU [MDHAFFAR, BOUGARES et al. 2024] en *WER*, *CER* et *CVER* lors des expérimentations sur les dimensions et nombres de couches suivant l'encodeur de parole XLS-R figé ou *fine-tuné*.

		WER	*CER*	*CVER*
figé	1 Dense (1 024)	$99,1$	$87,4$	$99,7$
	3 Dense (512)	$105,3$	$83,7$	$101,0$
	3 Dense (1 024)	$103,0$	$79,7$	$102,0$
	3 *bi-LSTM* (1 024)	**62,1**	**49,6**	**74,4**
fine-tuné	1 Dense (1 024)	$31,2$	$30,6$	$48,8$
	3 Dense (512)	$31,6$	$31,3$	$49,1$
	3 Dense (1 024)	$36,3$	$33,0$	$50,2$
	3 *bi-LSTM* (1 024)	**30,7**	**29,1**	**47,2**

TABLE 5.2 – Résultats du corpus de *test* de la première version de TARIC-SLU [MDHAFFAR, BOUGARES et al. 2024] en *WER*, *CER* et *CVER* lors des expérimentations sur les dimensions et nombres de couches suivant l'encodeur de parole SAMU-XLSR figé ou *fine-tuné*.

5.1.2 Résultats expérimentaux

Cette section présente les résultats expérimentaux obtenus suite à l'apprentissage *SLU* des données françaises MEDIA, italiennes PortMEDIA-it et tunisiennes TARIC-SLU avec l'architecture présentée en Figure 5.2. L'objectif de ces expérimentations est d'évaluer l'apport de l'enrichissement sémantique obtenu lors du pré-apprentissage de l'encodeur de parole SAMU-XLSR en le comparant au modèle multilingue XLS-R sur lequel il se base.

Tous les apprentissages ont été réalisés sur une carte graphique V100 à 32 Go de mémoire. L'apprentissage *SLU* pour 100 époques aura été réalisé en 27h pour MEDIA, $16,5$h pour PortMEDIA-it et $20,5$h pour TARIC-SLU lorsque l'encodeur de parole était *fine-tuné*. Lorsque celui-ci était figé, une diminution d'environ 40% de ces temps de traitement a été observée. On réalisera ainsi 100 époques en 12h pour MEDIA, 10h pour PortMEDIA-it et 12h pour TARIC-SLU.

Les intervalles de confiance à 95% calculés suivant la loi de Student sont respectivement de $0,4$, $0,7$ et $1,0$ points de *CER* pour le corpus de *test* de chaque ensemble de données. Ce *CER* a une marge d'erreur d'environ $0,4$ points pour chaque ensemble de données, définie via cinq apprentissages différant uniquement d'un changement d'initialisation aléatoire des paramètres du système. Nous estimons une amélioration des performances pour un ensemble de données comme étant pertinente lorsque celle-ci dépasse l'intervalle de confiance qui lui est attribué.

Étude du français

L'apprentissage textuel de LaBSE [FENG et al. 2022] comprend $854\ 297$ échantillons français et $100\ 059$ alignements entre des phrases françaises et anglaises. L'apprentissage sur la parole du modèle XLS-R [BABU et al. 2022] comprend $23\ 973$ segments audio français. SAMU-XLSR [KHURANA et al. 2022] réalise ensuite son *fine-tuning* grâce à LaBSE sur 826 segments de parole français supplémentaires, issus de la version 8.0 de CommonVoice.

La Table 5.3 donne les résultats obtenus en *WER*, *CER* et *CVER* sur le corpus de *test* de MEDIA. Les performances de nos systèmes en figeant les paramètres des encodeurs de parole nous démontrent l'enrichissement sémantique conséquent obtenu pour l'encodeur de parole XLS-R après son *fine-tuning* lors de l'apprentissage de SAMU-XLSR. En poursuivant l'apprentissage des encodeurs de parole sur notre tâche cible d'extraction sémantique, on peut voir que ce gain de performances pour le traitement du français reste impactant, SAMU-XLSR donnant un *CER* de $18,7$ face à $21,8$ pour XLS-R.

La faible différence de *WER* pour des encodeurs de parole figés peut être associée à l'agnosticisme à la langue de SAMU-XLSR. Celui-ci apprenant à s'aligner sur des représentations textuelles elles-mêmes agnostiques à la langue, il semble cohérent que l'amélioration de ses performances ne se situe pas spécifiquement sur sa capacité à générer une transcription, mais bien à capter et encoder la sémantique d'une phrase quelle qu'en soit sa nature.

		WER	*CER*	*CVER*
figé	XLS-R + SLU $_{FR}$	$21,7$	$33,8$	$46,0$
	SAMU-XLSR + SLU $_{FR}$	**21,3**	**27,4**	**41,6**
fine-tuné	XLS-R + SLU $_{FR}$	$13,5$	$21,8$	$32,8$
	SAMU-XLSR + SLU $_{FR}$	**11,7**	**18,7**	**29,4**

TABLE 5.3 – Résultats du corpus de *test* de MEDIA en *WER*, *CER* et *CVER* avec l'architecture *SLU* présentée en Figure 5.2 pour l'analyse de l'enrichissement sémantique réalisé lors du pré-apprentissage de SAMU-XLSR en comparaison avec XLS-R figés et *fine-tunés*.

Étude de l'italien

L'apprentissage textuel de LaBSE [FENG et al. 2022] comprend $179\ 279$ échantillons italiens et $100\ 000$ alignements entre des phrases italiennes et anglaises. L'apprentissage sur la parole du modèle XLS-R [BABU et al. 2022] comprend $21\ 943$ segments audio italiens. SAMU-XLSR [KHURANA et al. 2022] réalise ensuite son *fine-tuning* grâce à LaBSE sur 310 segments de parole italiens supplémentaires, issus de la version 8.0 de CommonVoice.

La Table 5.4 donne les résultats obtenus en *WER*, *CER* et *CVER* sur le corpus de *test* de PortMEDIA-it. On peut y voir les mêmes gains de performances que pour MEDIA avec néanmoins une amélioration plus nette du *WER* pour la génération des transcriptions italiennes lorsque les encodeurs de parole sont figés.

		WER	*CER*	*CVER*
figé	XLS-R + SLU $_{IT}$	$33,7$	$42,1$	$57,1$
	SAMU-XLSR + SLU $_{IT}$	**31,5**	**33,6**	**49,0**
fine-tuné	XLS-R + SLU $_{IT}$	$18,1$	$29,6$	$41,5$
	SAMU-XLSR + SLU $_{IT}$	**15,4**	**26,6**	**39,2**

TABLE 5.4 – Résultats du corpus de *test* de PortMEDIA-it en *WER*, *CER* et *CVER* avec l'architecture *SLU* présentée en Figure 5.2 pour l'analyse de l'enrichissement sémantique réalisé lors du pré-apprentissage de SAMU-XLSR en comparaison avec XLS-R figés et *fine-tunés* [LAPERRIÈRE, H. NGUYEN et al. 2023b ; LAPERRIÈRE, H. NGUYEN et al. 2023a].

Étude du tunisien

TARIC-SLU est un ensemble de données tunisiennes faisant suite à l'ensemble de données TARIC [MASMOUDI et al. 2014] distribué pour une tâche de reconnaissance de la parole. Mdhaffar

et al. [2024] poursuivirent l'annotation des données TARIC en y incorporant des concepts sémantiques pour une tâche d'extraction sémantique visant la Compréhension Automatique de la Parole dans un contexte de dialogues humain-machine. Une première version fut proposée via cet article puis mise à jour par la suite [1]. Son domaine d'annotation varie légèrement de MEDIA et PortMEDIA, étant composé d'enregistrements audio en guichet, en conditions réelles ou scénarisés, pour la réservation de trains. TARIC-SLU est composé de plus de $2\ 000$ dialogues de 108 locuteurs et utilise un total de 60 concepts sémantiques différents. Le nombre d'heures ainsi que le nombre de mots de chacun de ses corpora sont donnés en Table 5.5.

Corpus	Nb. Heures	Nb. Mots
train	07h 30m	$58,5$ k
dev	00h 29m	$3,5$ k
test	00h 54m	$7,0$ k

TABLE 5.5 – Nombre d'heures et de mots dans TARIC-SLU en ne tenant compte que des énoncés de l'utilisateur.

Il est important de noter que le pré-apprentissage de SAMU-XLSR ne traite d'aucune donnée tunisienne. Pour autant, d'autres langues classées dans la même famille des langues sémitiques sont présentes lors des apprentissages de LaBSE [FENG et al. 2022], XLS-R [BABU et al. 2022] ou SAMU-XLSR [KHURANA et al. 2022]. Parmi elles, l'arabe, l'hébreu, l'amharique et le maltais.

L'apprentissage textuel de LaBSE comprend $97\ 929$ échantillons arabes, 692 échantillons amhariques et 872 échantillons maltais, pour $100\ 000$ alignements entre des phrases arabes et anglaises, $28\ 815$ entre des phrases amhariques et anglaises et $74\ 859$ entre des phrases maltaises et anglaises. L'apprentissage sur la parole du modèle XLS-R comprend 95 segments audio arabes, 77 segments audio hébreux, 65 segments audio amhariques et $9\ 120$ segments audio maltais. SAMU-XLSR réalise ensuite son *fine-tuning* grâce à LaBSE sur 85 segments de parole arabes et 8 segments de parole maltais supplémentaires, issus de la version 8.0 de CommonVoice.

La Table 5.6 donne les résultats obtenus en *WER*, *CER* et *CVER* sur le corpus de *test* de TARIC-SLU. Bien que les gains de performance de SAMU-XLSR face à XLS-R soient importants lors du *fine-tuning* des encodeurs de parole, nous pouvons néanmoins observer une différence majeure face au traitement de MEDIA et PortMEDIA-it lorsque ceux-ci sont figés. En effet, figer les encodeurs de parole nous montre leur capacité initiale à encoder efficacement la sémantique à partir d'un audio en langue tunisienne. Nous pouvons ici voir que XLS-R parvient bien mieux à réaliser cette tâche que SAMU-XLSR. Ceci peut être dû au fait que SAMU-XLSR n'ait pas été affiné sur autant d'échantillons de langues sémitiques que l'a été XLS-R, négligeant alors cette famille de langues.

1. https://github.com/elyadata/TARIC-SLU

Pourtant, en considérant les résultats obtenus pour un encodeur de parole *fine-tuné*, SAMU-XLSR semble avoir le bagage nécessaire pour s'adapter à un encodage sémantiquement riche pour de nouvelles langues.

		WER	*CER*	*CVER*
figé	XLS-R + SLU $_{TU}$	**58,1**	**49,1**	**71,0**
	SAMU-XLSR + SLU $_{TU}$	$63,8$	$51,3$	$74,7$
fine-tuné	XLS-R + SLU $_{TU}$	$39,6$	$34,5$	$51,1$
	SAMU-XLSR + SLU $_{TU}$	**32,3**	**30,7**	**47,4**

TABLE 5.6 – Résultats du corpus de *test* de TARIC-SLU en *WER*, *CER* et *CVER* avec l'architecture *SLU* présentée en Figure 5.2 pour l'analyse de l'enrichissement sémantique réalisé lors du pré-apprentissage de SAMU-XLSR en comparaison avec XLS-R figés et *fine-tunés* [LAPERRIÈRE, GHANNAY et al. 2024].

5.1.3 Analyse linguistique et sémantique couche-par-couche

Afin d'approfondir notre analyse de l'enrichissement sémantique de SAMU-XLSR, nous avons extrait les représentations vectorielles de chacune de ses 24 couches d'encodage et réalisé l'apprentissage de l'architecture présentée en Figure 5.1 sur l'ensemble de données françaises MEDIA [LAPERRIÈRE, PELLOIN, ROUVIER et al. 2023]. Utiliser ces représentations intermédiaires dans notre architecture *SLU* nous permet de visualiser l'évolution des capacités d'encodage de SAMU-XLSR au niveau sémantique, avec le *CER*, et linguistique, avec le *WER*. Nous le comparons ici encore à l'encodeur de parole XLS-R afin de mesurer l'impact de la méthode d'apprentissage de SAMU-XLSR sur ce denier.

Concrètement, nous avons retiré les couches hautes de chaque encodeur de parole, une à une, afin d'extraire la représentation intermédiaire nous intéressant. Les couches gardées sont alors figées ou *fine-tunées*. Comme précédemment, nous souhaitons par cette méthode observer les capacités initiales des encodeurs mais aussi leurs facultés d'adaptation à la tâche cible.

La Figure 5.3 représente l'évolution de l'encodage linguistique au sein de SAMU-XLSR et XLS-R lorsque ceux-ci sont figés ou *fine-tunés*.

Nous pouvons voir que l'encodage linguistique de SAMU-XLSR est, pour une langue que les deux encodeurs de parole ont pu apprendre en quantité suffisante, bien meilleur que celui de XLS-R à toute étape de leur encodage. Nous constatons aussi que leur *WER* minimal, indiqué par un point sur nos courbes, est atteint pour des couches plus hautes pour SAMU-XLSR que pour XLS-R. Nous pouvons supposer que le rapprochement des représentations vectorielles de SAMU-XLSR à celles de LaBSE cause l'amélioration de son encodage linguistique dans ses couches plus hautes.

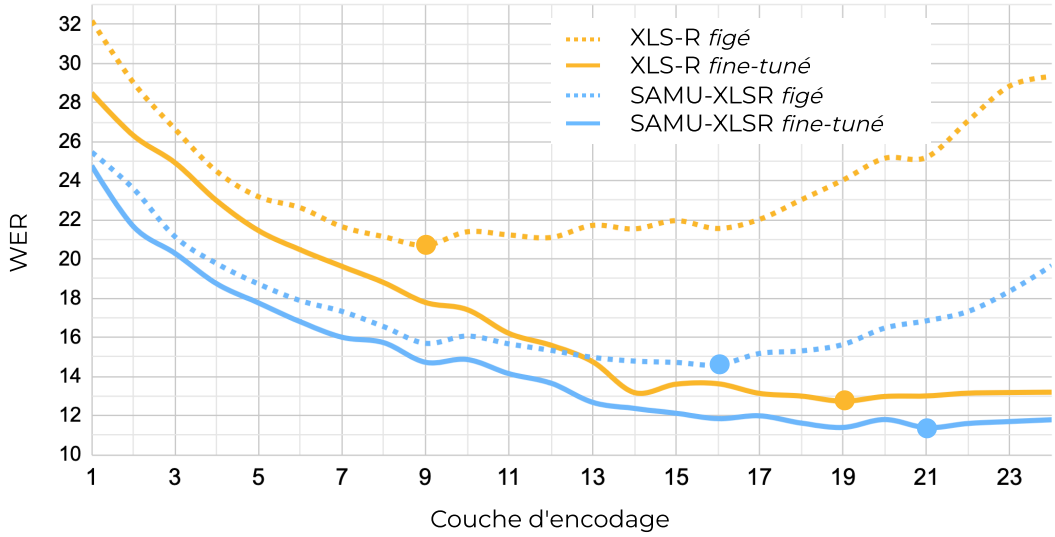

FIGURE 5.3 – Analyse couche-par-couche de l'encodage linguistique de SAMU-XLSR et XLS-R figés et *fine-tunés* suite aux *WER* obtenus sur le corpus de *test* de MEDIA pour l'apprentissage du système *SLU* présenté en Figure 5.1.

La Figure 5.4 représente l'évolution de l'encodage sémantique au sein de SAMU-XLSR et XLS-R lorsque ceux-ci sont figés ou *fine-tunés*.

Le plus intéressant ici se situe dans la perte de pertinence d'encodage pour la résolution d'une tâche d'extraction sémantique. Tandis que XLS-R perd presque 7 points de *CER* entre sa représentation vectorielle la plus pertinente sémantiquement obtenue en sortie de sa couche 15 et celle obtenue à sa dernière couche, SAMU-XLSR perd moins de 1 point de *CER* entre sa couche 14 et 24. Ceci est facilement explicable de part l'apprentissage de SAMU-XLSR, ayant pour objectif de projeter sa représentation finale vers l'espace sémantique de LaBSE. Le modèle tend donc à capturer et encoder la sémantique au mieux jusqu'à sa dernière couche.

5.2 Spécialisation sémantique

Comme dit précédemment, nous avons eu pour objectif de spécialiser l'enrichissement sémantique de SAMU-XLSR pour la résolution d'une tâche d'extraction sémantique complexe [LAPERRIÈRE, H. NGUYEN et al. 2023b ; LAPERRIÈRE, H. NGUYEN et al. 2023a]. Cette section détaille notre démarche pour une spécialisation sur les enregistrements audio et transcriptions non-annotées sémantiquement de MEDIA, PortMEDIA-it et TARIC-SLU, précédant l'apprentissage *SLU* habituel définit en Figure 5.2.

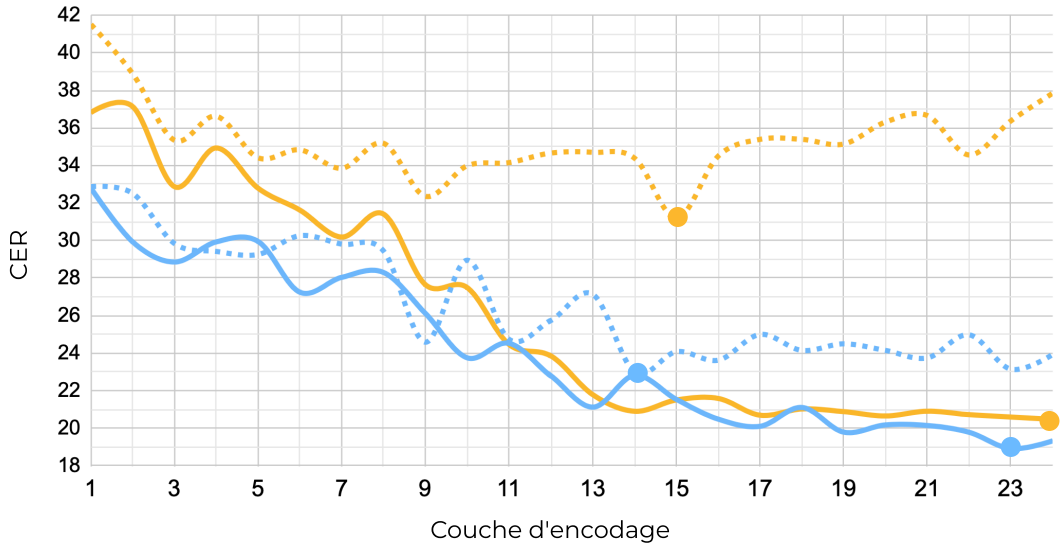

FIGURE 5.4 – Analyse couche-par-couche de l'encodage sémantique de SAMU-XLSR et XLS-R figés et *fine-tunés* suite aux *CER* obtenus sur le corpus de *test* de MEDIA pour l'apprentissage du système *SLU* présenté en Figure 5.1.

5.2.1 Architecture neuronale

L'apprentissage se fait ici en deux étapes séquentielles. Premièrement, nous réalisons un *fine-tuning*, nommé dans cette thèse «spécialisation», de SAMU-XLSR. Cette spécialisation suit exactement le mode opératoire donné dans le papier de Khurana et al. [2022] et décrit au Chapitre 3, apprenant $316,2$ M de paramètres. L'architecture y a été illustrée par la Figure 3.12. Nous donnons ainsi en entrée de ce système les enregistrements audio mais aussi les transcriptions brutes de nos ensembles de données.

Les modèles SAMU-XLSR spécialisés résultants de cet apprentissage, dont le temps d'apprentissage est indiqué entre parenthèses, seront nommés SAMU-XLSR $_{LANG}$ comme suit :

— SAMU-XLSR $_{FR}$ pour une spécialisation sur MEDIA (20h) ;
— SAMU-XLSR $_{IT}$ pour une spécialisation sur PortMEDIA-it $(13,5\text{h})$;
— SAMU-XLSR $_{TU}$ pour une spécialisation sur TARIC-SLU (16h) ;

S'en suit un apprentissage *SLU* classique tel que présenté par la Figure 5.2. Celui-ci utilisera alors une des spécialisations de SAMU-XLSR en tant qu'encodeur de parole qui pourra être figé ou *fine-tuné*.

5.2.2 Résultats expérimentaux

Cette section détaille les résultats obtenus pour une spécialisation de SAMU-XLSR suivie d'un apprentissage *SLU* pour MEDIA, PortMEDIA-it et TARIC-SLU. Les meilleurs scores *CER* obtenus précédemment seront indiqués et leur intitulé sera grisé dans chacune des tables.

Étude du français

La Table 5.7 donne les résultats obtenus en *WER*, *CER* et *CVER* sur le corpus de *test* de MEDIA pour une spécialisation de SAMU-XLSR sur cet ensemble de données.

On peut y voir une très nette amélioration des scores lorsque les poids de l'encodeur de parole sont figés. Pour autant, nous pouvons assimiler la phase de spécialisation à une sorte de *fine-tuning* sémantique, bien que non-ciblé sur des concepts sémantiques précis, expliquant une telle amélioration du *CER* et *CVER*. L'amélioration du *WER* lorsque l'encodeur de parole est figé peut s'expliquer par le fait que celui-ci ait pu se familiariser avec le domaine lexical et l'environnement d'enregistrement audio de MEDIA durant sa spécialisation.

Lors du *fine-tuning* du modèle déjà spécialisé, l'amélioration des scores n'est pas significative considérant notre mesure de confiance de $0,4$ points de *CER*. On peut déduire que l'apprentissage *SLU* apprend simplement au modèle déjà spécialisé sémantiquement quels concepts sont visés par la tâche MEDIA.

		WER	*CER*	*CVER*
figé	SAMU-XLSR + SLU $_{FR}$	$21,3$	$27,4$	$41,6$
	SAMU-XLSR $_{FR}$ + SLU $_{FR}$	**12,7**	**21,3**	**32,4**
fine-tuné	SAMU-XLSR + SLU $_{FR}$	$11,7$	$18,7$	$29,4$
	SAMU-XLSR $_{FR}$ + SLU $_{FR}$	**11,6**	**18,6**	**29,1**

TABLE 5.7 – Résultats du corpus de *test* de MEDIA en *WER*, *CER* et *CVER* avec l'architecture neuronale présentée en Figure 5.2 pour l'analyse de SAMU-XLSR $_{FR}$ figé et *fine-tuné* [LAPERRIÈRE, H. NGUYEN et al. 2023b ; LAPERRIÈRE, H. NGUYEN et al. 2023a].

Étude de l'italien

La Table 5.8 donne les résultats obtenus en *WER*, *CER* et *CVER* sur le corpus de *test* de PortMEDIA-it pour une spécialisation de SAMU-XLSR sur cet ensemble de données.

Une spécialisation purement italienne n'améliore pas les performances du système lorsque l'encodeur de parole est *fine-tuné*, sans doute pour les mêmes raisons que celles énoncées précédemment avec MEDIA. Nous rappelons que l'intervalle de confiance pour PortMEDIA-it est de $0,7$ points de *CER*. Notons tout de même l'amélioration importante des résultats lorsque l'encodeur de parole est figé durant l'apprentissage *SLU*, comme déjà constaté pour MEDIA.

		WER	*CER*	*CVER*
figé	SAMU-XLSR + SLU $_{IT}$	$31,5$	$33,6$	$49,0$
	SAMU-XLSR $_{IT}$ + SLU $_{IT}$	**19,6**	**30,3**	**42,9**
fine-tuné	SAMU-XLSR + SLU $_{IT}$	**15,4**	**26,6**	**39,2**
	SAMU-XLSR $_{IT}$ + SLU $_{IT}$	$15,7$	$26,8$	$39,5$

TABLE 5.8 – Résultats du corpus de *test* de PortMEDIA-it en *WER*, *CER* et *CVER* avec l'architecture neuronale présentée en Figure 5.2 pour l'analyse de SAMU-XLSR $_{IT}$ figé et *fine-tuné* [LAPERRIÈRE, H. NGUYEN et al. 2023b ; LAPERRIÈRE, H. NGUYEN et al. 2023a].

Étude du tunisien

La Table 5.9 donne les résultats obtenus en *WER*, *CER* et *CVER* sur le corpus de *test* de TARIC-SLU pour une spécialisation de SAMU-XLSR sur cet ensemble de données.

Nous constatons pour les expérimentations avec encodeur figé que l'absence de tunisien dans les données d'apprentissage des modèles permettant d'obtenir SAMU-XLSR est palliée par sa spécialisation sur TARIC-SLU. La spécialisation de SAMU-XLSR *fine-tuné* durant l'apprentissage *SLU* permet une légère amélioration des scores. Cette amélioration est pour autant peu pertinente considérant notre intervalle de confiance d'environ 1 point de *CER* pour ces données.

		WER	*CER*	*CVER*
figé	SAMU-XLSR + SLU $_{TU}$	$63,8$	$51,3$	$74,7$
	SAMU-XLSR $_{TU}$ + SLU $_{TU}$	**36,8**	**38,9**	**55,3**
fine-tuné	SAMU-XLSR + SLU $_{TU}$	$32,3$	$30,7$	$47,4$
	SAMU-XLSR $_{TU}$ + SLU $_{TU}$	**22,9**	**30,3**	**45,2**

TABLE 5.9 – Résultats du corpus de *test* de TARIC-SLU en *WER*, *CER* et *CVER* avec l'architecture neuronale présentée en Figure 5.2 pour l'analyse de SAMU-XLSR $_{TU}$ figé et *fine-tuné* [LAPERRIÈRE, GHANNAY et al. 2024].

5.3 Spécialisation contextuelle

SAMU-XLSR apprenant initialement à capturer la sémantique d'une phrase dans une unique représentation vectorielle pour un segment audio complet, nous nous sommes penchés vers la possibilité d'utiliser ce vecteur au *sentence-level* dans nos expérimentations *SLU*. En concaténant ce vecteur aux représentations *frame-level* plus couramment utilisées pour le traitement de tâches de reconnaissance et compréhension de la parole, nous espérons fournir plus de contexte et d'abstractions sémantiques à notre système *SLU* afin qu'il puisse utiliser au mieux les informations pertinentes disponibles aux divers niveaux de SAMU-XLSR et/ou XLS-R.

Cette section présente donc le système développé ainsi que nos résultats expérimentaux en constatant les limites d'une telle approche et discutant de perspectives possibles d'amélioration du traitement de ces représentations *sentence-level*.

5.3.1 Architecture neuronale

L'apprentissage *SLU* consiste en l'utilisation de notre architecture présentée en Figure 5.2 et des couches hautes de SAMU-XLSR réalisant le sous-échantillonage des vecteurs *frame-level*. Cette nouvelle architecture est présentée en Figure 5.5. Chaque représentation de la parole au *frame-level* (dimension 1024) sortant de SAMU-XLSR ou XLS-R est concaténée à la représentation *sentence-level* (dimension 768) correspondante de SAMU-XSLR. Cette dernière peut ou non être *fine-tunée* lors de l'apprentissage *SLU* en continuant d'adapter les paramètres des couches la générant. Les résultats présentés dans la section suivante ont été obtenus en réalisant le *fine-tuning* des couches de sous-échantillonage de SAMU-XLSR. Lorsque l'encodeur de parole est *fine-tuné*, ce système apprend un total de $394,1$ M de paramètres, contre $387,8$ M pour les modèles *SLU* précédemment étudiés.

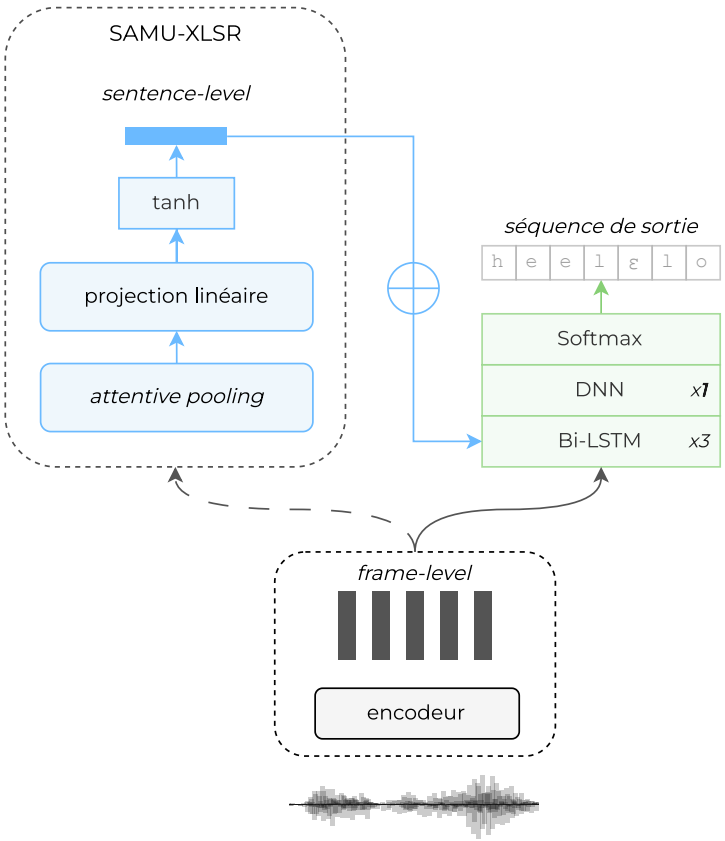

FIGURE 5.5 – Architecture neuronale *SLU* pour l'utilisation des représentations *sentence-level* de SAMU-XLSR concaténées aux représentations *frame-level* de SAMU-XLSR ou XLS-R.

5.3.2 Résultats expérimentaux

Cette section discute des résultats expérimentaux obtenus lors de l'utilisation de la représentation *sentence-level* de SAMU-XLSR de par sa concaténation aux représentations *frame-level* de SAMU-XLSR ou XLS-R correspondantes. Dans les Tables 5.10 et 5.11, nous nommons «*frame*» les systèmes utilisant uniquement les représentations *frame-level* et «*frame* $\oplus$ *sentence*» ceux utilisant une concaténation de représentations *frame-level* et *sentence-level*. Ces tables présentent les scores *CER* et *CVER* des expérimentations sur l'utilisation des représentations au *sentence-level* de SAMU-XLSR sur les ensembles de données MEDIA et PortMEDIA-it.

Nous pouvons voir que les résultats avec un encodeur de parole figé sont bien meilleurs lors de l'utilisation de ces représentations *sentence-level*, signifiant qu'elles apportent en effet des informations complémentaires pour le traitement de notre tâche *SLU*.

En revanche, lors du *fine-tuning* d'encodeurs non-spécialisés, les scores sont moins bons lorsque ces représentations sont utilisées. Cette diminution ou stagnation des performances démontre que le modèle n'arrive pas à extraire les informations pertinentes contenues dans la représentation sémantique *sentence-level* générée par SAMU-XLSR. Il est possible que les informations qui y sont encodées puissent déjà être aisément récupérables dans les représentations *frame-level* après une légère modification des paramètres de l'encodeur de parole.

Cette constatation n'est pas valable lors de l'utilisation d'un encodeur spécialisé. L'utilisation des représentations au *sentence-level* et *frame-level* de SAMU-XLSR $_{FR}$ et SAMU-XLSR $_{IT}$ *fine-tunées* durant l'apprentissage *SLU* mène à un *CER* de $18,3$ et un *CVER* de $28,4$ pour MEDIA et un *CER* de $26,3$ et un *CVER* de $38,7$ pour PortMEDIA-it, soit les meilleurs scores obtenus jusqu'à présent sur ces données.

Bien que l'amélioration de ces scores soit insuffisante pour être réellement pertinente considérant nos intervalles de confiance, le fait que les encodeurs spécialisés permettent un meilleur traitement de la représentation au *sentence-level* que ceux n'ayant jamais traité les données MEDIA ou PortMEDIA-it indique que ce vecteur pourrait être difficile à affiner directement pour une tâche *SLU*. Lorsque déjà spécialisé sur le domaine sémantique et audio de MEDIA lors d'une spécialisation de SAMU-XLSR, il serait ainsi plus simple à affiner pour la réalisation d'une tâche d'extraction de concepts sémantique.

Nous avons par la suite cherché à diminuer la dimensionnalité de cette représentation *sentence-level* grâce à des couches neuronales supplémentaires réalisant un goulet d'étranglement avant la génération des représentations finales de SAMU-XLSR. Ceci nous permit d'extraire des représentations intermédiaires de dimension 100 et 50 au lieu de 768. Malgré une similarité cosinus plus que convenables obtenue avec l'ajout de ce goulet d'étranglement, les performances de nos expérimentations *frame* $\oplus$ *sentence* avec cette réduction de dimensionnalité n'étaient pas concluantes.

		frame		*frame* ⊕ *sentence*	
		CER	*CVER*	*CER*	*CVER*
figé	XLS-R + SLU $_{FR}$	$33,8$	$46,0$	**30,9**	$43,7$
	SAMU-XLSR + SLU $_{FR}$	$27,4$	$41,6$	**26,4**	$40,6$
	SAMU-XLSR $_{FR}$ + SLU $_{FR}$	$21,3$	$32,4$	**20,6**	$31,2$
fine-tuné	XLS-R + SLU $_{FR}$	**21,8**	$32,8$	$21,9$	$32,8$
	SAMU-XLSR + SLU $_{FR}$	**18,7**	$29,4$	$19,1$	$29,8$
	SAMU-XLSR $_{FR}$ + SLU $_{FR}$	$18,6$	$29,1$	**18,3**	$28,4$

TABLE 5.10 – Résultats du corpus de *test* de MEDIA en *CER* et *CVER* pour l'analyse de l'apport sémantique de la représentation vectorielle au *sentence-level* de SAMU-XLSR.

		frame		*frame* ⊕ *sentence*	
		CER	*CVER*	*CER*	*CVER*
figé	XLS-R + SLU $_{IT}$	$42,1$	$57,1$	**40,2**	$55,9$
	SAMU-XLSR + SLU $_{IT}$	$33,6$	$49,0$	**32,5**	$49,4$
	SAMU-XLSR $_{IT}$ + SLU $_{IT}$	$30,3$	$42,9$	**29,0**	$41,8$
fine-tuné	XLS-R + SLU $_{IT}$	**29,6**	$41,5$	$29,7$	$42,1$
	SAMU-XLSR + SLU $_{IT}$	**26,6**	$39,2$	$27,6$	$39,5$
	SAMU-XLSR $_{IT}$ + SLU $_{IT}$	$26,8$	$39,5$	**26,3**	$38,7$

TABLE 5.11 – Résultats du corpus de *test* de PortMEDIA-it en *CER* et *CVER* pour l'analyse de l'apport sémantique de la représentation vectorielle au *sentence-level* de SAMU-XLSR.

5.4 Double spécialisation sémantique

Cette section présente nos expérimentations monolingues sur une double spécialisation de l'enrichissement sémantique d'un encodeur de parole [LAPERRIÈRE, GHANNAY et al. 2024]. Cette double spécialisation a surtout un intérêt dans un contexte expérimental multilingue. Nous étudierons donc son application multilingue dans le chapitre suivant. En réunissant en un unique modèle le *fine-tuning SLU* de SAMU-XLSR et sa spécialisation vers l'espace sémantique de LaBSE, nous souhaitons :

— lors de la spécialisation de l'encodeur, orienter son encodage sémantique vers le domaine sémantique précis ciblé par notre tâche *SLU* ;
— lors du *fine-tuning SLU* de l'encodeur, préserver sa faculté à générer certaines abstractions sémantiques et limiter la perte de ses capacités cross-lingues ;

Après avoir présenté cette nouvelle architecture neuronale, nous discuterons des résultats obtenus avec ce système.

5.4.1 Architecture neuronale

La Figure 5.6 présente notre architecture neuronale pour une double spécialisation sémantique de SAMU-XLSR. L'apprentissage de ce modèle de $385,6$ M de paramètres, contre $316,3$ M pour une spécialisation seule puis $387,8$ M pour l'apprentissage *SLU* habituellement nécessaire, sera réalisé une fois encore sur une carte graphique V100 de 32 Go pour 100 époques.

Les modèles résultants seront nommés SAMU-XLSR $_{LANG\ dual}$ comme suit :

— SAMU-XLSR $_{FR\ dual}$ pour une double spécialisation sur MEDIA (40h) ;
— SAMU-XLSR $_{IT\ dual}$ pour une double spécialisation sur PortMEDIA-it (25h) ;
— SAMU-XLSR $_{TU\ dual}$ pour une double spécialisation sur TARIC-SLU ($31,5$h) ;

La double spécialisation de SAMU-XLSR sera suivie d'un second apprentissage *SLU* (Figure 5.2) pour le traitement de TARIC-SLU, étant nécessaire pour l'obtention de résultats pertinents pour le traitement du tunisien. Ceci peut être expliqué par le manque de données tunisiennes dans les modules utilisés lors de l'apprentissage de SAMU-XLSR. Lorsqu'une spécialisation sur MEDIA ou PortMEDIA-it signifie un *fine-tuning* sur le français et l'italien, une spécialisation sur TARIC-SLU signifie un tout premier apprentissage sur le tunisien. Ceci justifie la nécessité de *fine-tuner* le modèle une fois de plus sur la tâche *SLU* seule afin de prendre en charge cette nouvelle langue sans utiliser LaBSE qui n'est jamais affiné pour le tunisien.

Le seul hyper-paramètre ayant été optimisé dans ce système concerne la distribution des fonctions de coût de chacun des modules se propageant lors de l'apprentissage. Nous considérons un coût global C comme suit, avec λ la valeur à optimiser :

$$C = C_{SAMU\text{-}XLSR} + \lambda\, C_{SLU} \tag{5.1}$$

De nombreux λ ont été testés, dont une quinzaine dans l'intervalle $]0;1[$ puis : 1, 2, 3, 4, 5 10, 15 et 20. Dans l'intervalle $]0;1[$, étudié en premier lors de nos expérimentations, une amélioration linéaire des scores *CER* aura été observée lorsque λ approchait 1. Nous nous sommes donc tournés vers une spécialisation de SAMU-XLSR plus impactée par la tâche *SLU* que par la tâche de rapprochement sémantique vers LaBSE et avons poursuivit l'optimisation de λ pour des valeurs plus importantes.

Pour autant, passé 1, donner encore plus d'importance à C_{SLU} qu'à $C_{SAMU\text{-}XLSR}$ ne permet pas d'amélioration distincte des scores en fin d'apprentissage. De ce fait, les expérimentations décrites dans la section suivante sont issues de systèmes utilisant différents λ compris dans $[1;20]$. Sont choisis ceux ayant obtenu les meilleurs scores sur le corpus de *dev* des ensembles de données traités.

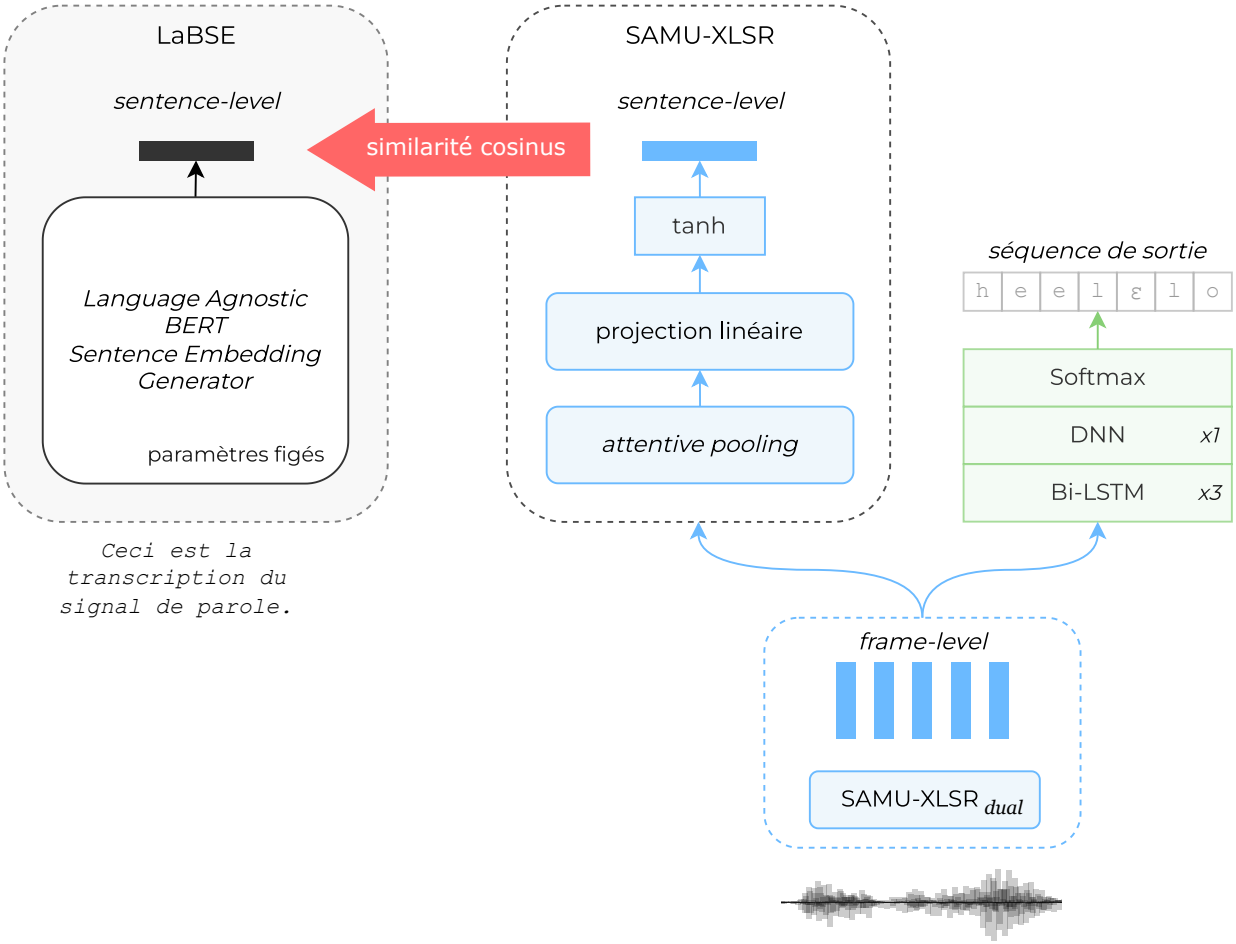

FIGURE 5.6 – Architecture neuronale pour une double spécialisation de SAMU-XLSR [LAPERRIÈRE, GHANNAY et al. 2024].

5.4.2 Résultats expérimentaux

Cette section présente les résultats expérimentaux obtenus lors d'une double spécialisation de l'encodeur de parole SAMU-XLSR pour MEDIA, PortMEDIA-it et TARIC-SLU. Les apprentissages réalisés pour la tâche TARIC-SLU seront suivis d'un second apprentissage *SLU* comme détaillé dans la Figure 5.2.

Étude du français

La Table 5.12 donne les résultats en *WER*, *CER* et *CVER* du corpus de *test* de MEDIA. Sont grisés les résultats obtenus précédemment dans cette thèse pour cet ensemble de données avec une spécialisation de SAMU-XLSR sur MEDIA suivie d'un apprentissage *SLU* sur ce même ensemble de données.

Ces scores sont légèrement dépassés par cette nouvelle architecture mêlant spécialisation et tâche *SLU* en un unique système, permettant ainsi de réduire grandement les coûts computationnels. C'est ce qui en fait son principal avantage pour des expérimentations monolingues, la diminution du *CER* étant plus petite que l'intervalle de confiance de $0,4$ points de MEDIA.

	WER	*CER*	*CVER*
SAMU-XLSR $_{FR}$ + SLU $_{FR}$	$11,6$	$18,6$	$29,1$
SAMU-XLSR $_{FR\ dual}$	**10,6**	**18,3**	**27,8**

TABLE 5.12 – Résultats du corpus de *test* de MEDIA en *WER*, *CER* et *CVER* pour l'analyse de SAMU-XLSR $_{FR\ dual}$ *fine-tuné* [LAPERRIÈRE, GHANNAY et al. 2024].

Étude de l'italien

La Table 5.13 donne les résultats en *WER*, *CER* et *CVER* du corpus de *test* de PortMEDIA-it. Sont ici aussi grisés les résultats obtenus précédemment dans cette thèse pour cet ensemble de données avec une spécialisation de SAMU-XLSR sur PortMEDIA-it suivie d'un apprentissage *SLU* sur ce même ensemble de données.

Tout comme pour MEDIA, la double spécialisation de l'encodeur de parole n'influence que légèrement les résultats, ne les dégradant pas non-plus significativement si on considère l'intervalle de confiance de $0,7$ points de *CER* de PortMEDIA-it. Le Chapitre 6 proposera de réaliser une portabilité depuis MEDIA vers PortMEDIA-it, permettant de réellement tirer parti de cette double spécialisation pour cet ensemble de données.

	WER	*CER*	*CVER*
SAMU-XLSR $_{IT}$ + SLU $_{IT}$	**15,4**	**26,6**	**39,2**
SAMU-XLSR $_{IT\ dual}$	$16,9$	$26,8$	$39,4$

TABLE 5.13 – Résultats du corpus de *test* de PortMEDIA-it en *WER*, *CER* et *CVER* pour l'analyse de SAMU-XLSR $_{IT\ dual}$ *fine-tuné* [LAPERRIÈRE, GHANNAY et al. 2024].

Étude du tunisien

La Table 5.14 donne les résultats en *WER*, *CER* et *CVER* du corpus de *test* de TARIC-SLU. Sont grisés dans la table les meilleurs résultats obtenus précédemment dans cette thèse pour cet ensemble de données avec une spécialisation de SAMU-XLSR sur TARIC-SLU suivie d'un apprentissage *SLU* sur ces mêmes données.

Comme évoqué précédemment, la double spécialisation seule ne permet pas d'obtenir de bons résultats sur le tunisien. Nous obtenons ainsi $32,4$ de *CER* contre $30,3$ pour une spécialisation

simple suivie d'un apprentissage *SLU*. Nous avons donc poursuivi nos expérimentations avec un second apprentissage purement *SLU*, comme présenté en Figure 5.2. LaBSE étant agnostique à la langue mais n'étant pas appris sur des données tunisiennes, nous pouvons supposer que le module réalisant le rapprochement de l'espace de représentation de SAMU-XLSR à celui de LaBSE empêche l'encodeur de se spécialiser précisément sur le traitement de cette langue. L'encodeur de parole aurait ainsi des difficultés à se focaliser uniquement sur le traitement du tunisien tout en gardant les apports pertinents venant de son cross-linguisme.

	WER	*CER*	*CVER*
SAMU-XLSR $_{TU}$ + SLU $_{TU}$	$22,9$	$30,3$	**45,2**
SAMU-XLSR $_{TU\ dual}$	$22,8$	$32,4$	$48,3$
SAMU-XLSR $_{TU\ dual}$ + SLU $_{TU}$	**22,6**	**29,9**	$46,8$

TABLE 5.14 – Résultats du corpus de *test* de TARIC-SLU en *WER*, *CER* et *CVER* pour l'analyse de SAMU-XLSR $_{TU\ dual}$ *fine-tuné* avec ou sans apprentissage *SLU* (Figure 5.2 supplémentaire [LAPERRIÈRE, GHANNAY et al. 2024].

5.5 Conclusion

Ce chapitre fait état de nos contributions concernant l'étude de l'enrichissement sémantique d'un encodeur de parole initialement proposé pour une tâche de recherche d'information vocale et traduction de la parole en l'appliquant à une tâche complexe visant la compréhension de la parole.

La section 5.1 permet de mettre en lumière l'utilité d'un tel encodeur pour la tâche d'extraction sémantique des ensembles de données françaises MEDIA, italiennes PortMEDIA-it et tunisiennes TARIC-SLU. En évaluant les scores de *WER*, *CER* et *CVER* obtenus sur leur corpus de *test*, nous avons pu observer l'enrichissement sémantique conséquent qu'a apporté LaBSE lors du pré-apprentissage de SAMU-XLSR.

Ce modèle ainsi que tous ses modules n'ayant jamais traité de tunisien mais seulement quelques langues sémitiques proches, nos expérimentations sur TARIC-SLU démontrent sa capacité à traiter de nouvelles langues lorsque *fine-tuné* sur celles-ci.

Une analyse poussée des représentations intermédiaires de cet encodeur de parole met en avant sa faculté à capturer efficacement la sémantique d'un segment audio dans ses couches médianes. Tandis qu'un encodeur multilingue classique tel que XLS-R détériore son encodage sémantique dans ses couches hautes, SAMU-XLSR conserve jusqu'à sa dernière couche un encodage pertinent pour la résolution d'une tâche d'extraction sémantique.

Après avoir analysé les capacités sémantiques et linguistiques obtenues via le pré-apprentissage cross-lingue de l'encodeur SAMU-XLSR, nous proposons sa spécialisation sémantique sur nos trois ensembles de données. Cette spécialisation, détaillée en section 5.2, mène à une amélioration globale de ses performances *SLU*.

C'est essentiellement le cas lorsque l'encodeur spécialisé est figé lors de l'apprentissage *SLU*, ce qui démontre sa capacité à se concentrer sur un domaine sémantique précis lors d'une spécialisation qui ne pointe pourtant pas les concepts sémantiques à extraire.

Ces expérimentations nous ont incités à spécialiser SAMU-XLSR sur d'autres langues comme le tamasheq [LAURENT et al. 2023].

La section 5.3 présente par la suite nos expérimentations sur l'utilisation de la représentation au *sentence-level* de SAMU-XLSR. Nous avons pu confirmer que celle-ci pouvait apporter des informations complémentaires pour le traitement de nos tâches d'extraction sémantique, que ce soit en contextualisant le segment de parole ou en apportant des abstractions sémantiques pertinentes.

Pour autant, nous n'avons pas réussi à tirer pleinement profit de son encodage sémantique. Nos résultats tendent à montrer que toute l'information sémantique nécessaire au traitement d'une trame de parole est déjà contenue dans sa représentation *frame-level*, ne nécessitant qu'un léger affinage afin de les extraire au mieux. Le fait que les encodeurs de parole préalablement spécialisés permettent une meilleure utilisation de ces vecteurs *sentence-level* nous laisse pourtant penser qu'ils pourraient simplement être difficiles à affiner en un unique apprentissage *SLU*. Ils nécessiteraient ainsi un premier affinage voire une réduction de dimensionnalité pertinente.

Enfin, nous présentons en section 5.4 une double spécialisation de l'enrichissement sémantique d'un encodeur de parole. À travers la fusion d'un module de spécialisation et d'extraction sémantique, nous souhaitons que les représentations sémantiques de SAMU-XLSR puissent être affinées pour correspondre au mieux à la tâche *SLU* ciblée tout en préservant sa capacité à générer certaines abstractions sémantiques cross-lingues. Préserver ces capacités cross-lingue sera tout particulièrement utile lors de nos expérimentations multilingues présentées au Chapitre 6.

Ces expérimentations de double spécialisation sémantique auront mené à de très légères améliorations de *CER* pour MEDIA et TARIC-SLU.

Nous poursuivons l'ensemble de ces expérimentations dans un contexte de portabilité cross-lingue au chapitre suivant.

Chapitre 6

Multilinguisme et compréhension de la parole

Sommaire

Publications liées à ce chapitre

- **JSALT 2022** – Multi-lingual speech to speech translation for under-resourced languages [LARCHER et al. 2022]
- **SLT 2023** – On the use of semantically-aligned speech representations for Spoken Language Understanding [LAPERRIÈRE, PELLOIN, ROUVIER et al. 2023]
- **SASB 2023** – Specialized semantic enrichment of speech representations [LAPERRIÈRE, H. NGUYEN et al. 2023b]
- **Interspeech 2023** – Semantic enrichment towards efficient speech representations [LAPERRIÈRE, H. NGUYEN et al. 2023a]
- **Interspeech 2024** – A dual task learning approach to fine-tune a multilingual semantic speech encoder for Spoken Language Understanding [LAPERRIÈRE, GHANNAY et al. 2024]

Ce chapitre fait suite au Chapitre 5 en apportant un contexte multilingue à nos analyses et expérimentations pour l'enrichissement sémantique d'un système de compréhension de la parole.

Considérant l'agnosticisme à la langue de l'encodeur de parole SAMU-XLSR [KHURANA et al. 2022], nos expérimentations cross-lingues consisteront en la spécialisation de son enrichissement sémantique sur plusieurs langues. Nos résultats seront évalués grâce aux métriques *WER*, *CER* et *CVER* décrites au Chapitre 4 et comparés aux résultats grisés dans nos tables, obtenus sur de précédents apprentissages. Comme au chapitre précédent, nous analyserons les représentations intermédiaires de nos encodeurs de parole spécialisés en les extrayant couche-par-couche.

Nous réaliserons par ailleurs des expérimentations concernant la portabilité de systèmes de compréhension de la parole d'une langue à une autre avant d'analyser en Section 6.2 leur portabilité entre différents domaines.

Cette thèse étudie le transfert de connaissances cross-lingues de diverses architectures et méthodes d'apprentissages neuronales afin de viser la compréhension de la parole pour des ensembles de données peu volumineux. Nous souhaitons ainsi tirer bénéfice de plus larges ensembles de données pour réaliser une extraction sémantique complexe de ces derniers.

Pour ce faire, nous considérons des langues proches comme le français avec l'ensemble de données MEDIA [BONNEAU-MAYNARD, ROSSET et al. 2005] et l'italien avec l'ensemble de données PortMEDIA-it [LEFÈVRE, MOSTEFA et al. 2012], tous deux présentés au Chapitre 4, et une langue plus distante telle que le tunisien avec l'ensemble de données TARIC-SLU [MDHAFFAR, BOUGARES et al. 2024], introduite au chapitre précédent. Nous considérons ici MEDIA comme étant notre ensemble de données le plus large, bien que lui-même déjà relativement petit. Nous souhaitons ainsi tirer parti de données françaises pour l'extraction sémantique de données italiennes et tunisiennes, mais n'excluons pas le possible apport de l'ensemble de ces langues sur chacune d'entre elles.

Notons que 809 des $2\ 549$ groupes de mots-support encadrés par des concepts sémantiques dans TARIC-SLU contiennent de l'alternance codique (*code-switching*) sur au moins un de leurs mots, autrement dit des mots prononcés en une autre langue que le tunisien. Cette alternance est, selon Mdhaffar et al. [2024] supposément à 80% française et 20% anglaise. Malgré l'apparition de mots français, nous pouvons tout de même estimer que le français et tunisien restent des langues distantes, ce qui est d'autant plus le cas pour le tunisien et l'italien.

6.1 Apprentissage cross-lingue

Suite aux spécialisations monolingues de l'encodeur de parole SAMU-XLSR présentées en Section 5.2 du Chapitre 5, nous présentons en Section 6.1.1 l'apport du multilinguisme pour l'extraction sémantique d'ensembles de données peu dotés à travers une spécialisation cross-lingue de SAMU-XLSR. Nous y analyserons l'apport d'une telle spécialisation dans ses représentations intermédiaires via leur extraction couche-par-couche.

Nous étudierons la portabilité sémantique et linguistique de nos systèmes de compréhension de la parole depuis le français vers l'italien et constaterons son impact lorsque jointe à l'utilisation de la représentation au *sentence-level* de SAMU-XLSR.

Nous finirons par poursuivre en Section 6.1.2 nos expérimentations menées en Section 5.4 du Chapitre 5 à travers une double spécialisation cross-lingue de SAMU-XLSR, menant à ce jour à des performances à l'état-de-l'art pour TARIC-SLU et pour la version *full* de MEDIA avec un modèle de bout-en-bout.

6.1.1 Lors d'une spécialisation sémantique

Nous détaillons dans cette section nos résultats expérimentaux suite à une spécialisation sémantique cross-lingue de SAMU-XLSR pour le traitement du français, de l'italien et du tunisien, avec les ensembles de données respectifs MEDIA, PortMEDIA-it et TARIC-SLU [LAPERRIÈRE, PELLOIN, ROUVIER et al. 2023 ; LAPERRIÈRE, H. NGUYEN et al. 2023b ; LAPERRIÈRE, H. NGUYEN et al. 2023a]. Nous analysons par la suite le gain d'une telle spécialisation en extrayant les représentations vectorielles en sortie de chaque couche de divers encodeurs de parole.

L'architecture neuronale utilisée lors de la spécialisation de SAMU-XLSR mais aussi lors de l'apprentissage *SLU* qui l'utilise sont telles que décrites en Section 5.2.1. Les modèles SAMU-XLSR spécialisés seront nommés SAMU-XLSR $_{LANG}$ comme suit :

— SAMU-XLSR $_{FR\oplus IT}$ pour une spécialisation sur MEDIA et PortMEDIA-it conjointe $(33, 5\text{h})$;
— SAMU-XLSR $_{FR\oplus IT\oplus TU}$ pour une spécialisation sur MEDIA, PortMEDIA-it et TARIC-SLU conjointe $(49, 5\text{h})$;

Afin de fournir à nos modèles plusieurs ensembles de données distincts, nous avons analysé l'impact de leur ordonnancement et répartition. Parmi nos analyses, nous avons tenté de mélanger aléatoirement chaque échantillon mais aussi des sous-ensembles d'échantillons monolingues. Nous avons étudié l'agencement des langues au sein d'une époque, réalisant en premier le traitement de données françaises puis italiennes, ou inversement. Enfin, nous avons tenté l'apprentissage de 20, 30 ou 50 époques purement monolingues avant de réaliser le reste de nos 100 époques habituelles de manière multilingue. Nous avons ainsi constaté que l'impact de l'ordre d'apprentissage de nos données multilingues était minime pour une quantité d'heures de parole aussi faible, la variation de résultats *CER* ne dépassant pas la marge d'erreur de $0,4$ points. Les ensembles de données utilisés lors de nos expérimentations multilingues seront donc fournis au modèle l'un à la suite de l'autre dans une même époque d'apprentissage, du plus fourni au moins doté.

Pour le traitement du français

La Table 6.1 présente les scores de *WER*, *CER* et *CVER* du corpus de *test* de MEDIA pour une spécialisation de SAMU-XLSR sur MEDIA et PortMEDIA-it.

Ceux-ci sont légèrement meilleurs que les scores obtenus avec une spécialisation monolingue française sur MEDIA. La différence étant globalement inférieure à $0,4$ points, on estime que ce gain de performances grâce aux données italiennes de PortMEDIA-it n'est pas réellement pertinent. Cette amélioration timide des résultats peut être due au fait que MEDIA ait quatre fois plus de données textuelles que PortMEDIA-it dans son corpus d'apprentissage. Celui-ci étant considérablement plus petit, son impact est donc minime. La quantité de données françaises utilisés pourrait de plus ne pas permettre un gain conséquent de performances à partir de l'ajout d'une faible quantité de données italiennes.

Pour autant, une amélioration persiste de par l'apport de données supplémentaires du même domaine sémantique, d'une langue proche et d'environnement audio similaire.

		WER	*CER*	*CVER*
figé	SAMU-XLSR $_{FR}$ + SLU $_{FR}$	$12,7$	$21,3$	$32,4$
	SAMU-XLSR $_{FR \oplus IT}$ + SLU $_{FR}$	**12,5**	**20,8**	**31,6**
fine-tuné	SAMU-XLSR $_{FR}$ + SLU $_{FR}$	$11,6$	$18,6$	**29,1**
	SAMU-XLSR $_{FR \oplus IT}$ + SLU $_{FR}$	**11,5**	**18,5**	**29,1**

TABLE 6.1 – Résultats du corpus de *test* de MEDIA en *WER*, *CER* et *CVER* avec l'architecture neuronale présentée en Figure 5.2 pour l'analyse des spécialisations multilingues de SAMU-XLSR figées et *fine-tunées* [LAPERRIÈRE, H. NGUYEN et al. 2023b ; LAPERRIÈRE, H. NGUYEN et al. 2023a].

Pour le traitement de l'italien

La Table 6.2 présente les scores de *WER*, *CER* et *CVER* du corpus de *test* de PortMEDIA-it pour une spécialisation de SAMU-XLSR sur MEDIA et PortMEDIA-it.

Contrairement aux résultats obtenus sur l'ensemble de données MEDIA, PortMEDIA-it bénéficie grandement d'une spécialisation de SAMU-XLSR commune à MEDIA. Ceci peut être expliqué par l'agnosticisme à la langue de SAMU-XLSR, tirant bénéfice de la quantité de données conséquente apportée par MEDIA pour une langue pourtant différente.

Nous avons pu constater dans le chapitre précédent que lors du *fine-tuning* de SAMU-XLSR, sa spécialisation monolingue italienne peinait à dépasser les performances du modèle originel. Ce n'est plus le cas ici, la réalisation d'une spécialisation cross-lingue permettant de dépasser de $1,0$ point de *CER* ce dernier qui obtenait de meilleurs résultats pour PortMEDIA-it. Cette amélioration est significative considérant son intervalle de confiance défini à $0,7$ points de *CER*.

		WER	*CER*	*CVER*
figé	SAMU-XLSR $_{IT}$ + SLU $_{IT}$	$19,6$	$30,3$	$42,9$
	SAMU-XLSR $_{FR \oplus IT}$ + SLU $_{IT}$	**18,7**	**29,4**	**41,6**
fine-tuné	SAMU-XLSR + SLU $_{IT}$	$15,4$	$26,6$	$39,2$
	SAMU-XLSR $_{FR \oplus IT}$ + SLU $_{IT}$	**14,5**	**25,6**	**37,6**

TABLE 6.2 – Résultats du corpus de *test* de PortMEDIA-it en *WER*, *CER* et *CVER* avec l'architecture neuronale présentée en Figure 5.2 pour l'analyse des spécialisations multilingues de SAMU-XLSR figées et *fine-tunées* [LAPERRIÈRE, H. NGUYEN et al. 2023b ; LAPERRIÈRE, H. NGUYEN et al. 2023a].

Dans l'optique de tirer au mieux bénéfice des données françaises MEDIA pour le traitement des données italiennes PortMEDIA-it, nous avons étudié le portage cross-lingue de notre système *SLU* entre ces deux ensembles de données.

L'objectif ici est de permettre au système *SLU* d'apprendre à résoudre précisément la tâche d'extraction de concepts sémantiques grâce à des données annotées avec le même lexique sémantique, avant d'affiner son apprentissage sur les données cibles d'une langue proche.

Nous réalisons 100 premières époques pour un apprentissage *SLU* sur MEDIA puis transférons les connaissances du modèle avec 100 époques complémentaires sur PortMEDIA-it. Nous nommons cet apprentissage SLU $_{FR \rightarrow IT}$ afin de rester cohérents avec nos publications, bien qu'équivalant à SLU $_{FR}$ + SLU $_{IT}$. Les encodeurs de parole utilisés pour ces expérimentations sont ceux ayant mené aux meilleurs résultats pour MEDIA, étant notre première étape d'apprentissage *SLU*.

La Table 6.3 donne les résultats en *CER*, *WER* et *CVER* pour ces expérimentations de portabilité cross-lingue du français à l'italien.

On peut y voir que l'utilisation d'un encodeur de parole spécialisé sur les données italiennes PortMEDIA-it et françaises MEDIA améliore grandement les résultats lorsque celui-ci est figé durant l'apprentissage *SLU* ($26,5$ de *CER*). Les performances sont dégradées lorsque SAMU-XLSR n'aura jamais pu être affiné sur PortMEDIA-it, que ce soit lors d'une spécialisation ou d'un *fine-tuning SLU* ($30,8$ et $31,0$ de *CER*).

Lorsque l'on *fine-tune* l'encodeur de parole, la différence quant à ses performances d'extraction sémantique est bien moins flagrante, SAMU-XLSR $_{FR \oplus IT}$ menant tout de même au meilleur *CER* obtenu jusqu'alors pour des expérimentations multilingues. Cette amélioration n'est pas significative, étant plus faible que l'intervalle de confiance de $0,7$ points de *CER* de PortMEDIA-it.

La baisse de performances linguistiques liées au *WER* et *CVER* peut être expliquée par un surapprentissage du français pour notre système *SLU*. Sa fonction de coût prenant en considération les transcriptions générées par le modèle, réaliser un *fine-tuning* de SAMU-XLSR sur le français provoquerait une baisse de ses capacités cross-lingues permettant initialement de traiter des données italiennes. Nous avons ainsi observé dans les prédictions du système SAMU-XLSR $_{FR \oplus IT}$ + SLU $_{FR \rightarrow IT}$ un étiquetage sémantique correct malgré des transcriptions mêlant italien et français dans leurs mots-support. Ce gain de performances sémantiques et cette perte de performances linguistiques pourraient être étudiés à des fins de traduction automatique via la génération d'une représentation linguistique commune à deux langues proches telles que le français et l'italien.

		WER	*CER*	*CVER*
figé	SAMU-XLSR $_{FR \oplus IT}$ + SLU $_{IT}$	$18,7$	$29,4$	$41,6$
	SAMU-XLSR + SLU $_{FR \rightarrow IT}$	$32,0$	$30,8$	$48,7$
	SAMU-XLSR $_{FR}$ + SLU $_{FR \rightarrow IT}$	$31,4$	$31,0$	$49,4$
	SAMU-XLSR $_{FR \oplus IT}$ + SLU $_{FR \rightarrow IT}$	**17,3**	**26,5**	**39,2**
fine-tuné	SAMU-XLSR $_{FR \oplus IT}$ + SLU $_{IT}$	**14,5**	$25,6$	**37,6**
	SAMU-XLSR + SLU $_{FR \rightarrow IT}$	$17,5$	$25,5$	$38,4$
	SAMU-XLSR $_{FR}$ + SLU $_{FR \rightarrow IT}$	$17,8$	$25,2$	$39,1$
	SAMU-XLSR $_{FR \oplus IT}$ + SLU $_{FR \rightarrow IT}$	$16,4$	**25,1**	$38,1$

TABLE 6.3 – Résultats du corpus de *test* de PortMEDIA-it en *WER*, *CER* et *CVER* pour les expérimentations de portabilité cross-lingue depuis le français vers l'italien, avec XLS-R, SAMU-XLSR et ses spécialisations figés et *fine-tunés* [LAPERRIÈRE, H. NGUYEN et al. 2023b ; LAPERRIÈRE, H. NGUYEN et al. 2023a].

Nous avons par la suite continué nos expérimentations sur la portabilité cross-lingue de notre système *SLU* en l'appliquant aux spécialisations contextuelles réalisées au chapitre précédent.

L'architecture utilisée a été décrite en Figure 5.5. Nous effectuons ici une première spécialisation cross-lingue de SAMU-XLSR afin d'utiliser cet encodeur pour nos expériences concaténant

ses représentations *frame-level* et *sentence-level* au sein d'un unique système *SLU*.

Nous reprenons la notation «*frame*» pour un système *SLU* utilisant les représentations *frame-level* de SAMU-XLSR et la notation «*frame* $\oplus$ *sentence*» pour un système *SLU* utilisant une concaténation des représentations *sentence-level* à leurs représentations *frame-level* correspondantes.

La Table 6.4 donne les scores de *WER*, *CER* et *CVER* pour le corpus de *test* de PortMEDIA-it.

Lorsque l'encodeur de parole est figé, les performances sont généralement bien meilleures pour les systèmes *frame* $\oplus$ *sentence*, démontrant ici encore l'apport de la représentation *sentence-level* de SAMU-XLSR. Une spécialisation $FR \oplus IT$ suivie d'un apprentissage SLU $_{FR \to IT}$ donne néanmoins des scores équivalents lors de l'utilisation ou non de la représentation au *sentence-level* de SAMU-XLSR.

Lorsqu'un encodeur de parole non-spécialisé est *fine-tuné*, le modèle *SLU* peine à utiliser les représentations *sentence-level* pour améliorer ses performances. Lorsqu'un encodeur de parole spécialisé est *fine-tuné*, il y parvient, menant à un *CER* de $24,2$ pour le corpus de *test* de PortMEDIA-it, soit le meilleur obtenu jusqu'à présent pour des expérimentations monolingues et multilingues confondues. Nous émettons les mêmes hypothèses que dans la Section 5.3 du Chapitre 5. Ces comportements pourraient être dus au fait que les représentations au *sentence-level* soient difficilement affinables en un seul apprentissage. Spécialisées en amont sur nos données cibles, elles seraient bien plus à même d'être *fine-tunées* lors de l'apprentissage *SLU*, permettant de condenser au mieux les informations pertinentes pour le domaine sémantique visé.

		frame		*frame* $\oplus$ *sentence*	
		CER	*CVER*	*CER*	*CVER*
figé	SAMU-XLSR $_{FR \oplus IT}$ + SLU $_{IT}$	$29,4$	$41,6$	**28,9**	$41,8$
	XLS-R + SLU $_{FR \to IT}$	$34,9$	$49,0$	**33,2**	$47,8$
	SAMU-XLSR + SLU $_{FR \to IT}$	$30,8$	$48,7$	**30,4**	$47,3$
	SAMU-XLSR $_{FR}$ + SLU $_{FR \to IT}$	$31,0$	$49,4$	**29,0**	$41,6$
	SAMU-XLSR $_{FR \oplus IT}$ + SLU $_{FR \to IT}$	$26,5$	$39,2$	$26,5$	$39,1$
fine-tuné	SAMU-XLSR $_{FR \oplus IT}$ + SLU $_{IT}$	**25,6**	$37,6$	$25,8$	$37,9$
	XLS-R + SLU $_{FR \to IT}$	**27,3**	$41,1$	$28,0$	$42,8$
	SAMU-XLSR + SLU $_{FR \to IT}$	**25,5**	$38,4$	$26,0$	$39,4$
	SAMU-XLSR $_{FR}$ + SLU $_{FR \to IT}$	$25,2$	$39,1$	**24,2**	$37,4$
	SAMU-XLSR $_{FR \oplus IT}$ + SLU $_{FR \to IT}$	$25,1$	$38,1$	**24,4**	$37,4$

TABLE 6.4 – Résultats du corpus de *test* de PortMEDIA-it en *CER* et *CVER* pour les expérimentations de portabilité cross-lingue depuis le français vers l'italien et l'utilisation de la représentation vectorielle au *sentence-level* de SAMU-XLSR.

Pour le traitement du tunisien

La Table 6.5 présente les scores de *WER*, *CER* et *CVER* du corpus de *test* de TARIC-SLU pour des spécialisations de SAMU-XLSR sur MEDIA, PortMEDIA-it et TARIC-SLU. Nous nous sommes ici essentiellement focalisés sur les expérimentations avec un encodeur de parole *fine-tuné*, ayant déjà analysé en profondeur les capacités sémantiques de SAMU-XLSR via MEDIA et PortMEDIA-it, et ayant manqué de temps suite à la distribution tardive de TARIC-SLU par rapport à l'état d'avancement de cette thèse.

Bien que nous puissions constater une légère amélioration du *CER* lorsque l'encodeur de parole est figé durant l'apprentissage *SLU* grâce à son agnosticisme à la langue et son pré-apprentissage sur des langues sémitiques, poursuivre son *fine-tuning* résulte étonnamment en une augmentation nette de toutes nos métriques. Ceci nous encouragera par la suite à trouver d'autres moyens d'améliorer le traitement de TARIC-SLU en bénéficiant au mieux d'un apprentissage sur les données MEDIA et PortMEDIA-it [LAPERRIÈRE, GHANNAY et al. 2024].

		WER	*CER*	*CVER*
figé	SAMU-XLSR $_{TU}$ + SLU $_{TU}$	**36,8**	$38,9$	**55,3**
	SAMU-XLSR $_{FR\oplus IT\oplus TU}$ + SLU $_{TU}$	$37,1$	**37,4**	$56,0$
fine-tuné	SAMU-XLSR $_{TU}$ + SLU $_{TU}$	**22,9**	**30,3**	**45,2**
	SAMU-XLSR $_{FR}$ + SLU $_{TU}$	$24,1$	$31,0$	$46,8$
	SAMU-XLSR $_{IT}$ + SLU $_{TU}$	$23,2$	$31,2$	$46,2$
	SAMU-XLSR $_{FR\oplus IT}$ + SLU $_{TU}$	$24,4$	$30,8$	$47,3$
	SAMU-XLSR $_{FR\oplus IT\oplus TU}$ + SLU $_{TU}$	$23,1$	$31,9$	$47,0$

TABLE 6.5 – Résultats du corpus de *test* de TARIC-SLU en *WER*, *CER* et *CVER* avec l'architecture neuronale présentée en Figure 5.2 pour l'analyse des spécialisations multilingues de SAMU-XLSR figées et *fine-tunées* [LAPERRIÈRE, H. NGUYEN et al. 2023b ; LAPERRIÈRE, H. NGUYEN et al. 2023a].

Analyse linguistique et sémantique couche-par-couche

Comme nous l'avons fait pour comparer l'encodage linguistique et sémantique de SAMU-XLSR et XLS-R en Section 5.1.3 du Chapitre 5, nous présentons dans cette section une analyse couche-par-couche de deux spécialisations de SAMU-XLSR pour le traitement de la tâche PortMEDIA-it. Les Figures 6.1 et 6.2 présentent notre analyse de l'encodage des modèles SAMU-XLSR $_{IT}$ et SAMU-XLSR $_{FR\oplus IT}$, dont les résultats expérimentaux pour PortMEDIA-it ont été discutés précédemment avec les Tables 5.8 et 6.2. Ils y sont comparés à SAMU-XLSR.

Comme nous l'avions constaté, réaliser un *fine-tuning* des modèles préalablement spécialisés

n'améliore pas grandement les scores de *WER* et *CER*. Il n'y a donc pas de surprise à ce que les courbes aux traits pleins aient une évolution relativement similaire.

En revanche, un point important de ces analyses se situe au niveau de l'amélioration de l'encodage de la linguistique et de la sémantique dans les couches intermédiaires des SAMU-XLSR spécialisés figés durant l'apprentissage *SLU*. En terme de *WER*, la représentation vectorielle de la parole issue de la 17ème couche de SAMU-XLSR $_{FR\oplus IT}$ figé obtient un score de $15,3$, identique au $15,4$ de *WER* obtenu grâce à celle issue de la 24ème couche de SAMU-XLSR *fine-tuné*. Ces résultats ont pu être obtenus en spécialisant SAMU-XLSR sur une tâche ne visant ni la reconnaissance de la parole ni sa compréhension, mais bien l'injection de sémantique dans ses représentations vectorielles.

SAMU-XLSR $_{FR\oplus IT}$ figé obtient grâce à cette 17ème couche un *CER* de $28,7$, moins bon que les $26,6$ points du modèle *fine-tuné* non-spécialisé. Nous avons poussé son analyse en l'utilisant pour la portabilité de notre système *SLU* depuis le français vers l'italien. Le système SAMU-XLSR $_{IT\oplus FR\ (17)}$ + SLU $_{FR\rightarrow IT}$ obtient $15,9$ de *WER*, $25,6$ de *CER* et $38,0$ de *CVER*, soit un *CER* de seulement $0,5$ points de plus que celui obtenu avec SAMU-XLSR $_{IT\oplus FR}$ + SLU $_{FR\rightarrow IT}$ en réalisant le *fine-tuning* de l'encodeur de parole.

Rappelons que *fine-tuner* SAMU-XLSR implique une augmentation du nombre de paramètres à apprendre, passant de $71,5$ M à $387,8$ M. Outre ces performances intéressantes, ce modèle est donc tout particulièrement intéressant du point de vue de sa taille.

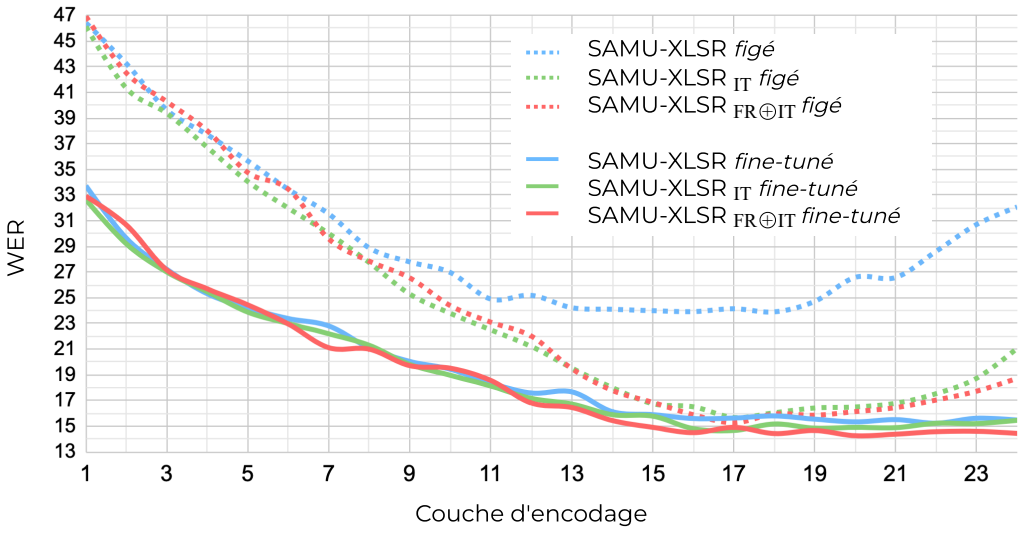

FIGURE 6.1 – Analyse couche-par-couche de l'encodage linguistique de SAMU-XLSR, SAMU-XLSR $_{IT}$ et SAMU-XLSR $_{FR\oplus IT}$ figés et *fine-tunés* suite aux *WER* obtenus sur le corpus de *test* de PortMEDIA-it pour l'apprentissage du système *SLU* présenté en Figure 5.2 [LAPERRIÈRE, H. NGUYEN et al. 2023a].

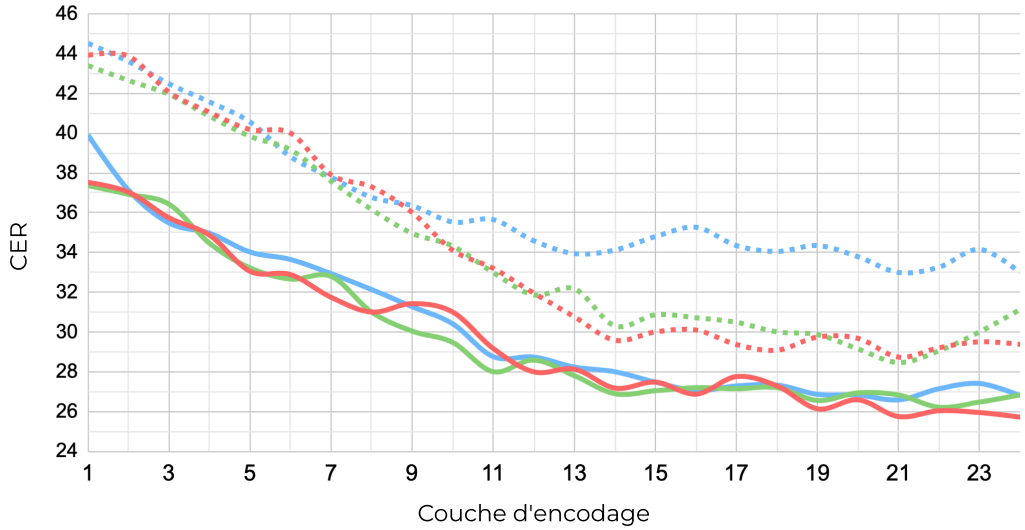

FIGURE 6.2 – Analyse couche-par-couche de l'encodage sémantique de SAMU-XLSR, SAMU-XLSR $_{IT}$ et SAMU-XLSR $_{FR \oplus IT}$ figés et *fine-tunés* suite aux *CER* obtenus sur le corpus de *test* de PortMEDIA-it pour l'apprentissage du système *SLU* présenté en Figure 5.2 [LAPERRIÈRE, H. NGUYEN et al. 2023a].

6.1.2 Lors d'une double spécialisation

Cette section présente nos expérimentations d'une double spécialisation sémantique cross-lingue de SAMU-XLSR [LAPERRIÈRE, GHANNAY et al. 2024], dont l'architecture est présentée en Section 5.4.1 du Chapitre 5. Le coefficient λ lié à la fonction de coût du module *SLU* suit la même méthode d'optimisation, étant choisi indépendamment pour chaque apprentissage en fonction du *CER* obtenu sur le corpus de *dev* traité.

Les trois mêmes ensembles de données seront traités, à savoir MEDIA, PortMEDIA-it et TARIC-SLU. Les modèles résultants seront nommés SAMU-XLSR $_{LANG\ dual}$ comme suit :

— SAMU-XLSR $_{FR \oplus IT\ dual}$ pour une double spécialisation sur MEDIA et PortMEDIA-it (49h) ;
— SAMU-XLSR $_{FR \oplus IT \oplus TU\ dual}$ pour une double spécialisation sur MEDIA, PortMEDIA-it et TARIC-SLU (79h) ;

Pour le traitement du français

La Table 6.6 présente les scores de *WER*, *CER* et *CVER* pour le corpus de *test* de l'ensemble de données MEDIA suite à une double spécialisation sémantique cross-lingue de l'encodeur de parole SAMU-XLSR.

Les scores obtenus pour une double spécialisation monolingue française sont nettement dépassés par l'utilisation conjointe des données italiennes PortMEDIA-it. En plus de réduire grandement le nombre de paramètres à apprendre, cette méthode permet ainsi l'obtention de résultats à l'état-de-l'art pour la version *full* de l'ensemble de données MEDIA avec une architecture de bout-en-bout, menant à un *CER* de $17,9$ et un *CVER* de $28,2$. Ces résultats sont significatifs considérant l'intervalle de confiance de $0,4$ points de *CER* défini pour MEDIA.

À notre connaissance, les meilleurs résultats obtenus pour la version *relax* de MEDIA sont actuellement de $10,8$ de *CER* et $17,0$ de *CVER* avec des systèmes en cascade [PELLOIN 2024] et de $12,1$ de *CER* et $14,7$ de *CVER* avec des systèmes de bout-en-bout [DENISOV et VU 2023]. Ces deux scores *CVER* sont issus d'une normalisation par règles humaines des mots-support en valeurs et sont donc très loin d'être comparables à nos résultats.

À des fins de comparaison, notre système SAMU-XLSR $_{FR \oplus IT\ dual}$ obtient un *CER* de $15,3$ lors d'une inférence réalisée sur le corpus de *test* de la version *relax* de MEDIA.

	WER	*CER*	*CVER*
SAMU-XLSR $_{FR\ dual}$	**10,6**	$18,3$	**27,8**
SAMU-XLSR $_{FR \oplus IT\ dual}$	**10,6**	**17,9**	$28,2$

TABLE 6.6 – Résultats du corpus de *test* de MEDIA en *WER*, *CER* et *CVER* pour l'analyse d'une double spécialisation multilingue de SAMU-XLSR [LAPERRIÈRE, GHANNAY et al. 2024].

Pour le traitement de l'italien

La Table 6.6 présente les scores de *WER*, *CER* et *CVER* pour le corpus de *test* de l'ensemble de données PortMEDIA-it suite à une double spécialisation sémantique cross-lingue de l'encodeur de parole SAMU-XLSR.

Tout comme pour MEDIA, la double spécialisation de l'encodeur de parole permet d'orienter la spécialisation de SAMU-XLSR vers l'espace sémantique souhaité pour la tâche de PortMEDIA-it. L'utilisation de données additionnelles françaises mène à un *CER* de $24,1$ dont l'amélioration est hautement significative considérant l'intervalle de confiance de $0,7$ points défini pour PortMEDIA-it.

Notons que ce score a été largement dépassé par Denisov et Vu [DENISOV et VU 2023] qui obtiennent un *CER* de $21,9$ pour PortMEDIA-it. Pour autant, nous pouvons insister sur la différence computationnelle de leur modèle et du système que nous étudions. Leur système optimise $1,16$ milliards de paramètres contre $385,6$ millions pour cet apprentissage. Bien que n'ayant pas l'information concernant le nombre d'heures d'apprentissage de leur système, nous pouvons noter qu'ils réalisent un pré-apprentissage sur $1\ 000$ heures de données multilingues en $25\ 000$ époques (*warmup steps*). Il faut de notre côté prendre en compte le premier pré-apprentissage de SAMU-XLSR. Sans ce pré-apprentissage, leur système obtient un *CER* de $25,1$.

	WER	*CER*	*CVER*
SAMU-XLSR $_{FR \oplus IT}$ + SLU $_{IT}$	**14,5**	$25,6$	**37,6**
SAMU-XLSR $_{IT\ dual}$	$16,9$	$26,8$	$39,4$
SAMU-XLSR $_{FR \oplus IT\ dual}$	$19,7$	**24,1**	$39,0$

TABLE 6.7 – Résultats du corpus de *test* de PortMEDIA-it en *WER*, *CER* et *CVER* pour l'analyse d'une double spécialisation multilingue de SAMU-XLSR [LAPERRIÈRE, GHANNAY et al. 2024].

Pour le traitement du tunisien

La Table 6.6 présente les scores de *WER*, *CER* et *CVER* pour le corpus de *test* de l'ensemble de données TARIC-SLU suite à une double spécialisation sémantique cross-lingue de l'encodeur de parole SAMU-XLSR.

Nous pouvons observer que les doubles spécialisations sur MEDIA et PortMEDIA-it seules, sans TARIC-SLU, ne permettent pas d'obtenir de bons scores de *CER*, étant tous au-dessus de 30. Ils obtiennent d'autant plus mauvais résultats concernant les scores de *WER*. Cela peut s'expliquer par le fait que SAMU-XLSR focalise alors sa spécialisation sur des langues trop distantes, perdant de son cross-linguisme qui ne permettait déjà pas de traiter efficacement la langue tunisienne.

Lors de l'ajout de TARIC-SLU dans les données utilisées pour la double spécialisation de SAMU-XLSR, le *CER* s'en voit amélioré. Une double spécialisation sur nos trois ensembles de données permet d'obtenir un *CER* à l'état-de-l'art de $29,1$ sur le corpus de *test* de TARIC-SLU, contre $29,9$ lorsque MEDIA et PortMEDIA-it ne sont pas considérés. Cette amélioration de *CER* peut être considérée significative si l'on compare nos résultats aux $30,3$ points de *CER* obtenus suite à une spécialisation monolingue de SAMU-XLSR disjointe à l'apprentissage *SLU*.

Nous constatons donc ici que l'utilisation combinée de données d'un domaine sémantique proche mais de langues distantes lors de cette double spécialisation permet l'obtention de meilleurs résultats pour une tâche d'extraction sémantique d'un ensemble de données de petite taille.

	WER	*CER*	*CVER*
SAMU-XLSR $_{TU\ dual}$ + SLU $_{TU}$	**22,6**	$29,9$	$46,8$
SAMU-XLSR $_{FR\ dual}$ + SLU $_{TU}$	$23,6$	$30,3$	$46,4$
SAMU-XLSR $_{IT\ dual}$ + SLU $_{TU}$	$23,1$	$30,8$	$46,4$
SAMU-XLSR $_{FR \oplus IT\ dual}$ + SLU $_{TU}$	$24,4$	$30,4$	$48,4$
SAMU-XLSR $_{FR \oplus IT \oplus TU\ dual}$ + SLU $_{TU}$	$22,8$	**29,1**	**46,2**

TABLE 6.8 – Résultats du corpus de *test* de TARIC-SLU en *WER*, *CER* et *CVER* pour l'analyse des doubles spécialisations multilingues de SAMU-XLSR *fine-tuné* lors des apprentissages *SLU* (Figure 5.2) supplémentaires [LAPERRIÈRE, GHANNAY et al. 2024].

6.2 Portabilité cross-domaine

Cette section présente une analyse de l'impact de la spécialisation de SAMU-XLSR sur le traitement d'ensembles de données de domaines plus ou moins distants à celui de MEDIA et PortMEDIA, étudié dans cette thèse [LAPERRIÈRE, H. NGUYEN et al. 2023a]. Les expérimentations réalisées consisteront en de simples inférences de nos modèles *SLU*. Les modèles sélectionnés seront parmi ceux affinés pour la même langue que celle de la tâche cross-domaine visée.

Par cette analyse, nous souhaitons premièrement attester d'un gain ou d'une perte des facultés d'encodage de SAMU-XLSR pour la résolution de tâches apprises lors de son premier pré-apprentissage. Pour cela, nous considérerons des systèmes *SLU* appris pour la résolution de la tâche PortMEDIA-it et réaliserons une inférence sur les données CommonVoice italiennes [1]. Nous avions déjà pu constater le gain d'un ajout de données françaises issues de CommonVoice pour la résolution de la tâche MEDIA [LAPERRIÈRE, PELLOIN, CAUBRIERE et al. 2022a ; LAPERRIÈRE, PELLOIN, CAUBRIERE et al. 2022b ; GHANNAY, CAUBRIERE, MDHAFFAR et al. 2021]. Ceci nous encouragea ici à voir si une spécialisation sur PortMEDIA-it détériore les performances de SAMU-XLSR sur cet ensemble de données alors même que son pré-apprentissage fut en partie réalisé sur celui-ci.

Nous souhaitons dans un second temps évaluer le possible gain d'une spécialisation de SAMU-XLSR pour des ensembles de données de même langue et de domaine proche. Nous considérerons ici des systèmes appris sur MEDIA pour des inférences réalisées sur l'ensemble de données PortMEDIA-fr présenté au Chapitre 4.

6.2.1 PortMEDIA-it vers CommonVoice

Cette section présente nos résultats expérimentaux pour l'inférence *ASR* de la version 9.0 de l'ensemble de données italien CommonVoice sur différents systèmes *SLU* initialement appris pour la résolution de la tâche PortMEDIA-it. Leur architecture fut décrite en Figure 5.2 et est ici composée des encodeurs XLS-R et SAMU-XLSR en guise de comparaison aux encodeurs spécialisés SAMU-XLSR $_{IT}$ discuté en Section 5.4 du Chapitre 5 et SAMU-XLSR $_{FR \oplus IT}$ discuté en Section 6.1.1 de ce chapitre.

Le domaine sémantique de CommonVoice correspond à de la lecture bénévole de textes provenant communément de l'encyclopédie Wikipédia [2]. Il est ainsi bien loin du domaine de réservation hôtelière et information touristique de PortMEDIA-it. Cet ensemble de données n'est cependant pas annoté sémantiquement. Ne disposant que d'enregistrements audio et leurs transcriptions

1. https://commonvoice.mozilla.org/fr/datasets
2. https://fr.wikipedia.org/wiki/

brutes, nous souhaitons ici évaluer l'évolution de la capacité de SAMU-XLSR à traiter et reconnaître le signal de parole des données CommonVoice suite à sa spécialisation sur des données cross-domaine de même langue. Nous utilisons donc les métriques de *ChER* (*Character Error Rate*) et *WER*, dont les scores sont donnés par la Table 6.9.

Nos expérimentations résultèrent en des scores approximant les 100 points de *WER*, quel que soit l'encodeur de parole utilisé, signifiant l'incapacité de notre modèle *SLU* à transcrire des mots issus d'un domaine trop lointain de celui appris. Notons ici que les données CommonVoice sont enregistrées de diverses manières. Les enregistrements audio de PortMEDIA-it sont réalisés par appel téléphonique, signifiant de possibles bruits de fond et interférences, sans compter une netteté vocale moindre, d'autant plus lorsque l'on considère l'année de diffusion de l'ensemble de données.

Les résultats donnés par la métrique *ChER* nous permettent de constater une nette différence entre les scores issus d'un apprentissage avec des encodeurs non-spécialisés et ceux issus d'un apprentissage avec des encodeurs spécialisés sur le domaine de MEDIA et PortMEDIA-it. Ces dégradations de plus de 2 points de *ChER* lors de l'utilisation de ces derniers nous indiquent une perte des facultés de SAMU-XLSR à traiter les données CommonVoice vues lors de son premier pré-apprentissage lorsque spécialisé pour un domaine trop distant.

		ChER	*WER*
figé	XLS-R + SLU $_{IT}$	$93,8$	$99,8$
	SAMU-XLSR + SLU $_{IT}$	**93,0**	$99,8$
	SAMU-XLSR $_{IT}$ + SLU $_{IT}$	$94,4$	$99,7$
	SAMU-XLSR $_{FR\oplus IT}$ + SLU $_{IT}$	$97,1$	$99,9$
fine-tuné	XLS-R + SLU $_{IT}$	**90,7**	$99,7$
	SAMU-XLSR + SLU $_{IT}$	$95,4$	$99,8$
	SAMU-XLSR $_{FR}$ + SLU $_{IT}$	$97,8$	$99,9$
	SAMU-XLSR $_{FR\oplus IT}$ + SLU $_{IT}$	$98,0$	$99,9$

TABLE 6.9 – Résultats *zero-shot* du corpus de *test* italien CommonVoice en *ChER* et *WER* pour les expérimentations de portabilité sémantique cross-domaine depuis PortMEDIA-it, avec XLS-R, SAMU-XLSR et ses spécialisations figés et *fine-tunés* [LAPERRIÈRE, H. NGUYEN et al. 2023a].

6.2.2 MEDIA vers PortMEDIA-fr

Cette section présente nos résultats expérimentaux pour l'inférence de PortMEDIA-fr, présenté au Chapitre 4, sur différents systèmes *SLU* initialement appris pour la résolution de la tâche MEDIA. Leur architecture fut décrite en Figure 5.2 et est ici composée des encodeurs XLS-R et SAMU-XLSR en guise de comparaison aux encodeurs spécialisés SAMU-XLSR$_{FR}$ discuté en Section 5.4 du Chapitre 5 et SAMU-XLSR$_{FR\oplus IT}$ discuté en Section 6.1.1 de ce chapitre.

Les données PortMEDIA-fr sont distribuées afin de proposer une tâche de portabilité cross-domaine de même langue depuis MEDIA, tout comme PortMEDIA-it est distribué pour une tâche de portabilité cross-lingue de même domaine depuis MEDIA. Nous souhaitons ainsi évaluer l'apport d'une spécialisation sémantique de SAMU-XLSR sur des données d'un domaine proche et d'une même langue. PortMEDIA-fr étant annoté en concepts sémantiques, nous utilisons pour cela les métriques *WER*, *CER* et *CVER* dont les scores sont donnés par la Table 6.10.

Les scores obtenus pour cette inférence semblent élevés. Il faut prendre en considération le changement de domaine entre MEDIA et PortMEDIA-fr ainsi que la possible complexité de cette nouvelle tâche. Les concepts sémantiques ne sont pas les seuls à avoir changé, le contenu linguistique étant lui aussi différent pour ces données. Les dialogues n'étant plus les mêmes, notre système *SLU* nécessiterait un affinage sur les données PortMEDIA-fr afin de les traiter au mieux.

Ce qui nous intéresse réellement se situe dans l'évolution des scores lors de l'utilisation d'un encodeur de parole spécialisé, d'autant plus lorsque cette spécialisation est multilingue. On peut constater des gains de performance importants, que ce soit en *fine-tunant* l'encodeur de parole sur MEDIA ou non. Nous constatons aussi que ce *fine-tuning* permet une meilleure compatibilité du système à la tâche PortMEDIA-fr. Nous pouvons affirmer qu'une spécialisation de SAMU-XLSR suivie de son *fine-tuning* permettent d'améliorer significativement l'extraction sémantique depuis la parole pour une tâche cible lorsque réalisée sur des données cross-domaine proches. De plus, l'amélioration considérable du *WER* pour une spécialisation multilingue démontre l'apport linguistique d'une telle approche pour le traitement de données cross-domaine et cross-lingue proches.

		WER	*CER*	*CVER*
figé	XLS-R + SLU $_{FR}$	$47,5$	$62,9$	$73,9$
	SAMU-XLSR + SLU $_{FR}$	$47,3$	$61,5$	$74,3$
	SAMU-XLSR $_{FR}$ + SLU $_{FR}$	$47,5$	$62,5$	$75,3$
	SAMU-XLSR $_{FR \oplus IT}$ + SLU $_{FR}$	**39,3**	**58,5**	**68,4**
fine-tuné	XLS-R + SLU $_{FR}$	$40,6$	$58,2$	$67,7$
	SAMU-XLSR + SLU $_{FR}$	$38,6$	$57,1$	$66,1$
	SAMU-XLSR $_{FR}$ + SLU $_{FR}$	$38,7$	**56,0**	**65,7**
	SAMU-XLSR $_{FR \oplus IT}$ + SLU $_{FR}$	**32,8**	$56,1$	$66,1$

TABLE 6.10 – Résultats *zero-shot* du corpus de *test* de PortMEDIA-fr en *WER*, *CER* et *CVER* pour les expérimentations de portabilité sémantique cross-domaine depuis MEDIA, avec XLS-R, SAMU-XLSR et ses spécialisations figés et *fine-tunés* [LAPERRIÈRE, H. NGUYEN et al. 2023a].

6.3 Conclusion

Ce chapitre conclut les contributions principales apportées par cette thèse en approfondissant les études monolingues présentées au Chapitre 5 concernant l'enrichissement sémantique d'un encodeur de parole et son application à une tâche complexe visant sa compréhension. Il apporte une dimension multilingue aux expérimentations précédentes à travers des apprentissages *SLU* cross-lingues et une portabilité entre langues et domaines plus ou moins distants.

La Section 6.1.1 poursuit les expérimentations de spécialisation sémantique de SAMU-XLSR présentées en Section 5.2 du chapitre précédent en y appliquant un contexte multilingue.

Lorsque l'encodeur spécialisé de manière cross-lingue est *fine-tuné* durant l'apprentissage *SLU*, l'amélioration de nos métriques est significative pour PortMEDIA-it. Ceci peut être expliqué par l'apport des données MEDIA plus conséquentes, de même domaine sémantique, d'environnement d'enregistrement audio similaire et de langue proche. Pour autant, une spécialisation même conjointe aux transcriptions tunisiennes de TARIC-SLU ne mène pas à une amélioration des performances sur cet ensemble de données. Ces résultats nous ont incités à chercher une autre manière de bénéficier au mieux de données multilingues de domaines sémantiques proches pour le traitement de TARIC-SLU.

Nous poussons par ailleurs l'étude de l'apport de données multilingues pour le traitement de PortMEDIA-it à travers des expérimentations de portabilité cross-lingue. Celles-ci mettent en évidence la capacité de nos systèmes à générer une annotation sémantique correcte jointe à une transcription incorrecte qui mélange les deux langues de cette portabilité, soit ici l'italien et le français. L'apport sémantique serait lié au portage depuis des données de même domaine, tandis que la perte linguistique serait due au sur-apprentissage de leurs transcriptions. Ces observations pourraient nous orienter vers des travaux de traduction automatique entre langues proches, suite à la génération d'une transcription multilingue grâce à ce portage *SLU* cross-lingue utilisant l'encodeur de parole SAMU-XLSR, lui-même en partie proposé pour le domaine de la traduction.

Une analyse linguistique et sémantique couche-par-couche des encodeurs spécialisés de manière multilingue est ensuite de nouveau réalisée. Elle permet de mettre en évidence un système *SLU* de moins de $71,5$ M de paramètres obtenant pour PortMEDIA-it le même *WER* qu'un système *SLU* de $387,8$ M de paramètres, cela grâce à la spécialisation de SAMU-XLSR sur les données PortMEDIA-it et MEDIA. Ce même système, n'utilisant que les 17 premières couches de SAMU-XLSR, obtient un *CER* à peine moins bon de deux points que le meilleur système jusqu'alors proposé pour le traitement de PortMEDIA-it. En réalisant sa portabilité depuis le français vers l'italien, nous obtenons un *CER* équivalant à celui obtenu par le même système de 24 couches *fine-tuné* durant l'apprentissage *SLU*, à $0,5$ points près.

Enfin, nous constatons l'apport des spécialisations multilingues de SAMU-XLSR jointes à une portabilité cross-lingue sur l'utilisation de ses représentations au *sentence-level*. Bien que la Sec-

tion 5.3 du chapitre précédent les montre peu pertinentes lorsque l'encodeur de parole spécialisé de manière monolingue est *fine-tuné* durant l'apprentissage *SLU*, le multilinguisme apporté par nos expérimentations permet une meilleure extraction sémantique de ces représentations contextuelles, menant à une amélioration significative des résultats en *CER* pour le corpus de *test* de PortMEDIA-it.

La Section 6.1.2 reprend les expérimentations de double spécialisation sémantique de SAMU-XLSR présentées en Section 5.4 du chapitre précédent en y appliquant ici aussi un contexte multilingue.

Ces expérimentations mènent à un *CER* à l'état-de-l'art pour les ensembles de données MEDIA $(17, 9)$ et TARIC-SLU $(29, 1)$, ainsi qu'à des résultats intéressants pour PortMEDIA-it $(24, 1$ de *CER*) bien qu'au-dessus de son état-de-l'art actuel $(21, 9$ de *CER*), obtenus d'un système moins imposant et de ce fait moins nécessiteux en ressources computationnelles.

Nous pouvons constater avec TARIC-SLU que l'utilisation de données d'un domaine sémantique proche mais de langues distantes durant la double spécialisation de SAMU-XLSR permet une meilleure extraction sémantique pour un ensemble de données étant peu doté et jamais considéré lors du premier pré-apprentissage de cet encodeur de parole.

Enfin, nous réalisons en Section 6.2 des analyses concernant la portabilité cross-domaine de nos systèmes *SLU*.

Il s'agit ici de constater l'évolution des capacités d'encodage linguistique et sémantique de SAMU-XLSR suite à sa spécialisation et son *fine-tuning*, pour le traitement de données initialement apprises lors de son premier pré-apprentissage mais aussi de nouvelles données jamais considérées jusqu'alors. Pour cela, nous orientons notre analyse sur l'inférence de systèmes *SLU* appris initialement sur des données d'un domaine plus ou moins distant de celui visé.

Les résultats obtenus pour une inférence sur les données italiennes CommonVoice suite à un apprentissage sur PortMEDIA-it démontrent l'incapacité d'un modèle SAMU-XLSR spécialisé sur des données cross-domaine distantes à traiter des données pourtant utilisées lors de son premier pré-apprentissage.

Nous constatons une amélioration significative du *CER* pour PortMEDIA-fr en utilisant des modèles appris sur le domaine proche de MEDIA avec une spécialisation monolingue ou multilingue de SAMU-XLSR. En revanche, c'est la spécialisation multilingue de cet encodeur de parole qui mène au meilleur score *WER* pour ces expérimentations, démontrant l'utilité d'une telle spécialisation pour le traitement de données d'un domaine proche.

CONCLUSIONS ET PERSPECTIVES

Nous apportons ici conclusions et perspectives aux travaux réalisés durant cette thèse.

Ceux-ci commencent par l'étude fine réalisée sur l'enrichissement sémantique de l'encodeur de parole SAMU-XLSR et de son encodage intermédiaire appliquée à une tâche de compréhension de la parole [LAPERRIÈRE, PELLOIN, ROUVIER et al. 2023 ; LARCHER et al. 2022]. Suit une proposition de spécialisation [LAPERRIÈRE, H. NGUYEN et al. 2023b ; LAPERRIÈRE, H. NGUYEN et al. 2023a] et double spécialisation [LAPERRIÈRE, GHANNAY et al. 2024] sémantique monolingue et cross-lingue de cet encodeur à un domaine visé lors de son utilisation pour une tâche d'extraction sémantique. Nous étudions dans un même temps l'apport de sa représentation sémantique d'un segment complet de parole. Une analyse complémentaire des capacités cross-domaine de SAMU-XLSR suite à sa spécialisation sémantique est finalement réalisée [LAPERRIÈRE, H. NGUYEN et al. 2023a].

Cette thèse présente par ailleurs la diffusion d'une recette SpeechBrain complète pour la tâche d'extraction sémantique proposée par l'ensemble de données françaises MEDIA [LAPERRIÈRE, PELLOIN, CAUBRIERE et al. 2022a ; LAPERRIÈRE, PELLOIN, CAUBRIERE et al. 2022b].

Conclusions

Cette thèse s'inscrit dans le cadre de l'Apprentissage Profond appliqué au domaine de la Compréhension Automatique de la Parole (*Spoken Language Understanding, SLU*). Son objectif principal consiste à tirer bénéfice de données existantes dans des langues bien dotées en annotation sémantique de la parole afin de développer des systèmes de compréhension performants dans des langues moins dotées. Cette compréhension est ici visée par une tâche riche et complexe d'extraction de concepts sémantiques depuis la parole.

Cette thèse aborde la tâche d'extraction sémantique comme celle d'une traduction depuis une langue source naturelle vers une langue cible conceptuelle. Ces dernières années ont connu des avancées considérables dans le domaine de la traduction de la parole. Une nouvelle approche permet de faire converger l'encodage de la parole de modèles auto-supervisés vers un encodage textuel appris sur de vastes quantités de données. L'encodeur de parole SAMU-XLSR [KHURANA et al. 2022] a été proposé pour la résolution d'une tâche de recherche d'information vocale et traduction de la parole, menant à des résultats à l'état-de-l'art dans ces domaines. Le rapprochement de ses représentations de la parole aux représentations textuelles de LaBSE [FENG et al.

2022] le rendent agnostique à la langue. Cette thèse étudie dans un premier temps l'enrichissement sémantique de SAMU-XLSR pour son application au domaine de la compréhension de la parole, avant d'explorer l'apport de données multilingues lors de différents affinages de ce modèle cross-lingue pour le traitement d'ensembles de données peu dotés.

Dans un souci d'accessibilité équitable à l'information et aux services que nous apportent les nouvelles technologies, nous étudions une tâche d'extraction sémantique de données annotées tunisiennes, une langue peu représentée dans le domaine du traitement de la parole. SAMU-XLSR n'étant pas appris sur des données tunisiennes, nos expérimentations sur l'ensemble de données TARIC-SLU [MDHAFFAR, BOUGARES et al. 2024] nous permettent d'évaluer sa capacité à traiter de nouvelles langues. Le traitement de cet ensemble de données étant complexe de par sa petite taille, cette thèse se tourne vers l'utilisation des données cross-lingues françaises MEDIA [BONNEAU-MAYNARD, ROSSET et al. 2005] et italiennes PortMEDIA-it [LEFÈVRE, MOSTEFA et al. 2012], menant à des résultats à l'état-de-l'art pour TARIC-SLU.

Étude de l'enrichissement sémantique d'un encodeur de parole

Nous souhaitons constater l'intérêt de la méthode d'apprentissage de SAMU-XLSR lors de son utilisation dans un système de compréhension de la parole. Nous comparons donc ses scores de *WER*, *CER* et *CVER* pour les ensembles de données MEDIA, PortMEDIA-it et TARIC-SLU à ceux de l'encodeur de parole multilingue XLS-R [BABU et al. 2022], utilisé pour l'apprentissage de SAMU-XLSR.

Nous constatons premièrement l'amélioration nette des métriques de *CER* et *CVER* lors de l'utilisation de l'encodeur de parole SAMU-XLSR dans notre système de compréhension de la parole. La similitude des *WER* lorsque les paramètres des encodeurs de parole sont figés durant l'apprentissage *SLU* finit de démontrer l'agnosticisme à la langue de SAMU-XLSR apporté par LaBSE.

Ce modèle ainsi que tous ses modules n'ayant jamais traité de tunisien mais seulement quelques langues sémitiques proches, nos expérimentations sur TARIC-SLU démontrent sa capacité à traiter de nouvelles langues lorsque *fine-tuné* sur celles-ci durant un apprentissage *SLU*.

Une analyse fine des représentations intermédiaires de SAMU-XLSR et XLS-R démontre la faculté de SAMU-XLSR à capturer efficacement la sémantique d'un segment audio dans ses couches médianes et à préserver cet encodage jusqu'à sa dernière couche, faisant de lui un encodeur de parole pertinent pour la résolution d'une tâche d'extraction sémantique. À contrario, nous pouvons constater qu'un encodeur multilingue classique tel que XLS-R détériore son encodage sémantique dans ses couches hautes.

Spécialisation sémantique d'un encodeur de parole

Nous avons ainsi constaté l'enrichissement sémantique conséquent de SAMU-XLSR apporté par la convergence de ses représentations de la parole vers les représentations textuelles de LaBSE. Nous nous intéressons par la suite à spécialiser cet enrichissement sémantique aux domaines sémantiques de nos ensembles de données. Les tâches ciblées relèvent d'une extraction sémantique pour la réservation d'hôtels ou de trains par appels téléphoniques. La spécialisation de SAMU-XLSR a pour objectif d'adapter son encodage sémantique à ces domaines.

La première étape consiste à continuer l'apprentissage de SAMU-XLSR en lui fournissant les transcriptions non-annotées et enregistrements audio de nos ensembles de données, les combinant afin de générer des spécialisations cross-lingues. La seconde étape consiste en un apprentissage *SLU* classique, utilisant les modèles SAMU-XLSR spécialisés en tant qu'encodeurs de parole, figeant ou non leurs paramètres.

Figer les paramètres de SAMU-XLSR nous permet de constater sa capacité à se concentrer sur un domaine sémantique précis lors d'une spécialisation ne spécifiant pourtant pas les concepts sémantiques à extraire lors d'un futur apprentissage. Une analyse poussée de ses représentations intermédiaires met en évidence des scores de *WER* similaires à ceux obtenus lors d'un *fine-tuning SLU* de SAMU-XLSR. Ces scores sont obtenus pour l'utilisation de seulement 17 des 24 couches de l'encodeur de parole, affinant $71,5$ M de paramètres contre $387,8$ M habituellement.

Une spécialisation monolingue de SAMU-XLSR *fine-tunée* lors de l'apprentissage *SLU* ne mène pas à une évolution significative des performances de nos systèmes d'extraction sémantique. Pour autant, l'apport de données cross-lingues permet une nette amélioration du traitement d'ensembles de données peu dotés lorsque leurs langues sont suffisamment proches.

Des expérimentations de portabilité cross-lingue entre le français et l'italien sont menées afin de fournir au système *SLU* une plus grande quantité de données annotées pour le domaine sémantique ciblé. Ces apprentissages *SLU* sur MEDIA puis sur PortMEDIA-it combinés à une spécialisation cross-lingue de SAMU-XLSR sur leurs transcriptions mènent à une amélioration significative du *CER* de PortMEDIA-it mais une dégradation tout autant significative de la transcription générée par le modèle, ceci dû à son sur-apprentissage pour la génération de transcriptions françaises. Une observation des sorties du modèle nous fait constater la prédiction correcte de concepts mêlée à une transcription mélangeant le vocabulaire français et italien. Cette constatation pourrait être plus amplement étudiée afin de tirer parti de cette transcription multilingue pour une tâche de traduction automatique depuis la parole.

Les tâches d'extraction sémantique traitent généralement des échantillons audio de 10 à 20 millisecondes. SAMU-XLSR génère des représentations sémantiques pour chaque segment de parole (*sentence-level*). C'est celles-ci qui sont poussées vers l'espace de représentation sémantique de LaBSE. Nous cherchons à tirer bénéfice de ces représentations afin d'amener contexte

et abstraction sémantique de haut niveau à nos représentations de la parole habituelles.

De prime abord, nos expérimentations confirment la pertinence des informations sémantiques encodées à ce niveau de représentation. Nos systèmes *SLU* gagnent grandement de leur utilisation lorsque SAMU-XLSR n'est pas *fine-tuné* durant l'apprentissage *SLU*.

En revanche, son *fine-tuning* pour nos tâche de compréhension de la parole démontre une probable duplication de l'information dans nos représentations habituelles, ne nécessitant qu'un simple affinage afin d'en extraire les mêmes informations.

Ces constatations ne s'appliquent pas lors de l'utilisation d'un modèle SAMU-XLSR spécialisé. Une autre possible explication à la stagnation de nos résultats sans spécialisation préalable peut donc se situer dans la difficulté à affiner une représentation supplémentaire à la fois trop abstraite et trop dense. La spécialisation servirait ici de première étape d'affinage.

Cette thèse présente par la suite une autre méthode de spécialisation qui continua à faire avancer l'état-de-l'art pour les ensembles de données MEDIA, PortMEDIA-it et TARIC-SLU. Cette double spécialisation sémantique a pour but de réunir en un unique système la spécialisation de SAMU-XLSR et son *fine-tuning SLU* afin de pouvoir :

— orienter son encodage de la parole vers le domaine sémantique visé ;
— préserver sa faculté à générer certaines abstractions sémantiques ;
— limiter la perte de ses capacités cross-lingues ;

Cette double spécialisation monolingue sur les ensembles de données MEDIA et PortMEDIA-it ne se différencie pas significativement des résultats déjà obtenus pour une spécialisation simple sur ces mêmes données.

Une double spécialisation sémantique multilingue française et italienne permet en revanche d'obtenir des résultats à l'état-de-l'art pour la version d'annotation sémantique *full* de MEDIA avec $17,9$ points de *CER*. En comparaison, ce système appris sur la version *full* de MEDIA permet d'obtenir $15,3$ points de *CER* lors d'une inférence réalisée sur sa version *relax*, communément utilisée par la communauté scientifique. L'état-de-l'art actuel pour cette version se situe à $12,1$ de *CER* pour des systèmes de bout-en-bout [DENISOV et VU 2023] et $10,8$ de *CER* pour des systèmes en cascade plus imposants et coûteux en ressources de calcul [PELLOIN 2024]. De la même manière que pour MEDIA, une double spécialisation multilingue française et italienne permet de faire avancer l'état-de-l'art actuel pour les données PortMEDIA-it, amenant à un *CER* de $24,1$ contre un état-de-l'art actuel à $21,9$ pour cet ensemble de données [DENISOV et VU 2023].

Comme dit précédemment, SAMU-XLSR n'a pas été appris sur des données tunisiennes. Seulement quelques langues sémitiques proches lui sont fournies lors de son pré-apprentissage. Nous avons constaté qu'il lui était difficile de s'adapter au domaine de TARIC-SLU avec une spécialisation et un *fine-tuning SLU* disjoints. Cette double spécialisation, même monolingue, permet un gain de performances pour cet ensemble de données. L'encodeur de parole parvient ici à s'affi-

ner pour générer un encodage propice à l'extraction sémantique de parole tunisienne tout en tirant bénéfice de son rapprochement vers les représentations textuelles cross-lingues de LaBSE.

Une double spécialisation sémantique multilingue sur nos données françaises, italiennes et tunisiennes réunies permet l'obtention de résultats à l'état-de-l'art pour TARIC-SLU, avec une amélioration significative de son *CER* à $29,1$ points. Nous constatons donc que l'utilisation combinée de données d'un domaine sémantique proche mais de langues distantes permet un enrichissement sémantique pertinent de l'encodeur de parole SAMU-XLSR lors de sa double spécialisation pour une tâche d'extraction sémantique complexe d'un ensemble de données peu doté.

Les Tables 6.11, 6.12 et 6.13 donnent un aperçu des résultats obtenus durant cette thèse pour nos données françaises MEDIA, italiennes PortMEDIA-it et tunisiennes TARIC-SLU avec XLS-R, SAMU-XLSR, les spécialisations SAMU-XLSR $_{LANG}$ et doubles spécialisation SAMU-XLSR $_{LANG\ dual}$, suivies d'un apprentissage SLU $_{LANG}$ affinant les paramètres de l'encodeur de parole ou les figeant.

		WER	*CER*	*CVER*
figé	XLS-R + SLU $_{FR}$	$21,7$	$33,8$	$46,0$
	SAMU-XSLR + SLU $_{FR}$	$21,3$	$27,4$	$41,6$
	SAMU-XLSR $_{FR}$ + SLU $_{FR}$	$12,7$	$21,3$	$32,4$
	SAMU-XLSR $_{FR\oplus IT}$ + SLU $_{FR}$	**12,5**	**20,8**	**31,6**
fine-tuné	XLS-R + SLU $_{FR}$	$13,5$	$21,8$	$32,8$
	SAMU-XSLR + SLU $_{FR}$	$11,7$	$18,7$	$29,4$
	SAMU-XLSR $_{FR}$ + SLU $_{FR}$	$11,6$	$18,6$	$29,1$
	SAMU-XLSR $_{FR\oplus IT}$ + SLU $_{FR}$	$11,5$	$18,5$	$29,1$
	SAMU-XLSR $_{FR\ dual}$	**10,6**	$18,3$	**27,8**
	SAMU-XLSR $_{FR\oplus IT\ dual}$	**10,6**	**17,9**	$28,2$

TABLE 6.11 – Sélection de résultats des corpora de *test* de MEDIA en *WER*, *CER* et *CVER* discutés durant cette thèse.

Enfin, nous concluons notre étude de l'enrichissement sémantique de SAMU-XLSR par l'analyse de l'impact de sa spécialisation sur sa capacité à traiter des données cross-domaine. Il s'agit alors de constater l'évolution des capacités d'encodage de SAMU-XLSR pour le traitement de nouveaux domaines sémantiques mais aussi de domaines déjà considérés lors de son premier apprentissage. Nous avons pour cela réalisé l'inférence des données italiennes CommonVoice [3] sur nos meilleurs systèmes pour PortMEDIA-it et l'inférence des données françaises PortMEDIA-fr

3. https://commonvoice.mozilla.org/fr/datasets

		WER	*CER*	*CVER*
figé	XLS-R + SLU $_{IT}$	33,7	42,1	57,1
	SAMU-XSLR + SLU $_{IT}$	31,5	33,6	49,0
	SAMU-XLSR $_{IT}$ + SLU $_{IT}$	19,6	30,3	42,9
	SAMU-XLSR $_{FR\oplus IT}$ + SLU $_{IT}$	18,7	29,4	41,6
	SAMU-XLSR $_{FR\oplus IT}$ + SLU $_{FR\rightarrow IT}$	17,3	26,5	39,2
	SAMU-XLSR $_{IT\oplus FR\ (17)}$ + SLU $_{FR\rightarrow IT}$	**15,9**	**25,6**	**38,0**
fine-tuné	XLS-R + SLU $_{IT}$	18,1	29,6	41,5
	SAMU-XSLR + SLU $_{IT}$	**15,4**	26,6	39,2
	SAMU-XLSR $_{IT}$ + SLU $_{IT}$	15,7	26,8	39,5
	SAMU-XLSR $_{FR\oplus IT}$ + SLU $_{IT}$	**14,5**	25,6	**37,6**
	SAMU-XLSR $_{FR\oplus IT}$ + SLU $_{FR\rightarrow IT}$	16,4	25,1	38,1
	SAMU-XLSR $_{FR\ dual}$	16,9	26,8	39,4
	SAMU-XLSR $_{FR\oplus IT\ dual}$	19,7	**24,1**	39,0

TABLE 6.12 – Sélection de résultats des corpora de *test* de PortMEDIA-it en *WER*, *CER* et *CVER* discutés durant cette thèse.

		WER	*CER*	*CVER*
figé	XLS-R + SLU $_{TU}$	58,1	49,1	71,0
	SAMU-XSLR + SLU $_{TU}$	63,8	51,3	74,7
	SAMU-XLSR $_{TU}$ + SLU $_{TU}$	**36,8**	**38,9**	**55,3**
fine-tuné	XLS-R + SLU $_{TU}$	39,6	34,5	51,1
	SAMU-XSLR + SLU $_{TU}$	**32,3**	30,7	47,4
	SAMU-XLSR $_{TU}$ + SLU $_{TU}$	22,9	30,3	**45,2**
	SAMU-XLSR $_{TU\ dual}$ + SLU $_{TU}$	**22,6**	29,9	46,8
	SAMU-XLSR $_{FR\oplus IT\oplus TU\ dual}$ + SLU $_{TU}$	22,8	**29,1**	46,2

TABLE 6.13 – Sélection de résultats des corpora de *test* de TARIC-SLU en *WER*, *CER* et *CVER* discutés durant cette thèse.

[LEFÈVRE, MOSTEFA et al. 2012] sur nos meilleurs systèmes pour MEDIA. Nous attestons ainsi l'incapacité de SAMU-XLSR à traiter des données utilisées lors de son premier apprentissage lorsque spécialisé sur des données d'un domaine distant. Pour autant, une amélioration significative du *CER* pour PortMEDIA-fr est constatée lors d'une spécialisation monolingue ou multilingue de SAMU-XSLR sur le domaine proche de MEDIA et PortMEDIA-it. Le *WER* de PortMEDIA-fr se

voit d'autant plus amélioré lors d'une spécialisation française et italienne cross-domaine démontrant l'efficacité d'une spécialisation multilingue pour le traitement de données peu nombreuses d'un domaine proche.

Diffusion du benchmark MEDIA

En complément de notre étude pour la spécialisation de l'enrichissement sémantique d'un encodeur de parole cross-lingue, cette thèse propose une recette complète de l'ensemble de données françaises MEDIA pour sa tâche d'extraction sémantique depuis la parole. Bien que faisant partie des très rares ensembles de données disponibles pour cette tâche, nous constatons que MEDIA est rarement utilisé au-delà de la communauté scientifique française. C'est pourquoi nous avons choisi l'outil open-source SpeechBrain [RAVANELLI, PARCOLLET et al. 2021] afin d'y intégrer notre recette. Cet outil permet dors et déjà une seconde mise en lumière de cet ensemble de données tout en promettant la persistance de sa recette dans les années à venir.

Perspectives

Cette partie présente quelques perspectives aux travaux réalisés durant cette thèse, tant au niveau de la gestion de nos ensembles de données qu'à celui de nos choix d'architectures neuronales.

Tirer bénéfice des représentations *sentence-level* de SAMU-XLSR

Cette thèse explore l'utilisation de la représentation au *sentence-level* générée par SAMU-XLSR pour l'ensemble d'un segment de parole. Nous avons conclu qu'une réduction de sa dimension ou un affinage supplémentaire devait être réalisé afin de pouvoir tirer pleinement bénéfice de son encodage. Nous avons tenté de réduire celle-ci avec l'aide d'un goulet d'étranglement ajouté dans l'architecture initiale de SAMU-XLSR. Cet ajout n'impactait pas les résultats de son apprentissage mais ne nous permettait pas, une fois la représentation intermédiaire extraite, d'améliorer les performances de notre système de compréhension de la parole. Ces travaux pourraient être continués afin d'obtenir une représentation compacte pertinente pour une tâche d'extraction sémantique. Une possibilité envisagée serait l'utilisation d'un auto-encodeur de débruitage (*Denoising Auto-Encoder*).

Une analyse complémentaire pourrait aussi être réalisée afin de mieux comprendre le comportement des couches d'attention générant cette représentation, par exemple via l'extraction de leurs paramètres pour différents segments de parole. Il serait ici intéressant de constater où se pose l'attention du modèle dans un segment fourni sans ses annotations sémantiques.

Spécialiser un encodeur de parole à des fins de portabilité cross-domaine

Comme nous l'avons constaté avec l'inférence de l'ensemble de données PortMEDIA-fr sur un modèle appris sur les données de même langue MEDIA, une spécialisation de SAMU-XLSR sur un domaine proche de celui ciblé permettrait un meilleur traitement de ce dernier. Il serait intéressant de continuer ces expérimentations avec cette fois des apprentissages complets plutôt que de simples inférences sur des données cross-domaine.

Utiliser de nouvelles données multilingues

Afin de compléter nos expérimentations multilingues sur des données de langues proches latines, il serait envisageable d'utiliser l'ensemble de données espagnoles Dihana [BENEDÍ et al. 2004]. Celui-ci est annoté sémantiquement et correspond au domaine de MEDIA et PortMEDIA. Ses dialogues sont composés d'enregistrements téléphoniques entre un utilisateur et un magicien d'Oz pour la réservation de trains.

Contextualiser les tours de parole «humain» grâce aux tours de parole «machine»

L'ensemble de données MEDIA est fourni avec les transcriptions des tours utilisateurs annotés sémantiquement mais aussi les transcriptions brutes des tours de son interlocuteur le magicien d'Oz. Ces derniers ne sont pas considérés dans nos expérimentations. Il serait possible de les utiliser afin d'apporter un contexte supplémentaire aux tours de parole de l'utilisateur. Une idée serait de les encoder en une représentation *sentence-level* tel que réalisé dans SAMU-XLSR. Il serait alors possible de les concaténer aux représentations *frame-level* de l'utilisateur. Une autre possibilité serait de les traiter comme des tours de parole classiques en retirant préalablement le premier et dernier tour de la conversation téléphonique correspondants aux formules d'accueil et de clôture de l'appel.

Optimiser la phase de décodage des représentations de la parole

Une perspective d'amélioration de nos architectures neuronales se situe dans le module de décodage des représentations de la parole générées par SAMU-XLSR. Ce module est actuellement composé de couches récurrentes et denses.

Il serait intéressant d'évaluer les performances des couches *Sli-GRU* récemment proposées en tant qu'amélioration des *Li-GRU* pour des tâches de traitement de la parole [MOUMEN et PARCOLLET 2023].

Une possibilité qui nous contraindrait à augmenter drastiquement le nombre de paramètres de nos systèmes *SLU* serait l'utilisation de modèles Transformer [VASWANI et al. 2017] ou plus simplement de mécanismes d'attention. Nous avons pu constater que les systèmes à l'état-de-l'art

pour la tâche MEDIA étaient généralement composés de modules d'encodage et décodage avec mécanismes d'attention [PELLOIN 2024]. Denisov et Vu [2023] proposent une approche combinant le modèle mBART [Y. LIU, GU et al. 2020] à l'utilisation de couches Conformer.

Réaliser un SAMU-Voxpopuli et SAMU-LeBenchmark monolingues

Tout comme SAMU-XLSR est appris en partant de l'encodeur de parole cross-lingue XLS-R, il serait possible de réaliser l'apprentissage de modèles SAMU-XLSR pour une langue précise. Nous pensons ici à un SAMU-Voxpopuli [C. WANG, RIVIERE et al. 2021] italien ou un SAMU-LeBenchmark [EVAIN, H. NGUYEN et al. 2021] français. L'intérêt serait de permettre à un modèle monolingue d'enrichir son encodage sémantique grâce à une modalité textuelle telle que LaBSE ou autre encodeur textuel adapté à la tâche visée. La contrainte principale de la conception d'un tel modèle se situe dans le coût d'apprentissage de la méthode SAMU et la quantité de données nécessaire.

Appliquer ces études à de nouveaux encodeurs de parole

L'ensemble de ces études pourrait être mené sur différents encodeurs de parole. Nous pourrions par exemple considérer SONAR [DUQUENNE, SCHWENK et al. 2023], lui aussi multi-modal et multilingue. Celui-ci n'étant pas appris pour correspondre aux représentations textuelles de LaBSE mais à celles du modèle NLLB [NLLB-TEAM 2022], il serait aussi envisageable d'étudier l'apport de différents encodeurs textuels pour ces apprentissages multi-modaux à l'état-de-l'art. Une étude des capacités d'encodage sémantique de Whisper [RADFORD, J. KIM et al. 2023] pourrait aussi être faite.

Appliquer ces études à la reconnaissance d'entités nommées

Cette thèse étudie une tâche d'extraction de concepts sémantiques. Le principe d'une annotation en concepts sémantiques est fortement similaire à celui d'une annotation en entités nommées. Les entités nommées ne correspondent pas au domaine sémantique d'un ensemble de données précis mais sont plus génériques et ainsi moins complexes. Ceci a pour conséquence l'existence d'une plus vaste quantité de données disponible pour la résolution de cette tâche.

Il serait possible d'appliquer nos expérimentations à une tâche de reconnaissance d'entités nommées avec les ensembles de données françaises ETAPE [GRAVIER, ADDA et al. 2012] et ESTER [GRAVIER, BONASTRE et al. 2004], allemandes et anglaises CoNLL-2003 [TJONG KIM SANG et DE MEULDER 2003] et bien d'autres [4].

4. urlhttps://github.com/juand-r/entity-recognition-datasets

Proposer une nouvelle méthode de normalisation pour la métrique de CVER

Comme discuté dans cette thèse, la métrique *CVER* souvent utilisée pour l'évaluation de l'ensemble de données MEDIA est perfectible. Il devient nécessaire de trouver un moyen de normaliser les mots-supports prédits par les systèmes traitant cet ensemble de données qui ne passe pas par la création de règles humaines. Il serait possible de les évaluer de manière phonétique afin de ne pas prendre en considération les éventuelles fautes de prédiction n'impactant pas notre compréhension du groupe de mots. Des solutions neuronales pourraient aussi être explorées afin de générer une valeur normalisée.

Re-segmenter PortMEDIA-it

Nous avons constaté par nos analyses statistiques de PortMEDIA-it mais aussi de MEDIA et TARIC-SLU que l'ensemble de données italiennes consistait en un nombre d'heures de parole élevé (14h 41m) pour un nombre de mots ($44,1$ k) pourtant bien plus faible que TARIC-SLU (8h 53m de dialogue pour 69 k mots). Nous nous questionnons donc sur la segmentation de cet ensemble de données et la présence de blancs de parole dans les tours de l'utilisateur. Nous avons segmenté à nouveau MEDIA pour la recette SpeechBrain et nos expérimentations afin de retirer les blancs de parole indiqués par des balises présentes dans la distribution d'origine de l'ELRA. Il serait intéressant d'appliquer des méthodes de segmentation de la parole aux corpora de PortMEDIA-it afin d'éclaircir la raison derrière cette incohérence.

ANNEXES

Approche cascade pour MEDIA

Au commencement de cette thèse, j'ai pu collaborer avec Ghannay et al. [2021] pour l'évaluation des systèmes à l'état-de-l'art pour le traitement d'une tâche d'extraction sémantique telle que MEDIA [BONNEAU-MAYNARD, ROSSET et al. 2005]. L'article publié fait état des architectures propices à la réalisation d'un système en cascade, composé d'un module de reconnaissance de la parole suivi d'un module de compréhension de celle-ci, comme abordé au Chapitre 2. Il compare ces systèmes à des systèmes de bout-en-bout, comme ceux utilisés au long de cette thèse. Après avoir rappelé les bénéfices et inconvénients de ces deux approches et précisé l'utilisation d'une fonction de coût CTC [GRAVES, FERNÁNDEZ et al. 2006] dans leurs études, l'article présente ses expérimentations.

Le module *ASR* du système en cascade proposé est composé d'un modèle wav2vec 2.0 LeBenchmark [EVAIN, H. NGUYEN et al. 2021], décrit en Chapitre 3 de cette thèse. Cet encodeur de parole français est utilisé afin de générer la transcription brute des enregistrements audio de MEDIA, étant simplement suivit d'une couche linéaire de 1024 neurones et d'une fonction Softmax. Ce modèle est préalablement *fine-tuné* sur les données audio françaises CommonVoice 6.1 [5] et leurs transcriptions non-annotées sémantiquement avant de l'être à nouveau sur les transcriptions de MEDIA.

Le module *NLU* du système en cascade proposé est composé d'un modèle BERT CamemBERT [MARTIN et al. 2019], permettant d'annoter en concepts sémantiques les transcriptions générées par le précédent module. Celui-ci est aussi *fine-tuné* pour résoudre la tâche correspondant à la version *relax* de MEDIA. Nous rappelons que les travaux présentés dans cette thèse portent sur la version *full* de cet ensemble de données, dont l'annotation est plus complexe et correspond à celle utilisée pour PortMEDIA-it.

Ce système en cascade sera noté dans la Table 14 «LeBenchmark $_{CV \rightarrow M(ASR)}$ + CamemBERT $_{M(SLU)}$» afin de correspondre aux notations utilisées dans cette thèse.

Le système de bout-en-bout utilisé est composé du même encodeur de parole ainsi que de la même couche linéaire puis fonction Softmax, pouvant réaliser les mêmes apprentissages que

5. https://commonvoice.mozilla.org/fr/datasets

lorsque utilisé en tant que module *ASR* du système en cascade. Les différents systèmes qui en découlent sont cependant obligatoirement *fine-tunés* sur la tâche *SLU* d'extraction sémantique de MEDIA. Ils seront nommés en fonction des étapes d'apprentissage réalisées.

La Table 14 présente les résultats obtenus en *CER* et *CVER* normalisé selon des règles humaines, comme discuté au Chapitre 4. Les scores *CVER* sont donc d'autant moins comparables aux résultats présentés précédemment dans cette thèse. Cet article ne détenant plus à ce jour les résultats à l'état-de-l'art pour la version *relax* de MEDIA, il aura cependant permis de mettre en évidence les avantages d'un système en cascade pour cette tâche, en insistant sur le lien de causalité entre performances et quantité de données pour des systèmes de bout-en-bout.

		CER	*CVER*
Bout-en-bout	LeBenchmark$_{M(SLU)}$	$18,8$	$23,6$
	LeBenchmark$_{CV \rightarrow M(SLU)}$	$15,8$	$20,4$
	LeBenchmark$_{CV \rightarrow M(ASR) \rightarrow M(SLU)}$	$14,5$	$18,8$
Cascade	LeBenchmark$_{CV \rightarrow M(ASR)}$ + CamemBERT$_{M(SLU)}$	**11,2**	**17,2**

TABLE 14 – Résultats en *CER* et *CVER* normalisé du corpus de *test* de la version *relax* de MEDIA pour des architectures en cascade et de bout-en-bout [GHANNAY, CAUBRIERE, MDHAFFAR et al. 2021].

Enrichissement en intentions du corpus MEDIA

Au cours de cette thèse, j'ai eu l'opportunité de collaborer avec Alavoine et al. [2024] pour l'ajout d'annotations d'intention pour l'ensemble de données MEDIA en version *full* et *relax* distribué à travers la recette SpeechBrain présentée au Chapitre 4. Le principe de reconnaissance d'intention d'un segment audio pour la compréhension de la parole est décrit dans la première section du Chapitre 2. L'article publié explique le processus d'annotation en intention avant de réaliser quelques expérimentations évaluant l'apport de cette nouvelle tâche sur la tâche d'extraction de concepts sémantiques déjà existante et inversement.

Ces deux tâches y sont apprises par un unique système réalisant le classification d'intention et la reconnaissance de concepts sémantiques simultanément. La première tâche est évaluée en *EMR* (*Exact Match Ratio*) [SOROWER 2010] et *accuracy* [GODBOLE et SARAWAGI 2004] tandis que la seconde est évaluée en *CER* et F-mesure des concepts prédits passés au format *multi-hot* [VAN RIJSBERGEN 1974], l'équivalent du format *one-hot* décrit au Chapitre 2 pour l'activation possible de plusieurs classes. Un score *SFA* (*Semantic Frame Accuracy*) est aussi proposé afin d'évaluer la justesse de prédiction des deux tâches pour un même segment audio [WELD et al. 2022].

Ici aussi, une architecture en cascade et une architecture de bout-en-bout sont proposées afin d'apporter les premiers résultats expérimentaux pour la résolution conjointe de ces tâches.

Le module *ASR* de l'architecture en cascade est tel que présenté en Figure 5.2, à la différence qu'il utilise l'encodeur de parole français LeBenchmark. Concernant le module *NLU*, de nombreux modèles sont évalués. Pour l'annotation sémantique des annotations manuelles de MEDIA, différents modèles permettent d'obtenir les meilleurs scores dans une ou deux métriques, sans qu'un unique modèle ne se démarque particulièrement. Lorsque celles-ci sont générées automatiquement grâce au module *ASR*, des modèles *NLU* se démarquent, notamment FlauBERT [HERVÉ et al. 2022] et CamemBERT.

L'architecture de bout-en-bout suit aussi celle présentée en Figure 5.2. Sont testés les encodeurs de parole LeBenchmark et SAMU-XLSR [KHURANA et al. 2022] ainsi que ses versions spécialisées sur MEDIA et PortMEDIA-it [LAPERRIÈRE, H. NGUYEN et al. 2023b ; LAPERRIÈRE, H. NGUYEN et al. 2023a]. La Table 15 présente quelques scores obtenus pour cette architecture. Sur la version *full* de MEDIA distribuée via la recette SpeechBrain présentée au Chapitre 4, l'encodeur spécialisé de manière cross-lingue SAMU-XLSR $_{FR\oplus IT}$ mène aux meilleurs résultats pour les deux tâches. Pour la version *relax* distribuée de la même manière, l'encodeur de parole LeBenchmark est légèrement plus performant pour la tâche d'extraction de concepts sémantiques tandis que l'encodeur non-spécialisé SAMU-XLSR obtient de peu les meilleurs résultats de classification d'intention. Les scores *SFA* attirent tout particulièrement notre attention, avec une différence de performance plus significative. Elle l'est d'autant plus considérant les meilleurs *SFA* de $65,0$ pour la version *relax* de MEDIA et $64,1$ pour sa version *full* obtenus avec un système en cascade, démontrant une bien meilleure optimisation jointe des deux tâches dans un système de bout-en-bout.

		accuracy	*EMR*	F-mesure	*CER*	*SFA*
relax	SAMU-XLSR	**90,7**	**88,9**	90,0	15,3	68,5
	SAMU-XLSR $_{FR}$	90,4	88,3	90,7	15,1	69,4
	SAMU-XLSR $_{FR\oplus IT}$	90,5	88,6	90,7	15,2	70,8
	LeBenchmark FR 3k large	90,2	88,0	**90,8**	**15,1**	**71,1**
full	SAMU-XLSR	90,5	88,7	88,9	18,5	69,7
	SAMU-XLSR $_{FR}$	90,3	88,2	88,1	19,7	68,5
	SAMU-XLSR $_{FR\oplus IT}$	**91,0**	**88,9**	**89,1**	**18,3**	**70,6**
	LeBenchmark FR 3k large	90,1	88,0	88,0	19,7	69,1

TABLE 15 – Résultats en *EMR*, *accuracy*, F1-mesure, *CER* et *SFA* du corpus de test des versions full et relax de MEDIA annotées en intentions pour une architecture de bout-en-bout telle que décrite par la Figure 5.2 pour divers encodeurs de parole *fine-tunés* [ALAVOINE et al. 2024].

Coûts environnementaux

Cette thèse, annexes comprises, aura nécessité $65\ 439,4$ heures de calcul sur des cartes graphiques V100 de 32 Go de mémoire et $410,12$ heures de calcul sur des cartes graphiques A100 de 80 Go de mémoire localisées dans le super-calculateur français Jean Zay. Ceci correspond selon l'infrastructure de recherche GENCI à environ $1\ 740$ kg d'équivalence CO_2.

En terme de déplacements professionnels, cette thèse aura nécessité plusieurs trajets à l'international pour des vols totalisant $3\ 403$ kg d'équivalence CO_2.

Ontologie de l'annotation sémantique de MEDIA

Les étiquettes, spécificateurs et modificateurs de MEDIA, et la base de ceux de PortMEDIA-fr et PortMEDIA-it sont recensées des Figures 4 à 16. Leur légende est présentée en Figure 3. Une description complète de ces ensembles de données est donnée au Chapitre 4.

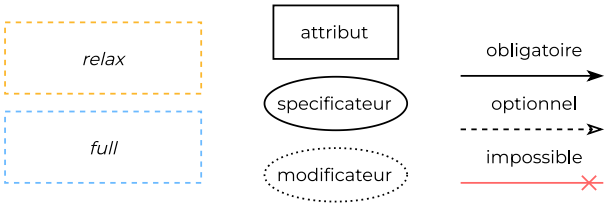

FIGURE 3 – Légende pour les Figures 4 à 16.

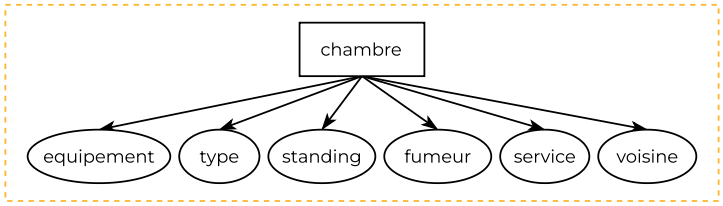

FIGURE 4 – Concept «chambre» dans MEDIA [BONNEAU-MAYNARD, ROSSET et al. 2005].

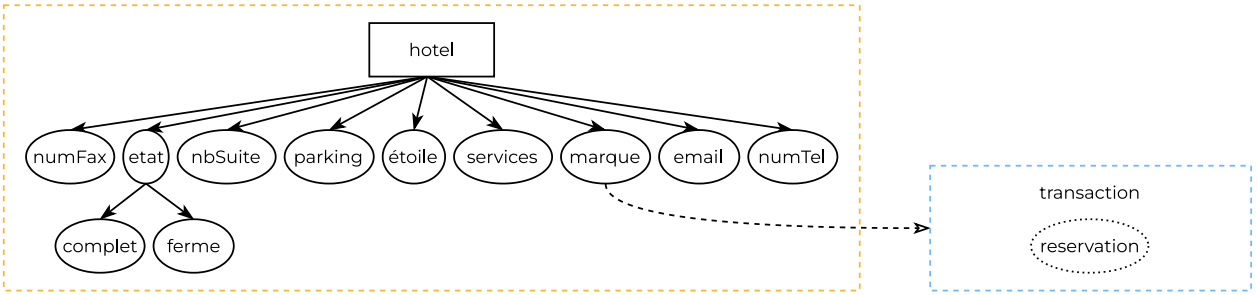

FIGURE 5 – Concept «hotel» dans MEDIA [BONNEAU-MAYNARD, ROSSET et al. 2005].

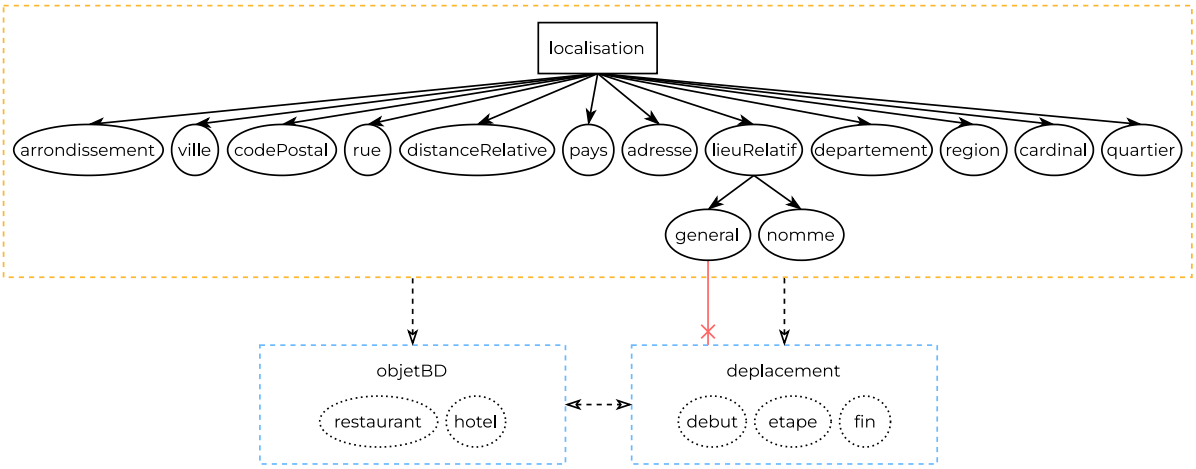

FIGURE 6 – Concept «localisation» dans MEDIA [BONNEAU-MAYNARD, ROSSET et al. 2005].

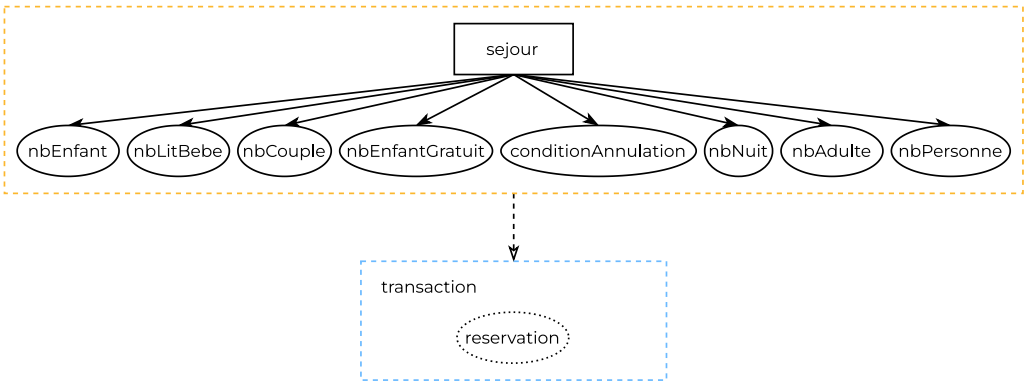

FIGURE 7 – Concept «sejour» dans MEDIA [BONNEAU-MAYNARD, ROSSET et al. 2005].

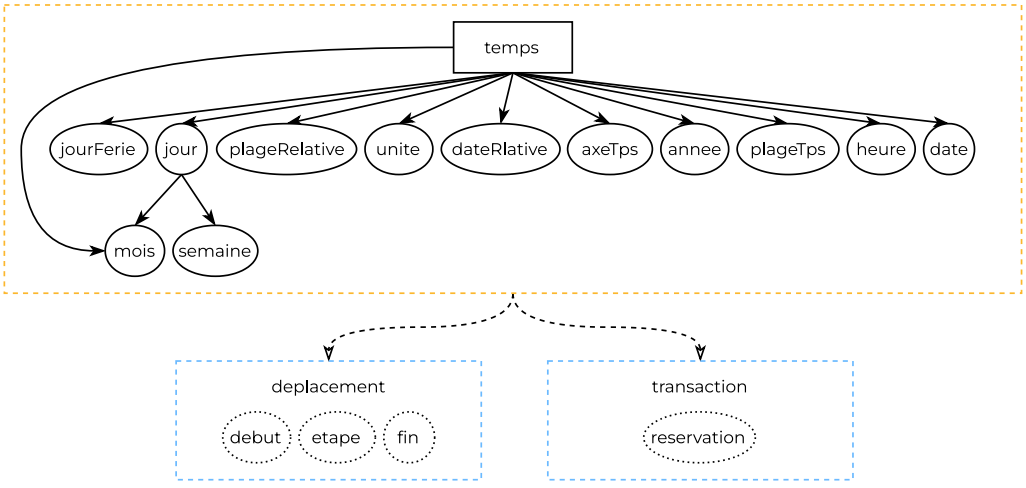

FIGURE 8 – Concept «temps» dans MEDIA [BONNEAU-MAYNARD, ROSSET et al. 2005].

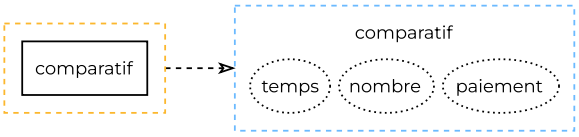

FIGURE 9 – Concept «comparatif» dans MEDIA [BONNEAU-MAYNARD, ROSSET et al. 2005].

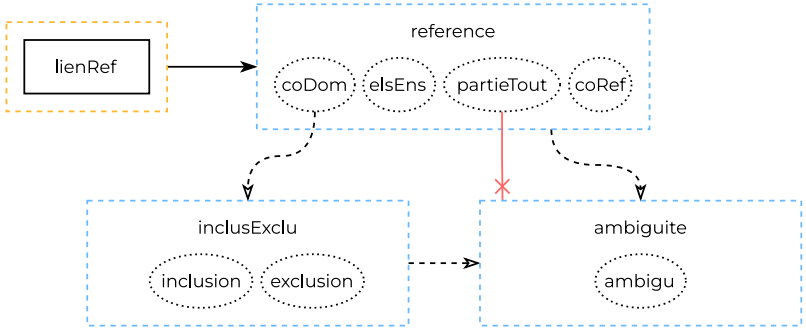

FIGURE 10 – Concept «lienRef» dans MEDIA [BONNEAU-MAYNARD, ROSSET et al. 2005].

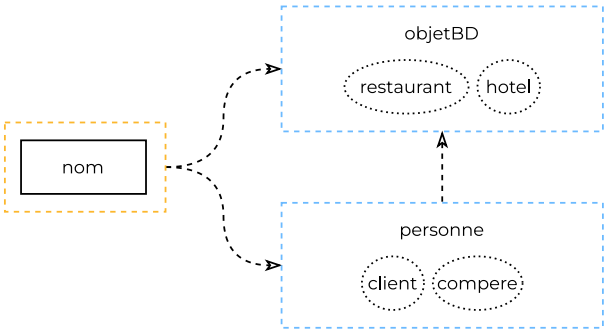

FIGURE 11 – Concept «nom» dans MEDIA [BONNEAU-MAYNARD, ROSSET et al. 2005].

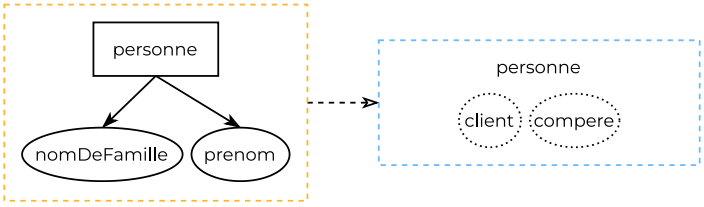

FIGURE 12 – Concept «personne» dans MEDIA [BONNEAU-MAYNARD, ROSSET et al. 2005].

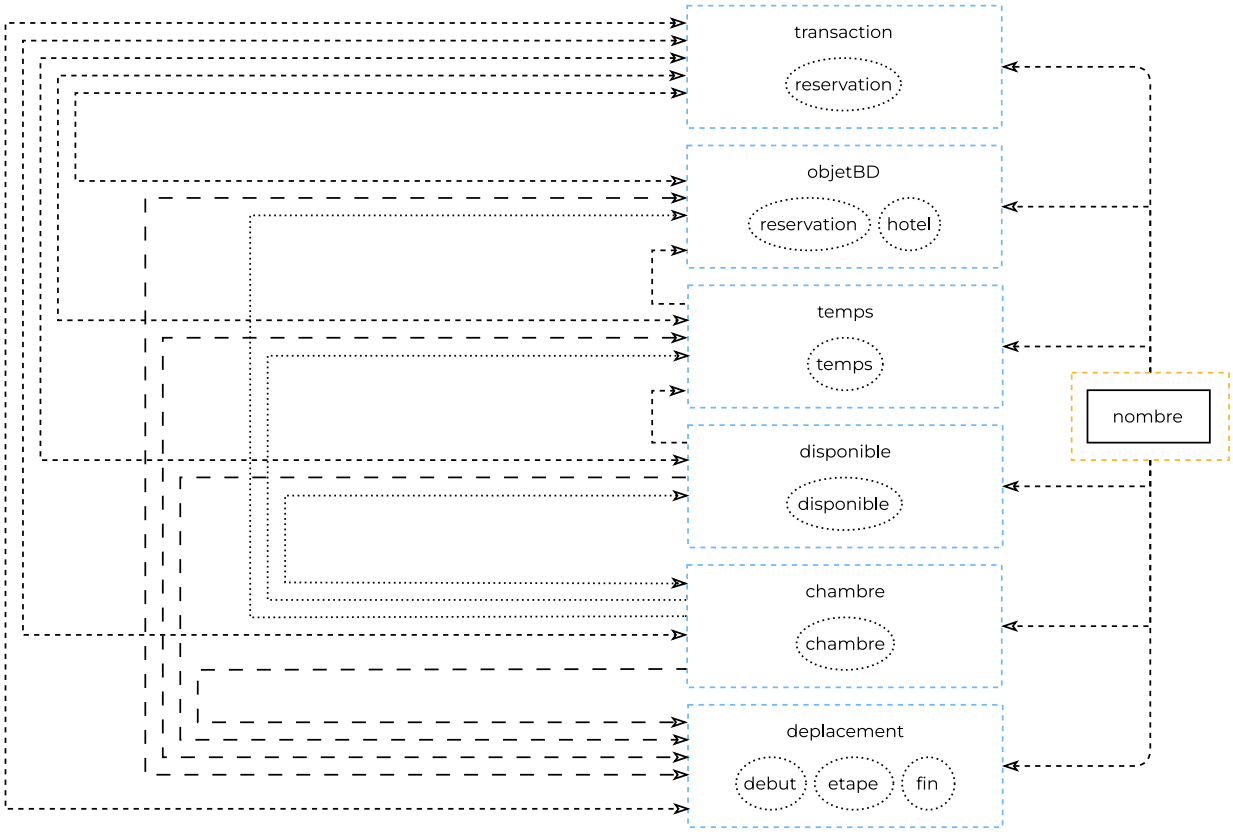

FIGURE 13 – Concept «nombre» dans MEDIA [BONNEAU-MAYNARD, ROSSET et al. 2005].

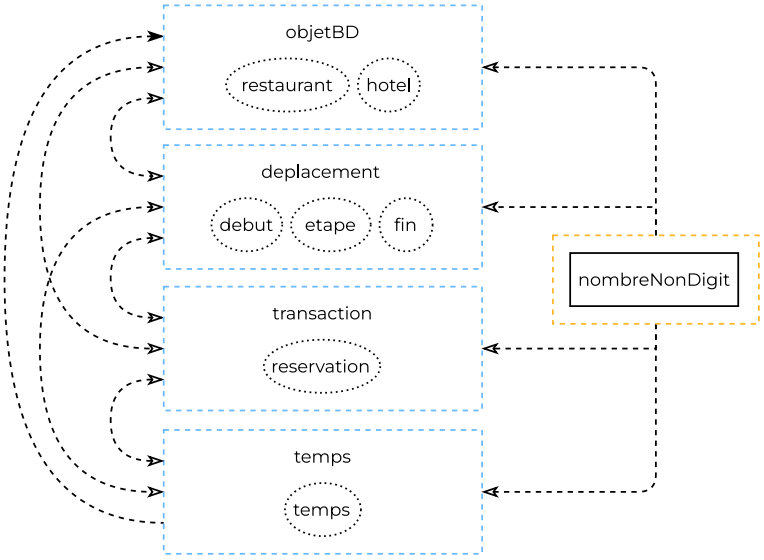

FIGURE 14 – Concept «nombreNonDigit» dans MEDIA [BONNEAU-MAYNARD, ROSSET et al. 2005].

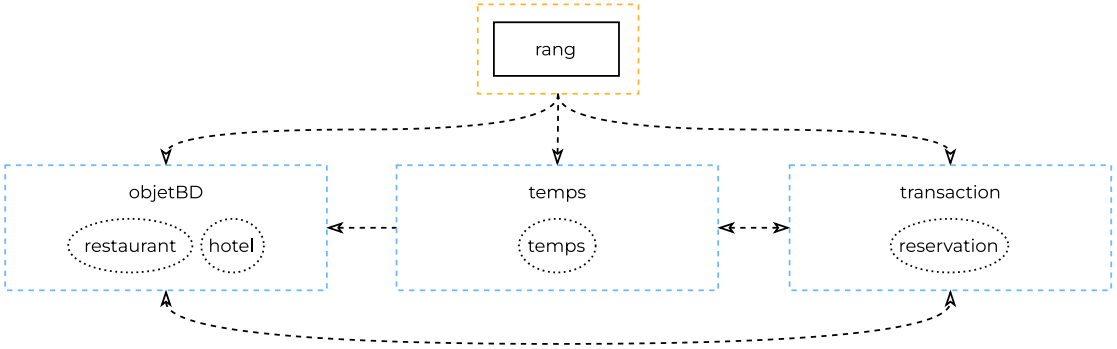

FIGURE 15 – Concept «rang» dans MEDIA [BONNEAU-MAYNARD, ROSSET et al. 2005].

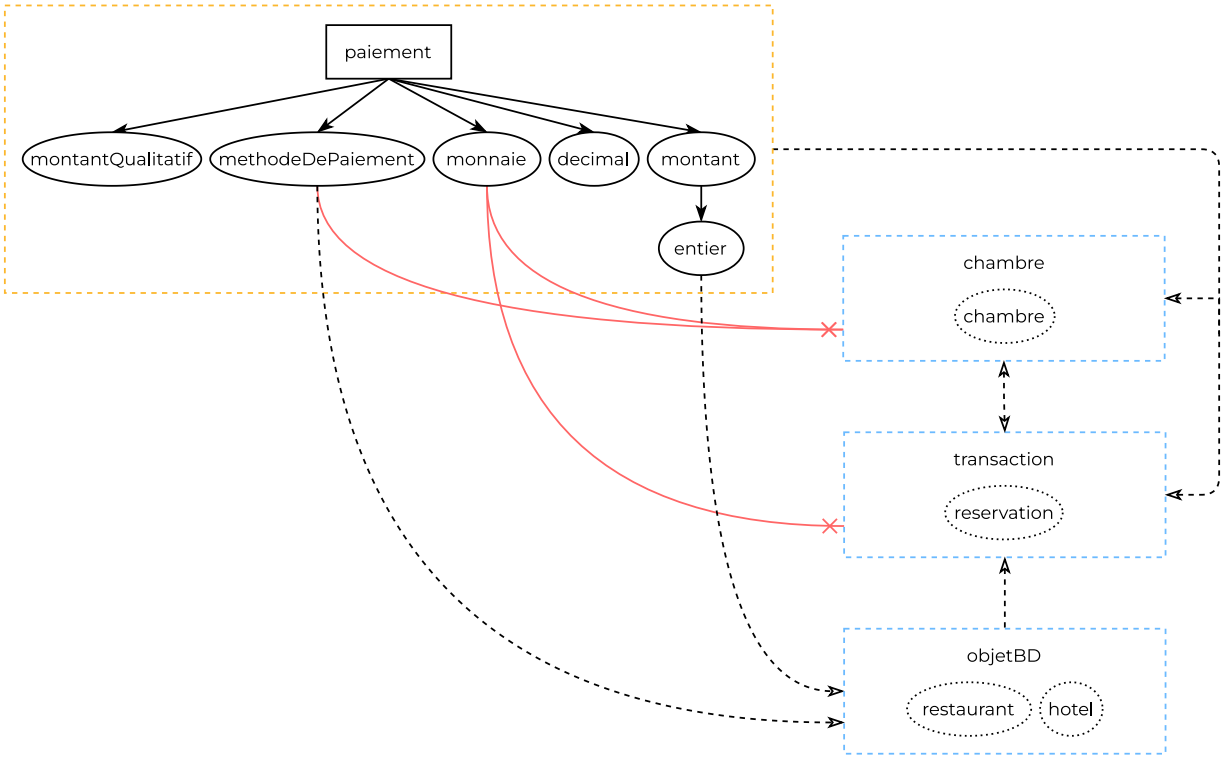

FIGURE 16 – Concept «paiement» dans MEDIA [BONNEAU-MAYNARD, ROSSET et al. 2005].

ACRONYMES

Adam	*Adaptative Moment Estimation*
ASR	Reconnaissance de la parole (*Automatic Speech Recognition*)
Bi-GRU	GRU bi-directionnel (*Bidirectional GRU*)
Bi-LSTM	LSTM bi-directionnel (*Bidirectional LSTM*)
BIO	*Beginning Inside Outside*
CER	Taux d'erreur de concepts (*Concept Error Rate*)
ChER	Taux d'erreur de caractères (*Character Error Rate*)
CNN	Réseau de neurones convolutif (*Convolutional Neural Network*)
CRF	Champs aléatoires conditionnels (*Conditional Random Fields*)
CTC	Classification Temporelle Connexionniste (*Connectionist Temporal Classification*)
CVER	Taux d'erreur de concepts et valeurs (*Concept and Value Error Rate*)
DNN	Réseau de neurones dense (*Dense Neural Network*)
FFT	Transformée de Fourier rapide (*Fast Fourier Transform*)
FSM	Automate à état fini (*Finite State Machine*)
GMM	Modèle de mélange Gaussien (*Gaussian Mixture Model*)
GRU	Unité récurrente fermée (*Gated Recurrent Unit*)
HMM	Modèle de Markov caché (*Hidden Markov Model*)
LLM	Large modèle de langue (*Large Language Model*)
LSTM	Mémoire à court et long terme (*Long-Short Term Memory*)
MFCC	Coefficients cepstraux à l'échelle de Mel (*Mel-Frequency Cepstral Coefficients*)
MSE	Erreur Quadratique Moyenne (*Mean Squared Error Rate*)
NLU	Compréhension du langage naturel (*Natural Language Understanding*)
RNN	Réseau de neurones récurrent (*Recurrent Neural Network*)
SLU	Compréhension de la parole (*Spoken Language Understanding*)
SSL	Apprentissage auto-supervisé (*Self-Supervised Learning*)
SVM	Machines à vecteur de support (*Support Vector Machines*)
WAV	*Waveform audio file format*
WER	Taux d'erreur de mots (*Word Error Rate*)
WoZ	Magicien d'Oz (*Wizard of Oz*)

PUBLICATIONS PERSONNELLES

N. ALAVOINE, G. LAPERRIERE, C. SERVAN, S. GHANNAY et S. ROSSET. “New semantic task for the french Spoken Language Understanding MEDIA benchmark”. *Language Resources and Evaluation Conference (LREC)* (2024), p. 12227-12246. DOI : 10.48550/arXiv.2403.19727 (cité pages 184, 185).

S. GHANNAY, A. CAUBRIERE, S. MDHAFFAR, G. LAPERRIÈRE, B. JABAIAN et Y. ESTÈVE. “Where are we in semantic concept extraction for Spoken Language Understanding ?” : *Speech and Computer (SPECOM)* (2021). DOI : 10.48550/arXiv.2106.13045 (cité pages 73, 79, 81, 83, 126, 167, 183, 184).

G. LAPERRIÈRE, S. GHANNAY, B. JABAIAN et Y. ESTÈVE. “A dual task learning approach to fine-tune a multilingual semantic speech encoder for Spoken Language Understanding”. *Interspeech* (2024). DOI : 10.48550/arXiv.2406.12141 (cité pages 134, 137, 142, 146, 149, 151-153, 156, 162, 164-166, 173).

G. LAPERRIÈRE, H. NGUYEN, S. GHANNAY, B. JABAIAN et Y. ESTÈVE. “Semantic enrichment towards efficient speech representations”. *Interspeech* (2023), p. 705-709. DOI : 10.21437/Interspeech.2023-2234 (cité pages 134, 137, 140, 143, 145, 146, 156-160, 162-164, 167-169, 173, 185).

G. LAPERRIÈRE, H. NGUYEN, S. GHANNAY, B. JABAIAN et Y. ESTÈVE. “Specialized semantic enrichment of speech representations”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2023), p. 1-5. DOI : 10.1109/ICASSPW59220.2023.10193452 (cité pages 134, 137, 140, 143, 145, 146, 156-160, 162, 173, 185).

G. LAPERRIÈRE, V. PELLOIN, A. CAUBRIERE, S. MDHAFFAR, N. CAMELIN, S. GHANNAY, B. JABAIAN et Y. ESTÈVE. “Le benchmark MEDIA revisité : données, outils et évaluation dans un contexte d’apprentissage profond”. *Journées d’Étude sur la Parole (JEP)* (2022). URL : https://hal.science/hal-03770588 (cité pages 22, 116, 121, 122, 129, 130, 132, 167, 173).

G. LAPERRIÈRE, V. PELLOIN, A. CAUBRIERE, S. MDHAFFAR, N. CAMELIN, S. GHANNAY, B. JABAIAN et Y. ESTÈVE. “The Spoken Language Understanding MEDIA Benchmark Dataset in the era of Deep Learning : Data updates, training and evaluation tools”. *Language Resources and Evaluation*

Conference (LREC) (2022), p. 1595-1602. URL : https://aclanthology.org/2022.lrec-1.171 (cité pages 22, 116, 121, 122, 129, 130, 132, 167, 173).

G. LAPERRIÈRE, V. PELLOIN, M. ROUVIER, T. STAFYLAKIS et Y. ESTÈVE. "On the use of semantically-aligned speech representations for Spoken Language Understanding". *Spoken Language Technology (SLT)* (2023). DOI : 10.48550/arXiv.2210.05291 (cité pages 134-137, 142, 156, 157, 173).

A. LARCHER, Y. ESTÈVE, M. ROUVIER et al. "Multi-lingual speech to speech translation for under-resourced languages". (2022). URL : https://hal.science/hal-04176910 (cité pages 134, 135, 156, 173).

A. LAURENT, S. GAHBICHE, H. NGUYEN et al. "ON-TRAC consortium systems for the IWSLT 2023 dialectal and low-resource speech translation tasks". *International Conference on Spoken Language Translation (IWSLT)* (2023), p. 219-226. DOI : 10.18653/v1/2023.iwslt-1.18 (cité pages 134, 154).

RÉFÉRENCES

O. ABDEL-HAMID, A. MOHAMED, H. JIANG et G. PENN. “Applying Convolutional Neural Networks concepts to hybrid NN-HMM model for Speech Recognition”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2012), p. 4277-4280. DOI : 10.1109/icassp.2012.6288864 (cité page 95).

B. AGRAWAL, M. MÜLLER, M. RADFAR, S. CHOUDHARY, A. MOUCHTARIS et S. KUNZMANN. “Tie your embeddings down : Cross-modal latent spaces for end-to-end Spoken Language Understanding”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2020), p. 7157-7161. DOI : 10.1109/icassp43922.2022.9747759 (cité pages 83, 86).

M. AKBACAK, Y. GAO, L. GU et H. KUO. “Rapid transition to new spoken dialogue domains : Language Model training using knowledge from previous domain applications and web text resources”. 2005, p. 1873-1876. DOI : 10.21437/Interspeech.2005-590 (cité page 85).

C. AKKUS, L. CHU, V. DJAKOVIC et al. “Multimodal Deep Learning”. *CoRR* (2023). DOI : 10.48550/arXiv.2301.04856 (cité page 86).

J. ALLEN, M. MANSHADI, M. DZIKOVSKA et M. SWIFT. “Deep linguistic processing for spoken dialogue systems”. *Deep Linguistic Processing (DeepLP)* (2007), p. 49-56. URL : https://aclanthology.org/W07-1207 (cité page 74).

D. AMODEI, R. ANUBHAI, E. BATTENBERG et al. “Deep speech 2 : End-to-end Speech Recognition in English and Mandarin”. *International Conference on Machine Learning (ICML)* 48 (2016), p. 173-182. DOI : 10.48550/arXiv.1512.02595 (cité pages 20, 48, 72, 81).

A. ANASTASOPOULOS et D. CHIANG. “Tied multitask learning for neural speech translation”. *North American Chapter of the Association for Computational Linguistics (NAACL)* (2018), p. 82-91. DOI : 10.18653/v1/N18-1008 (cité page 87).

A. ANASTASOPOULOS, D. CHIANG et L. DUONG. “An unsupervised probability model for speech-to-translation alignment of low-resource languages”. *Empirical Methods in Natural Language Processing (EMNLP)* (2016), p. 1255-1263. DOI : 10.18653/v1/D16-1133 (cité page 87).

V. ANDRÉ et E. CANUT. “Mise à disposition de corpus oraux interactifs : Le projet TCOF (Traitement des Corpus Oraux en Français)”. *Pratiques : linguistique, littérature, didactique* (2010), p. 147-148. DOI : 10.4000/pratiques.1597 (cité page 100).

B. ARCAS. “Do Large Language Models understand us ?” : *Daedalus* 151.2 (2022), p. 183-197. DOI : 10.1162/daed_a_01909 (cité page 79).

R. ARDILA, M. BRANSON, J. DAVIS et al. “Common Voice : A massively-multilingual speech corpus”. *Language Resources and Evaluation Conference (LREC)* (2020), p. 4218-4222. DOI : 10.48550/arXiv.1912.06670 (cité pages 100, 101, 103, 108).

M. ARTETXE et H. SCHWENK. “Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond”. *Association for Computational Linguistics (ACL)* 7 (2019), p. 597-610. DOI : 10.48550/arXiv.1812.10464 (cité pages 88, 109).

A. BABU, C. WANG, A. TJANDRA et al. “XLS-R : Self-Supervised Cross-Lingual Speech Representation learning at scale”. *Interspeech* (2022). DOI : 10.48550/arXiv.2111.09296 (cité pages 20, 73, 103, 105, 109, 134, 139-141, 174).

A. BAEVSKI, M. AULI et A. MOHAMED. “Effectiveness of Self-Supervised pre-training for Speech Recognition”. *CoRR* (2019). DOI : 10.48550/arXiv.1911.03912 (cité page 110).

A. BAEVSKI, S. SCHNEIDER et M. AULI. “vq-wav2vec : Self-Supervised Learning of discrete speech representations”. *International Conference for Learning Representations (ICLR)* (2020). DOI : 10.48550/arXiv.1910.05453 (cité pages 85, 110).

A. BAEVSKI, H. ZHOU, A. MOHAMED et M. AULI. “wav2vec 2.0 : A framework for Self-Supervised Learning of speech representations”. *Advances in Neural Information Processing Systems (NeurIPS)* 1044 (2020), p. 12449-12460. DOI : 10.48550/arXiv.2006.11477 (cité pages 20, 73, 83, 95).

D. BAHDANAU, K. CHO et Y. BENGIO. “Neural Machine Translation by jointly learning to align and translate”. *International Conference for Learning Representations (ICLR)* (2015). DOI : 10.48550/arXiv.1409.0473 (cité pages 50, 78, 87).

D. BAHDANAU, J. CHOROWSKI, D. SERDYUK, P. BRAKEL et Y. BENGIO. “End-to-end attention-based large vocabulary Speech Recognition”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2016), p. 4945-4949. DOI : 10.1109/ICASSP.2016.7472618 (cité page 72).

C. BAKER, C. FILLMORE et J. LOWE. “The Berkeley FrameNet project”. *Association for Computational Linguistics (ACL)* (1998), p. 86-90. DOI : 10.3115/980845.980860 (cité page 68).

T. BÄNZIGER, M. MORTILLARO et K. SCHERER. “Introducing the Geneva Multimodal expression corpus for experimental research on emotion perception”. *Emotion* 12.5 (2011), p. 1161-1179. DOI : 10.1037/a0025827 (cité page 100).

A. BAPNA, C. CHERRY, Y. ZHANG et al. “mSLAM : Massively multilingual joint pre-training for speech and text”. *CoRR* (2022). DOI : 10.48550/arXiv.2202.01374 (cité page 87).

E. BASTIANELLI, A. VANZO, P. SWIETOJANSKI et V. RIESER. “SLURP : A Spoken Language Understanding Resource Package”. *Empirical Methods in Natural Language Processing (EMNLP)* (2020), p. 7252-7262. DOI : 10.18653/v1/2020.emnlp-main.588 (cité pages 22, 79).

L. BAUM. “An inequality and associated maximization technique in statistical estimation of probabilistic functions of a Markov process”. *Inequalities* (1972), p. 1-8. URL : https://api.semanticscholar.org/CorpusID:60804212 (cité page 70).

F. BECHET, B. MAZA, N. BIGOUROUX, T. BAZILLON, M. EL-BÈZE, R. DE MORI et E. ARBILLOT. “DECODA : A call-centre human-human spoken conversation corpus”. *Language Resources and Evaluation (LREC)* (2012), p. 1343-1347. URL : http://www.lrec-conf.org/proceedings/lrec2012/pdf/684_Paper.pdf (cité page 63).

F. BÉCHET. “Modèles numériques pour la Compréhension Automatique de la Parole”. (2007) (cité pages 19, 60).

F. BÉCHET et C. RAYMOND. “Is ATIS too shallow to go deeper for benchmarking Spoken Language Understanding models ?” : *Interspeech* (2018). URL : https://api.semanticscholar.org/CorpusID:52189372 (cité page 63).

F. BÉCHET et C. RAYMOND. “Benchmarking benchmarks : Introducing new automatic indicators for benchmarking Spoken Language Understanding corpora”. *Interspeech* (2019). DOI : 10.21437/interspeech.2019-3033 (cité pages 21, 118).

M. BEN JANNET, M. ADDA-DECKER, O. GALIBERT, J. KAHN et S. ROSSET. “ETER : a new metric for the evaluation of hierarchical Named Entity Recognition”. *Language Resources and Evaluation (LREC)* (2014). URL : http://www.lrec-conf.org/proceedings/lrec2014/pdf/960_Paper.pdf (cité page 62).

E. BENDER, T. GEBRU, A. MCMILLAN-MAJOR et S. SHMITCHELL. “On the dangers of stochastic parrots : Can Language Models be too big ?” : *ACM Conference on Fairness, Accountability, and Transparency (FAccT)* (2021), p. 610-623. DOI : 10.1145/3442188.3445922 (cité pages 79, 83).

Y. BENGIO, R. DUCHARME et P. VINCENT. “A neural probabilistic Language Model”. *Advances in Neural Information Processing Systems (NeurIPS)* 13 (2000). URL : https://dl.acm.org/doi/10.5555/944919.944966 (cité pages 69, 71, 77, 78).

Y. BENGIO, J. LOURADOUR, R. COLLOBERT et J. WESTON. “Curriculum learning”. *Journal of the American Podiatry Association* 60 (2009). DOI : 10.1145/1553374.1553380 (cité page 82).

Y. BENGIO, P. SIMARD et P. FRASCONI. “Learning long-term dependencies with gradient descent is difficult”. *Transactions on Neural Networks* 5.2 (1994), p. 157-166. DOI : 10.1109/72.279181 (cité page 45).

A. BÉRARD, L. BESACIER, A. KOCABIYIKOGLU et O. PIETQUIN. “End-to-end automatic speech translation of audiobooks”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2018), p. 6224-6228. DOI : 10.1109/ICASSP.2018.8461690 (cité page 87).

S. BHOSALE, I. SHEIKH, S. DUMPALA et S. KOPPARAPU. “End-to-end Spoken Language Understanding : Bootstrapping in low resource scenarios”. *Interspeech* (2019). DOI : 10.21437/interspeech.2019-2366 (cité pages 21, 81, 86).

H. BONNEAU-MAYNARD, C. AYACHE, F. BECHET et al. “Results of the French Evalda-Media evaluation campaign for literal understanding”. *Language Resources and Evaluation (LREC)* (2006). URL : https://aclanthology.org/L06-1385 (cité pages 20, 63, 100, 124).

H. BONNEAU-MAYNARD et F. LEFÈVRE. “A 2+1-level stochastic understanding model”. *Automatic Speech Recognition and Understanding (ASRU)* (2005). DOI : 10.1109/ASRU.2005.1566476 (cité page 76).

H. BONNEAU-MAYNARD, S. ROSSET, C. AYACHE, A. KUHN et D. MOSTEFA. “Semantic annotation of the French MEDIA dialog corpus”. *Interspeech* (2005). DOI : 10.21437/Interspeech.2005-312 (cité pages 21, 67, 68, 75, 116, 117, 119-121, 132, 135, 156, 174, 183, 186-190).

A. BOUCHEKIF. “Structuration automatique de documents audio”. (2016) (cité pages 19, 65).

H. BOULARD et C. WELLEKENS. “Multilayer perceptrons and Automatic Speech Recognition”. *International Joint Conference on Neural Networks (IJCNN)* 4 (1987), p. 407-126 (cité page 71).

H. BOURLARD et N. MORGAN. “A continuous Speech Recognition system embedding MLP into HMM”. *Advances in Neural Information Processing Systems (NeurIPS)* 2 (1989), p. 186-193. URL : https://proceedings.neurips.cc/paper_files/paper/1989/file/bcbe3365e6ac95ea2c0343a2395834dd-Paper.pdf (cité page 71).

H. BOURLARD et N. MORGAN. “Connectionist Speech Recognition”. *Springer Science* (1994). DOI : 10.1007/978-1-4615-3210-1 (cité page 72).

R. BRACHMAN. “On the epistemological status of semantic networks”. *Associative Networks* (1979), p. 3-50. DOI : 10.1016/B978-0-12-256380-5.50007-4 (cité page 67).

R. BRACHMAN et J. SCHMOLZE. “An overview of the KL-ONE knowledge representation system”. *Cognitive Science* 9.2 (1985), p. 171-216. DOI : 10.1016/S0364-0213(85)80014-8 (cité page 67).

S. BRANCA-ROSOFF, S. FLEURY, F. LEFEUVRE et M. PIRES. “Discours sur la ville. Corpus de français parlé parisien des années 2000”. (2009). URL : http://cfpp2000.univ-paris3.fr/CFPP2000.pdf (cité page 100).

E. BRIGHAM et R. MORROW. “The Fast Fourier Transform”. *Spectrum* 4.12 (1967), p. 63-70. DOI : 10.1109/MSPEC.1967.5217220 (cité page 91).

P. BROWN, J. COCKE, S. DELLA PIETRA, V. DELLA PIETRA, F. JELINEK, J. LAFFERTY, R. MERCER et P. ROOSSIN. “A statistical approach to Machine Translation”. *Computational Linguistics (COLING)* 16.2 (1990), p. 79-85. URL : https://api.semanticscholar.org/CorpusID:14386564 (cité page 87).

P. BROWN, V. DELLA PIETRA, S. DELLA PIETRA et R. MERCER. “The mathematics of statistical Machine Translation : Parameter estimation”. *Computational Linguistics (COLING)* 19.2 (1993), p. 263-311. URL : https://aclanthology.org/J93-2003.pdf (cité page 87).

T. BROWN, B. MANN, N. RYDER et al. “Language models are few-shot learners”. *Advances in Neural Information Processing Systems (NeurIPS)* 33 (2020), p. 1877-1901. DOI : 10.48550/arXiv.2005.14165 (cité pages 54, 79).

P. BUDZIANOWSKI, T. WEN, B. TSENG, I. CASANUEVA, S. ULTES, O. RAMADAN et M. GAI. “MultiWOZ - A large-scale multi-domain Wizard-of-Oz dataset for task-oriented dialogue modelling”. *Empirical Methods in Natural Language Processing (EMNLP)* (2018), p. 5016-5026. DOI : 10.18653/v1/D18-1547 (cité page 63).

M. BUNDSCHUS, M. DEJORI, M. STETTER, V. TRESP et H. KRIEGEL. “Extraction of semantic biomedical relations from text using conditional random fields”. *BMC Bioinformatics* 9 (2008). DOI : 10.1186/1471-2105-9-207 (cité page 76).

O. CATTAN, S. GHANNAY, C. SERVAN et S. ROSSET. “Étude comparative de modèles Transformers en compréhension de la parole en français”. *Journées d’Étude sur la Parole (JEP)* (2022). URL : https://hal.science/hal-03701654 (cité page 79).

A. CAUBRIERE, S. ROSSET, Y. ESTÈVE, A. LAURENT et E. MORIN. “Where are we in Named Entity Recognition from speech ?” : *Language Resources and Evaluation Conference (LREC)* (2020), p. 4514-4520. URL : https://aclanthology.org/2020.lrec-1.556 (cité page 81).

A. CAUBRIERE, N. TOMASHENKO, A. LAURENT, E. MORIN, N. CAMELIN et Y. ESTÈVE. “Curriculum-Based transfer learning for an effective end-to-end Spoken Language Understanding and domain portability”. (2019), p. 1198-1202. DOI : 10.21437/Interspeech.2019-1832 (cité pages 82, 126).

D. CER, Y. YANG, S. KONG et al. “Universal Sentence Encoder”. *CoRR* (2018). DOI : 10.48550/arXiv.1803.11175 (cité page 88).

W. CHAN, N. JAITLY, Q. LE et O. VINYALS. “Listen, attend and spell : A neural network for large vocabulary conversational Speech Recognition”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2016), p. 4960-4964. DOI : 10.1109/ICASSP.2016.7472621 (cité pages 71, 72).

W. CHAN, D. PARK, C. LEE, Y. ZHANG, Q. LE et M. NOROUZI. “SpeechStew : Simply mix all available Speech Recognition data to train one large neural network”. *CoRR* (2021). DOI : 10.48550/arXiv.2104.02133 (cité page 103).

S. CHEN et J. GOODMAN. “An empirical study of smoothing techniques for Language Modeling”. *Computer, Speech and Language* 13.4 (1999), p. 359-394. DOI : 10.1006/csla.1999.0128 (cité page 71).

W. CHEN, M. HASEGAWA-JOHNSON et N. CHEN. “Topic and keyword identification for low-resourced speech using cross-language transfer learning”. *Interspeech* (2018), p. 2047-2051. DOI : 10.21437/Interspeech.2018-1283 (cité pages 21, 86).

C. CHIU, T. SAINATH, Y. WU et al. “State-of-the-art Speech Recognition with sequence-to-sequence models”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2017), p. 4774-4778. DOI : 10.48550/arXiv.1712.01769 (cité page 72).

S. CHIU et B. CHEN. “Innovative Bert-Based reranking Language Models for Speech Recognition”. *Spoken Language Technology (SLT)* (2021), p. 266-271. DOI : 10.1109/SLT48900.2021.9383557 (cité page 78).

K. CHO, B. van MERRIENBOER, D. BAHDANAU et Y. BENGIO. “On the properties of Neural Machine Translation : Encoderdecoder approaches”. *Empirical Methods in Natural Language Processing (EMNLP)* (2014). DOI : 10.3115/v1/W14-4012 (cité page 87).

K. CHO, B. van MERRIENBOER, C. GULCEHRE, D. BAHDANAU, F. BOUGARES, H. SCHWENK et Y. BENGIO. “Learning phrase representations using RNN encoder-decoder for statistical Machine Translation”. *Empirical Methods in Natural Language Processing (EMNLP)* (2014), p. 1724-1734. DOI : 10.48550/arXiv.1406.1078 (cité pages 47, 49, 87, 109).

N. CHOMSKY. “Syntactic structures”. *De Gruyter* 4 (1957). DOI : 10.1515/9783112316009 (cité page 73).

N. CHOMSKY. “Aspects of the theory of syntax”. *MIT Press* 50 (1965). URL : https://www.jstor.org/stable/j.ctt17kk81z (cité page 73).

J. CHOROWSKI, D. BAHDANAU, K. CHO et Y. BENGIO. “End-to-end continuous Speech Recognition using attention-based Recurrent NN : First results”. *CoRR* (2014). DOI : 10.48550/arXiv.1412.1602 (cité page 72).

J. CHOROWSKI, D. BAHDANAU, D. SERDYUK, K. CHO et Y. BENGIO. “Attention-based models for Speech Recognition”. *Advances in Neural Information Processing Systems (NeurIPS)* 1 (2015), p. 577-585. DOI : 10.48550/arXiv.1506.07503 (cité page 72).

A. CHOWDHERY, S. NARANG, J. DEVLIN et al. “PaLM : Scaling Language Modeling with pathways”. *Journal of Machine Learning Research (JMLR)* 24.1 (2024). DOI : 10.5555/3648699.3648939 (cité page 79).

Y. CHUNG, Y. ZHANG, W. HAN, C. CHIU, J. QIN, R. PANG et Y. WU. “w2v-BERT : Combining contrastive learning and Masked Language Modeling for Self-Supervised speech pre-training”. *Automatic Speech Recognition and Understanding (ASRU)* (2021), p. 244-250. DOI : 10.48550/arXiv.2108.06209 (cité pages 87, 109).

Y. CHUNG, C. ZHU et M. ZENG. “SPLAT : SPeech-LAnguage joint pre-Training for Spoken Language Understanding”. *North American chapter of the Association for Computational Linguistics (NAACL)* (2021). DOI : 10.18653/V1/2F2021.NAACL-MAIN.152 (cité page 83).

A. CONNEAU, A. BAEVSKI, R. COLLOBERT, A. MOHAMED et M. AULI. “Unsupervised Cross-lingual representation Learning for Speech Recognition”. *Interspeech* (2020). DOI : 10.48550/arXiv.2006.13979 (cité pages 21, 85, 101, 109).

A. CONNEAU, K. KHANDELWAL, N. GOYAL et al. “Unsupervised Cross-lingual representation learning at scale”. *Association for Computational Linguistics (ACL)* (2020), p. 8440-8451. DOI : 10.18653/v1/2020.acl-main.747 (cité pages 20, 78, 84).

A. CONNEAU, D. KIELA, H. SCHWENK, L. BARRAULT et A. BORDES. “Supervised learning of universal sentence representations from natural language inference data”. *Empirical Methods in Natural Language Processing (EMNLP)* (2017), p. 670-680. DOI : 10.18653/v1/D17-1070 (cité page 88).

A. CONNEAU et G. LAMPLE. “Cross-lingual Language Model pretraining”. *Advances in Neural Information Processing Systems (NeurIPS)* (2019). DOI : 10.48550/arXiv.1901.07291 (cité pages 21, 78, 84).

A. CONNEAU, G. LAMPLE, R. RINOTT, A. WILLIAMS, S. BOWMAN, H. SCHWENK et V. STOYANOV. “XNLI : Evaluating Cross-lingual sentence representations”. *Empirical Methods in Natural Language Processing (EMNLP)* (2018). DOI : 10.18653/v1/2FD18-1269 (cité pages 20, 84).

G. DAHL, D. YU, L. DENG et A. ACERO. “Context-dependent pre-trained Deep Neural Networks for large-vocabulary Speech Recognition”. *Transactions on Audio, Speech and Language Processing (TASLP)* 20.1 (2012), p. 30-42. DOI : 10.1109/TASL.2011.2134090 (cité pages 71, 118).

K. DAVIS, R. BIDDULPH et S. BALASHEK. “Automatic recognition of spoken digits”. *Acoustical Society of America (ASA)* 24 (1952), p. 637-642. DOI : 10.1121/1.1906946 (cité page 69).

S. DAVIS et P. MERMELSTEIN. “Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences”. *Transactions on Acoustics, Speech and Signal Processing (TASSP)* 28.4 (1980), p. 357-366. DOI : 10.1109/tassp.1980.1163420 (cité pages 20, 79, 90-93).

R. DE MORI. “Computer models of speech using fuzzy algorithms”. *Plenum press* (1983). DOI : 10.1007/978-1-4613-3742-3 (cité page 73).

R. DE MORI. “Spoken language understanding : A survey”. *Automatic Speech Recognition and Understanding (ASRU)* (2007), p. 365-376. DOI : 10.1109/ASRU.2007.4430139 (cité pages 19, 60, 73).

A. DEMPSTER, N. LAIRD et D. RUBIN. “Maximum likelihood from incomplete data via the EM”. *Royal Statistical Society* 39.1 (1977), p. 1-38. DOI : 10.1111/j.2517-6161.1977.tb01600.x (cité page 70).

L. DENG, G. TUR, X. HE et D. HAKKANI-TUR. “Use of kernel deep convex networks and end-to-end learning for Spoken Language Understanding”. *Spoken Language Technology (SLT)* (2012), p. 210-215. DOI : 10.1109/SLT.2012.6424224 (cité page 77).

P. DENISOV et N. VU. “Leveraging multilingual Self-Supervised pretrained models for sequence-to-sequence end-to-end Spoken Language Understanding”. *Automatic Speech Recognition and*

Understanding (ASRU) (2023), p. 1-8. DOI : 10.1109/ASRU57964.2023.10389655 (cité pages 87, 165, 176, 181).

T. DESOT, F. PORTET et M. VACHER. “Towards end-to-end spoken intent recognition in smart home”. *Speech Technology and Human-Computer Dialogue (SpeD)* (2019). DOI : 10.1109/SPED.2019.8906584 (cité pages 19, 64).

T. DESOT, F. PORTET et M. VACHER. “Corpus generation for voice command in smart home and the effect of speech synthesis on end-to-end SLU”. *Language Resources and Evaluation Conference (LREC)* (2020), p. 6395-6404. URL : https://aclanthology.org/2020.lrec-1.786 (cité page 81).

J DEVLIN, M. CHANG, K. LEE et K. TOUTANOVA. “BERT : Pre-training of deep bidirectional Transformers for Language Understanding”. *North American chapter of the Association for Computational Linguistics (NAACL)* (2019). DOI : 10.48550/arXiv.1810.04805 (cité pages 20, 54, 73, 78, 84-86, 96, 102, 106, 109).

M. DINARELLI, N. KAPOOR, B. JABAIAN et L. BESACIER. “A data efficient end-to-end Spoken Language Understanding architecture”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2020), p. 8519-8523. DOI : 10.1109/ICASSP40776.2020.9054564 (cité pages 82, 125).

M. DINARELLI, M. NAGUIB et F. PORTET. “Toward low-cost end-to-end Spoken Language Understanding”. *CoRR* (2022). DOI : 10.48550/arXiv.2207.00352 (cité page 83).

M. DINARELLI et I. TELLIER. “Improving Recurrent Neural Networks for sequence labelling”. *CoRR* (2016). DOI : 10.48550/arXiv.1606.02555 (cité pages 21, 77, 119).

M. DINARELLI, V. VUKOTIC et C. RAYMOND. “Label-dependency coding in simple recurrent networks for Spoken Language Understanding”. *Interspeech* (2017). DOI : 10.21437/Interspeech.2017-1480 (cité page 77).

Q. DO et J. GASPERS. “Cross-lingual transfer learning with data selection for large-scale Spoken Language Understanding”. *Empirical Methods in Natural Language Processing (EMNLP) and International Joint Conference on Natural Language Processing (IJCNLP)* (2019), p. 1455-1460. DOI : 10.18653/v1/D19-1153 (cité page 86).

D. DONG, H. WU, W. HE, D. YU et H. WANG. “Multi-task learning for multiple language translation”. *Association for Computational Linguistics (ACL) and International Joint Conference on Natural Language Processing (IJCNLP)* (2015), p. 1723-1732. DOI : 10.3115/v1/P15-1166 (cité page 87).

L. DONG, S. XU et B. XU. “Speech-Transformer : A no-recurrence sequence-to-sequence model for Speech Recognition”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2018), p. 5884-5888. DOI : 10.1109/ICASSP.2018.8462506 (cité pages 71, 72).

B. DORR, P. JORDAN et J. BENOIT. “A survey of current paradigms in Machine Translation”. 49 (1999), p. 1-68. DOI : 10.1016/S0065-2458(08)60282-X (cité page 87).

J. DOWDING, J. GAWRON, D. APPELT, J. BEAR, L. CHERNY, R. MOORE et D. MORAN. “GEMINI : A natural language system for Spoken Language Understanding”. *Association for Computational Linguistics (ACL)* (1993), p. 54-61. DOI : 10.3115/981574.981582 (cité page 73).

T. DOZAT. “Incorporating Nesterov momentum into Adam”. *International Conference for Learning Representations (ICLR)* (2016). URL : https://api.semanticscholar.org/CorpusID:620137 (cité page 41).

J. DUCHI, E. HAZAN et Y. SINGER. “Adaptive subgradient methods for online learning and stochastic optimization”. *The Journal of Machine Learning Research* 12 (2011), p. 2121-2159. URL : https://dl.acm.org/doi/abs/10.5555/1953048.2021068 (cité page 40).

L. DUONG, A. ANASTASOPOULOS, D. CHIANG, S. BIRD et T. COHN. “An attentional model for speech translation without transcription”. *North American chapter of the Association for Computational Linguistics (NAACL)* (2016). DOI : 10.18653/v1/N16-1109 (cité page 87).

P. DUQUENNE, H. GONG et H. SCHWENK. “Multimodal and multilingual embeddings for large-scale speech mining”. *Advances in Neural Information Processing Systems (NeurIPS)* (2021). URL : https://api.semanticscholar.org/CorpusID:245011132 (cité page 109).

P. DUQUENNE, H. SCHWENK et B. SAGOT. “SONAR : Sentence-level multimOdal and language-Agnostic Representations”. *CoRR* (2023). DOI : 10.48550/arXiv.2308.11466 (cité pages 21, 109, 181).

J. ELMAN. “Finding structure in time”. *Cognitive Science* 14.2 (1990), p. 179-211. DOI : 10.1016/0364-0213(90)90002-E (cité page 77).

A. ERIGUCHI, Y. TSURUOKA et K. CHO. “Learning to parse and translate improves Neural Machine Translation”. *Association for Computational Linguistics (ACL)* (2017). DOI : 10.18653/v1/P17-2012 (cité page 87).

I. ESHKOL-TARAVELLA, O. BAUDE, D. MAUREL, L. HRIBA, C. DUGUA et I. TELLIER. “Un grand corpus oral disponible : Le corpus d’Orléans 1968-2012”. *Traitement Automatique des Langues (TAL)* 52.3 (2011), p. 17-46. URL : https://aclanthology.org/2011.tal-3.2 (cité page 100).

Y. ESTÈVE, T. BAZILLON, J. ANTOINE, F. BÉCHET et J. FARINAS. “The EPAC Corpus : Manual and automatic annotations of conversational speech in French broadcast news”. *Language Resources and Evaluation (LREC)* (2010). URL : https://aclanthology.org/L10-1442 (cité page 100).

Y. ESTÈVE, C. RAYMOND, F. BÉCHET et R. DE MORI. “Conceptual decoding for spoken dialog systems”. *Eurospeech* (2003). DOI : 10.21437/Eurospeech.2003-260 (cité page 81).

S. EVAIN, H. NGUYEN, H. LE et al. “LeBenchmark : A reproducible framework for assessing Self-Supervised representation Learning from speech”. *Interspeech* (2021). DOI : 10.48550/arXiv.2104.11462 (cité pages 20, 73, 83, 100, 130, 181, 183).

S. EVAIN, M. NGUYEN, H. LE et al. “Task agnostic and task specific Self-Supervised Learning from speech with LeBenchmark”. *Advances in Neural Information Processing Systems (NeurIPS)* (2021). URL : https://hal.science/hal-03407172 (cité page 83).

A. FAN, M. LEWIS et Y. DAUPHIN. “Hierarchical neural story generation”. *Association for Computational Linguistics (ACL)* 1 (2018), p. 889-898. DOI : 10.48550/arXiv.1805.04833 (cité page 57).

F. FENG, Y. YANG, D. CER, N. ARIVAZHAGAN et W. WANG. “Language-agnostic BERT Sentence Embedding”. *Association for Computational Linguistics (ACL)* (2022), p. 878-891. DOI : 10.48550/arXiv.2007.01852 (cité pages 20, 78, 88, 105, 109, 134, 139-141, 173).

R. FÉR, P. MATJKA, F. GRÉZL, O. PLCHOT, K. VESELÝ et J. ERNOCKÝ. “Multilingually trained bottleneck features in spoken language recognition”. *Computer Speech and Language* 46 (2017), p. 252-267. DOI : 10.1016/j.csl.2017.06.008 (cité pages 21, 85).

S. FERNÁNDEZ, A. GRAVES et J. SCHMIDHUBER. “Phoneme recognition in TIMIT with BLSTM-CTC”. *CoRR* (2008). DOI : 10.48550/arXiv.0804.3269 (cité page 35).

C. FILLMORE. “Frame semantics and the nature of language”. *Origins and Evolution of Language and Speech* 280.1 (1976), p. 20-32. DOI : 10.1111/j.1749-6632.1976.tb25467.x (cité page 68).

G. FORNEY. “The Viterbi algorithm”. *IEEE* 61.3 (1973), p. 268-278. DOI : 10.1109/PROC.1973.9030 (cité page 70).

F. GADET. “Les parlers jeunes dans lÎle-de-France multiculturelle Paris”. *Langue et Société* 163.1 (2017), p. 192-195. DOI : 10.4000/lidil.4854 (cité page 100).

M. GALES, K. KNILL, A. RAGNI et S. RATH. “Speech Recognition and keyword spotting for low-resource languages : BABEL project research at CUED”. *Spoken Language Technologies for Under-resourced Languages (SLTU)* (2014). URL : https://api.semanticscholar.org/CorpusID:7439227 (cité pages 101, 103).

O. GALIBERT, S. ROSSET, C. GROUIN, P. ZWEIGENBAUM et L. QUINTARD. “Structured and extended Named Entity evaluation in automatic speech transcriptions”. *IJCNLP* (2011), p. 518-526. URL : https://aclanthology.org/I11-1058 (cité page 61).

S. GALLIANO, G. GRAVIER et L. CHAUBARD. “The ESTER 2 evaluation campaign for the rich transcription of French radio broadcasts”. *Interspeech* (2009). DOI : 10.21437/Interspeech.2009-680 (cité pages 19, 62).

Y. GAO, L. GU et H. KUO. “Portability challenges in developing interactive dialogue systems”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* 5 (2005), p. 1017-1020. DOI : 10.1109/ICASSP.2005.1416479 (cité page 86).

J. GAUVAIN, L. LAMEL et G. ADDA. “The LIMSI broadcast news transcription system”. *Speech Communication (SPECOM)* 37.1 (2002), p. 89-108. DOI : 10.1016/S0167-6393(01)00061-9 (cité page 79).

J. GAUVAIN et C. LEE. “Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains”. *Transactions on Speech and Audio Processing (TSAP)* 2.2 (1994), p. 291-298. DOI : 10.1109/89.279278 (cité page 70).

S. GHANNAY. “Étude sur les representations continues de mots appliquées à la detection automatique des erreurs de reconnaissance de la parole”. (2017) (cité pages 20, 79).

S. GHANNAY, A. CAUBRIERE, Y. ESTÈVE, N. CAMELIN, E. SIMONNET, A. LAURENT et E. MORIN. “End-to-end named entity and semantic concept extraction from speech”. *2018 IEEE Spoken Language Technology Workshop (SLT)* (2018), p. 692-699. DOI : 10.1109/SLT.2018.8639513 (cité pages 20, 66, 81, 82, 125).

S. GHANNAY, C. SERVAN et S. ROSSET. “Neural Networks approaches focused on French Spoken Language Understanding : Application to the MEDIA Evaluation Task”. *Computational Linguistics (COLING)* (2020), p. 2722-2727. DOI : 10.18653/v1/2020.coling-main.245 (cité page 79).

B. GHORBANI, O. FIRAT, M. FREITAG, A. BAPNA, M. KRIKUN, X. GARCIA, C. CHELBA et C. CHERRY. “Scaling laws of neural Machine Translation”. *International Conference for Learning Representations (ICLR)* (2022). DOI : 10.48550/arXiv.2109.07740 (cité page 104).

A. GHOSHAL, P. SWIETOJANSKI et S. RENALS. “Multilingual training of Deep Neural Networks”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2013), p. 7319-7323. DOI : 10.1109/ICASSP.2013.6639084 (cité page 85).

X GLOROT et Y. BENGIO. “Understanding the difficulty of training deep feedforward neural networks”. *Artificial Intelligence and Statistics (AIStat)* 9 (2010), p. 249-256. URL : https://api.semanticscholar.org/CorpusID:5575601 (cité page 42).

Y. GONG. “Speech Recognition in noisy environments : A survey”. *Speech Communication (SPECOM)* 16.3 (1995), p. 261-291. DOI : 10.1016/0167-6393(94)00059-J (cité pages 20, 79).

A GORIN, B. PARKER, R. SACHS et J. WILPON. “How may I help you ?” : *Interactive Voice Technology for Telecommunications Applications (IVTTA)* (1996), p. 57-60. DOI : 10.1109/IVTTA.1996.552741 (cité page 64).

P. GOURNAY, O. LAHAIE et R. LEFEBVRE. “A Canadian French emotional speech dataset”. *Multimedia Systems (MMSys)* (2018), p. 399-402. DOI : 10.1145/3204949.3208121 (cité page 100).

A. GRAVES, S. FERNÁNDEZ, F. GOMEZ et J. SCHMIDHUBER. “Connectionist Temporal Classification : Labelling unsegmented sequence data with Recurrent Neural Networks”. *International Conference on Machine Learning (ICML)* (2006), p. 369-376. DOI : 10.1145/1143844.1143891 (cité pages 34, 87, 130, 136, 183).

A. GRAVES et N. JAITLY. “Towards end-to-end Speech Recognition with Recurrent Neural Networks”. *International Conference on Machine Learning (ICML)* (2014). DOI : 10.48550/arXiv.1701.02720 (cité page 72).

G. GRAVIER, G. ADDA, N. PAULSSON, M. CARRÉ, A. GIRAUDEL et O. GALIBERT. “The ETAPE corpus for the evaluation of speech-based TV content processing in the French language”. *Language Resources and Evaluation (LREC)* (2012), p. 114-118. URL : https://aclanthology.org/L12-1270 (cité pages 62, 68, 100, 181).

G. GRAVIER, J. BONASTRE, E. GEOFFROIS, S. GALLIANO, K. MCTAIT et K. CHOUKRI. “The ESTER evaluation campaign for the rich transcription of French broadcast news”. *Language Resources and Evaluation (LREC)* (2004). URL : http://www.lrec-conf.org/proceedings/lrec2004/pdf/672.pdf (cité pages 62, 181).

R. GRISHMAN et B. SUNDHEIM. “Message Understanding Conference-6 : A brief history”. *Computational Linguistics (COLING)* (1996). URL : https://aclanthology.org/C96-1079 (cité page 61).

C. GROUIN, S. ROSSET, P. ZWEIGENBAUM, K. FORT, O. GALIBERT et L. QUINTARD. “Proposal for an extension of traditional Named Entities : From guidelines to evaluation, an overview”. *Linguistic Annotation Workshop (LAW)* (2011), p. 92-100. URL : https://aclanthology.org/W11-0411 (cité pages 19, 62, 68).

C. GUINAUDEAU. “Structuration automatique de flux télévisuels”. (2011) (cité page 65).

A. GULATI, J. QIN, C. CHIU et al. “Conformer : Convolution-augmented Transformer for Speech Recognition”. *Interspeech* (2020). DOI : 10.48550/arXiv.2005.08100 (cité page 110).

P. HAGHANI, A. NARAYANAN, M. BACCHIANI et al. “From audio to semantics : Approaches to end-to-end Spoken Language Understanding”. *Spoken Language Technology (SLT)* (2018), p. 720-726. DOI : 10.1109/SLT.2018.8639043 (cité page 83).

S. HAHN, M. DINARELLI, C. RAYMOND et al. “Comparing stochastic approaches to Spoken Language Understanding in multiple languages”. *Transactions on Audio, Speech, and Language Processing (TASLP)* 19.6 (2011), p. 1569-1583. DOI : 10.1109/TASL.2010.2093520 (cité pages 21, 76, 77, 85, 119, 125, 126).

D. HAKKANI-TÜR, G. TÜR, A. CELIKYILMAZ, Y. CHEN, J. GAO, L. DENG et Y. WANG. “Multi-domain joint semantic frame parsing using bi-directional RNN-LSTM”. *Interspeech* (2016). DOI : 10.21437/Interspeech.2016-402 (cité page 77).

J. HAMMERSLEY et P. CLIFFORD. “Markov fields on finite graphs and lattices”. (1971). URL : https://api.semanticscholar.org/CorpusID:118635048 (cité page 76).

A. HANNUN, C. CASE, J. CASPER et al. “Deep Speech : Scaling up end-to-end Speech Recognition”. *CoRR* (2014). DOI : 10.48550/arXiv.1412.5567 (cité page 81).

M. HATMI, C. JACQUIN, E. MORIN et S. MEIGNIER. “Incorporating Named Entity Recognition into the speech transcription process”. *Interspeech* (2013). DOI : 10.21437/Interspeech.2013-588 (cité pages 20, 81).

K. HAYASHI, H. TSUKADA, K. SUDOH, K. DUH et S. YAMAMOTO. “Hierarchical phrase-based Machine Translation with word-based reordering model”. *Computational Linguistics (COLING)* 2 (2010), p. 439-446. URL : https://aclanthology.org/C10-1050.pdf (cité page 87).

W. HEERINGA. “The origin of the Afrikaans pronunciation : A comparison to West Germanic languages and Dutch dialects”. (2008). URL : https://api.semanticscholar.org/CorpusID:6020342 (cité page 84).

K. HEFFERNAN, O. ÇELEBI et H. SCHWENK. “Bitext mining using distilled sentence representations for low-resource languages”. *Association for Computational Linguistics (ACL)* (2022), p. 2101-2112. DOI : 10.48550/arXiv.2205.12654 (cité page 109).

C. HEMPHILL, J. GODFREY et G. DODDINGTON. “The ATIS spoken language systems pilot corpus”. *Speech and Natural Language* (1990). URL : https://aclanthology.org/H90-1021 (cité page 63).

H. HERMANSKY. “Perceptual Linear Predictive (PLP) analysis of speech”. *Acoustical Society of America (ASA)* 87 (1990), p. 1738-1752. DOI : 10.1121/1.399423 (cité page 91).

N. HERVÉ, V. PELLOIN, B. FAVRE, F. DARY, A. LAURENT, S. MEIGNIER et L. BESACIER. “Using ASR-generated text for spoken Language Modeling”. *BigScience* (2022), p. 17-25. DOI : 10.18653/v1/2022.bigscience-1.2 (cité pages 78, 185).

G. HINTON, L. DENG, D. YU et al. “Deep Neural Networks for acoustic modeling in Speech Recognition : The shared views of four research groups”. *Signal Processing Magazine* 29.6 (2012), p. 82-97. DOI : 10.1109/MSP.2012.2205597 (cité page 71).

S. HOCHREITER et J. SCHMIDHUBER. “Long Short-Term Memory”. *Neural Computation* 9.8 (1997), p. 1735-1780. DOI : 10.1162/neco.1997.9.8.1735 (cité page 45).

Ari HOLTZMAN, J. BUYS, L. DU, M. FORBES et Y. CHOI. “The curious case of neural text degeneration”. *International Conference for Learning Representations (ICLR)* (2020). DOI : 10.48550/arXiv.1904.09751 (cité page 57).

T. HORI, J. CHO et S. WATANABE. “End-to-end Speech Recognition with word-based RNN Language Models”. *Spoken Language Technology (SLT)* (2018), p. 389-396. DOI : 10.1109/SLT.2018.8639693 (cité page 72).

T. HORI et A. NAKAMURA. “An extremely large vocabulary approach to Named Entity extraction from speech”. *International Conference on Acoustics Speech and Signal Processing (ICASSP)* (2006). DOI : 10.1109/ICASSP.2006.1660185 (cité page 81).

T. HORI, S. WATANABE et J. HERSHEY. “Multi-level Language Modeling and decoding for open vocabulary end-to-end Speech Recognition”. *Automatic Speech Recognition and Understanding (ASRU)* (2017), p. 287-293. DOI : 10.1109/ASRU.2017.8268948 (cité page 70).

T. HORI, S. WATANABE, Y. ZHANG et W. CHAN. “Advances in joint CTC-Attention based end-to-end Speech Recognition with a deep CNN encoder and RNN-LM”. *CoRR* (2017). DOI : 10.48550/arXiv.1706.02737 (cité pages 69, 72).

W. HSU, B. BOLTE, Y. TSAI, K. LAKHOTIA, R. SALAKHUTDINOV et A. MOHAMED. “HuBERT : Self-supervised speech representation learning by masked prediction of hidden units”. *Transactions on Audio, Speech and Language Processing (TASLP)* 29 (2021), p. 3451-3460. DOI : 10.48550/arXiv.2106.07447 (cité pages 73, 83, 110).

J. HU, S. RUDER, A. SIDDHANT, G. NEUBIG, O. FIRAT et M. JOHNSON. “XTREME : A massively multilingual multi-task benchmark for evaluating cross-lingual generalisation”. *International Conference on Machine Learning (ICML)* (2020). DOI : 10.48550/arXiv.2003.11080 (cité page 84).

J. HUANG, J. LI, D. YU, L. DENG et Y. GONG. “Cross-language knowledge transfer using multilingual Deep Neural Network with shared hidden layers”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2013), p. 7304-7308. DOI : 10.1109/ICASSP.2013.6639081 (cité pages 21, 85).

Y. HUANG, H. KUO, S. THOMAS, Z. KONS, K. AUDHKHASI, B. KINGSBURY, R. HOORY et M. PICHENY. “Leveraging unpaired text data for training end-to-end speech-to-intent systems”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2020), p. 7984-7988. DOI : 10.1109/ICASSP40776.2020.9053281 (cité pages 81, 87).

D. IMSENG, P. MOTLICEK, H. BOURLARD et P. GARNER. “Using out-of-language data to improve an under-resourced Speech Recognizer”. *Speech Communication (SPECOM)* 56 (2014), p. 142-151. DOI : 10.1016/j.specom.2013.01.007 (cité page 84).

B. JABAIAN. “Systèmes de compréhension et de traduction de la parole : Vers une approche unifiée dans le cadre de la portabilité multilingue des systèmes de dialogue”. (2012) (cité pages 19, 63, 117).

B. JABAIAN, L. BESACIER et F. LEFÈVRE. “Comparison and combination of lightly supervised approaches for language portability of a Spoken Language Understanding system”. *Transactions on Audio, Speech, and Language Processing (TASLP)* 21.3 (2013), p. 636-648. DOI : 10.1109/TASL.2012.2229983 (cité pages 21, 86).

F. JELINEK. “Continuous speech recognition by statistical methods”. *IEEE* 64.4 (1976), p. 532-556. DOI : 10.1109/PROC.1976.10159 (cité pages 69, 70).

X. JIA, J. WANG, Z. ZHANG, N. CHENG et J. XIAO. “Large-scale transfer learning for low-resource Spoken Language Understanding”. *Interspeech* (2020). DOI : 10.21437/interspeech.2020-0059 (cité pages 21, 86).

T. JOACHIMS. “Text categorization with support vector machines : Learning with many relevant features”. *European Conference on Machine Learning (ECML)* (1998), p. 137-142. DOI : 10.1007/BFb0026683 (cité page 75).

M. JOHNSON, M. SCHUSTER, Q. LE et al. “Google’s multilingual Neural Machine Translation system : Enabling zero-shot translation”. *Association for Computational Linguistics (ACL)* 5 (2017), p. 339-351. DOI : 10.1162/tacl_a_00065 (cité page 85).

M. JORDAN. “Attractor dynamics and parallelism in a connectionist sequential machine”. *Artificial neural networks : concept learning* (1990), p. 112-127. URL : https://dl.acm.org/doi/10.5555/104134.104148 (cité page 44).

M. JORDAN. “Serial Order : A parallel distributed processing approach”. *Neural Network Models of Cognition* 121 (1997), p. 471-495. DOI : 10.1016/S0166-4115(97)80111-2 (cité page 77).

M. JOSHI, D. CHEN, Y. LIU, D. WELD, L. ZETTLEMOYER et O. LEVY. “SpanBERT : Improving pre-training by representing and predicting spans”. *Transactions of the Association for Computational Linguistics (TACL)* 8 (2020), p. 64-77. DOI : 10.1162/tacl_a_00300 (cité page 87).

R. KADARI, Y. ZHANG, W. ZHANG et T. LIU. “CCG supertagging via bidirectional LSTM-CRF neural architecture”. *Neurocomputing* 283 (2018), p. 31-37. DOI : 10.1016/j.neucom.2017.12.050 (cité page 78).

J. KAHN, M. RIVIÈRE, W. ZHENG et al. “Libri-Light : A benchmark for ASR with limited or no supervision”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2020), p. 7669-7673. DOI : 10.48550/arXiv.1912.07875 (cité pages 54, 110).

R. KAPLAN et J. BRESNAN. “Lexical-functional grammar : A formal system for grammatical representation”. *The Mental Representation of Grammatical Relations* (1982), p. 173-281 (cité page 87).

S. KATZ. “Estimation of probabilities from sparse data for the Language Model component of a speech recognizer”. *Transactions on Acoustics, Speech and Signal Processing (TASSP)* 35.3 (1987), p. 400-401. DOI : 10.1109/TASSP.1987.1165125 (cité page 71).

K. KAWAKAMI, L. WANG, C. DYER, P. BLUNSOM et A. van den OORD. “Learning robust and multilingual speech representations”. *Empirical Methods in Natural Language Processing (EMNLP)* (2020), p. 1182-1192. DOI : 10.18653/v1/2020.findings-emnlp.106 (cité pages 21, 85).

J. KELLEY. “An iterative design methodology for user-friendly natural language office information applications”. *ACM Transactions on Information Systems (TOIS)* 2.1 (1984), p. 26-41. DOI : 10.1145/357417.357420 (cité page 117).

S. KHURANA, A. LAURENT et J. GLASS. “SAMU-XLSR : Semantically-Aligned Multimodal Utterance-level Cross-Lingual Speech Representation”. *Journal of Selected Topics in Signal Processing* 16.6 (2022), p. 1493-1504. DOI : 10.48550/arXiv.2205.08180 (cité pages 20, 22, 88, 105, 108, 109, 134, 135, 139-141, 144, 156, 173, 185).

S. KIM, T. HORI et S. WATANABE. “Joint CTC-attention based end-to-end Speech Recognition using multi-task learning”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2017), p. 4835-4839. DOI : 10.1109/ICASSP.2017.7953075 (cité page 72).

D. KINGMA et J. BA. “Adam : A method for stochastic optimization”. *International Conference for Learning Representations (ICLR)* (2015). DOI : 10.48550/arXiv.1412.6980 (cité page 41).

D. KLATT. “Review of the ARPA speech understanding project”. *Acoustical Society of America (ASA)* 62 (1977), p. 1345-1366. DOI : 10.1121/1.381666 (cité page 73).

T. KLEINBAUER, S. BECKER et T. BECKER. “Combining multiple information layers for the automatic generation of indicative meeting abstracts”. *European Workshop on Natural Language Generation (ENLG)* (2007), p. 151-154. URL : https://aclanthology.org/W07-2324 (cité page 65).

K. KOMATANI, K. TANAKA, H. KASHIMA et T. KAWAHARA. “Domain-independent spoken dialogue platform using key-phrase spotting based on combined Language Model”. *Eurospeech* (2001), p. 1319-1322. DOI : 10.21437/Eurospeech.2001-341 (cité pages 21, 85).

M. KORPUSIK, Z. LIU et J. GLASS. “A comparison of Deep Learning methods for language understanding”. *Interspeech* (2019), p. 849-853. DOI : 10.21437/Interspeech.2019-1262 (cité page 79).

S. KUDUGUNTA, A. BAPNA, I. CASWELL, N. ARIVAZHAGAN et O. FIRAT. “Investigating multilingual NMT representations at scale”. *Empirical Methods in Natural Language Processing (EMNLP)* (2019). DOI : 10.18653/v1/D19-1167 (cité page 87).

J. LAFFERTY, A. MCCALLUM et F. PEREIRA. “Conditional Random Fields : Probabilistic models for segmenting and labeling sequence data”. *International Conference on Machine Learning (ICLM)* (2001), p. 282-289. DOI : 10.5555/645530.655813 (cité page 76).

G. LAMPLE et A. CONNEAU. “Cross-lingual Language Model pretraining”. *Advances in Neural Information Processing Systems (NeurIPS)* 634 (2019), p. 7059-7069. DOI : 10.48550/arXiv.1901.07291 (cité pages 73, 102, 106).

T. LANDAUER, C. KAMM et S. SINGHAL. “Learning a minimally structured back propagation network to recognize speech”. *Cognitive Science Society (CogSci)* (1987), p. 531-536 (cité page 72).

P. LANGLAIS, F. YVON et P. ZWEIGENBAUM. “Analogical translation of medical words in different languages”. *Advances in Natural Language Processing (GoTAL)* (2008). DOI : 10.1007/978-3-540-85287-2_27 (cité page 87).

H. LE, L. VIAL, J. FREJ et al. “FlauBERT : Unsupervised Language Model pre-training for French”. *Language Resources and Evaluation Conference (LREC)* (2020), p. 2479-2490. DOI : 10.48550/arXiv.1912.05372 (cité page 78).

C. LE MOINE et N. OBIN. “Att-HACK : An expressive speech database with social attitudes”. *Speech Prosody* (2020). DOI : 10.48550/arXiv.2004.04410 (cité page 100).

T. LE SCAO, T. WANG, D. HESSLOW et al. “What Language Model to train if you have one Million GPU hours ?” : *Empirical Methods in Natural Language Processing (EMNLP)* (2022), p. 765-782. DOI : 10.48550/arXiv.2210.15424 (cité page 79).

Y. LECUN, B. BOSER, J. DENKER, D. HENDERSON, R. HOWARD, W. HUBBARD et L. JACKEL. “Handwritten digit recognition with a back-propagation network”. *Advances in Neural Information Processing Systems (NeurIPS)* (1989). URL : https://api.semanticscholar.org/CorpusID:2542741 (cité page 47).

F. LEFÈVRE. “Dynamic bayesian networks and discriminative classifiers for multi-stage semantic interpretation”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* 4 (2007). DOI : 10.1109/ICASSP.2007.367151 (cité page 77).

F. LEFÈVRE, J. GAUVAIN et L. LAMEL. “Genericity and portability for task-independent speech recognition”. *Computer Speech and Language* 19.3 (2005), p. 345-363. DOI : 10.1016/j.csl.2004.11.001 (cité pages 21, 85).

F. LEFÈVRE, D. MOSTEFA, L. BESACIER et al. “Robustesse et portabilités multilingue et multidomaines des systèmes de compréhension de la parole : Les corpus du projet PortMedia”. *JEP-TALN-RECITAL* 1 (2012), p. 779-786. URL : https://aclanthology.org/F12-1098 (cité pages 21, 63, 100, 101, 116, 117, 119, 120, 122-124, 132, 135, 156, 174, 178).

J. LEHUEN et T. LEMEUNIER. “A Robust Semantic Parser Designed for Spoken Dialog Sytems”. *International Conference on Semantic Computing (ICSC)* (2010), p. 52-55. DOI : 10.1109/ICSC.2010.22 (cité page 124).

V. LEVENSHTEIN. “Binary codes capable of correcting deletions, insertions, and reversals”. *Soviet Physics Doklady* 10 (1965), p. 707-710. URL : https://api.semanticscholar.org/CorpusID:60827152 (cité page 125).

T. LIKHOMANENKO, Q. XU, V. PRATAP, P. TOMASELLO, J. KAHN, G. AVIDOV, R. COLLOBERT et G. SYNNAEVE. “Rethinking evaluation in ASR : Are our models robust enough ?” : *Interspeech* (2020). DOI : 10.48550/arXiv.2010.11745 (cité page 103).

H. LIN, L. DENG, D. YU, Y. GONG, A. ACERO et C. LEE. “A study on multilingual acoustic modeling for large vocabulary ASR”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2009), p. 4333-4336. DOI : 10.1109/ICASSP.2009.4960588 (cité pages 21, 84).

A. LIU, S. LI et H. LEE. “TERA : Self-Supervised Learning of Transformer encoder representation for speech”. *Transactions on Audio, Speech and Language Processing (TASLP)* 29 (2021), p. 2351-2366. DOI : 10.1109/TASLP.2021.3095662 (cité page 83).

A. LIU, S. YANG, P. CHI, P. HSU et H. LEE. “Mockingjay : Unsupervised speech representation learning with deep bidirectional Transformer encoders”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2019), p. 6419-6423. DOI : 10.1109/ICASSP40776.2020.9054458 (cité page 72).

B. LIU et I. LANE. “Attention-based Recurrent Neural Network models for joint intent detection and slot filling”. *CoRR* (2016). DOI : 10.48550/arXiv.1609.01454 (cité pages 19, 64, 78).

Y. LIU, J. GU, N. GOYAL, X. LI, S. EDUNOV, M. GHAZVININEJAD, M. LEWIS et L. ZETTLEMOYER. “Multilingual denoising pre-training for neural Machine Translation”. *Association for Computational Linguistics (ACL)* 8 (2020), p. 726-742. DOI : 10.48550/arXiv.2001.08210 (cité pages 87, 110, 181).

Y. LIU, M. OTT, N. GOYAL et al. “RoBERTa : A Robustly optimized BERT pretraining approach”. *CoRR* (2019). DOI : 10.48550/arXiv.1907.11692 (cité page 78).

I. LOSHCHILOV et F. HUTTER. “Decoupled weight decay regularization”. *International Conference for Learning Representations (ICLR)* (2019). DOI : 10.48550/arXiv.1711.05101 (cité page 41).

L. LU, A. GHOSHAL et S. RENALS. “Cross-Lingual subspace Gaussian Mixture Models for low-resource Speech Recognition”. *Transactions on Audio, Speech, and Language Processing (TASLP)* 22.1 (2014), p. 17-27. DOI : 10.1109/TASL.2013.2281575 (cité page 84).

L. LUGOSCH, B. MEYER, D. NOWROUZEZAHRAI et M. RAVANELLI. “Using speech synthesis to train end-to-end Spoken Language Understanding models”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2020), p. 8499-8503. DOI : 10.1109/ICASSP40776.2020.9053063 (cité page 81).

L. LUGOSCH, M. RAVANELLI, P. IGNOTO, V. TOMAR et Y. BENGIO. “Speech model pre-training for end-to-end Spoken Language Understanding”. *CoRR* (2019). DOI : 10.21437/interspeech.2019-2396 (cité pages 21, 82, 86).

H. LUHN. “The automatic creation of literature abstracts”. *IBM Journal of Research and Development* 2.2 (1958), p. 159-165. DOI : 10.1147/rd.22.0159 (cité page 65).

M. LUONG, Q. LE, I. SUTSKEVER, O. VINYALS et L. KAISER. “Multi-task sequence-to-sequence learning”. *International Conference for Learning Representations (ICLR)* (2016). DOI : 10.48550/arXiv.1511.06114 (cité page 87).

T. LUONG, H. PHAM et C. MANNING. “Effective approaches to attention-based neural Machine Translation”. *Empirical Methods in Natural Language Processing (EMNLP)* (2015), p. 1412-1421. DOI : 10.48550/arXiv.1508.04025 (cité page 50).

W. MA et D. VAN COMPERNOLLE. “TDNN labeling for a HMM recognizer”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (1990), p. 421-423. URL : https://api.semanticscholar.org/CorpusID:61931966 (cité page 71).

X. MA et E. HOVY. “End-to-end sequence labeling via bi-directional LSTM-CNNs-CRF”. *Association for Computational Linguistics (ACL)* (2016), p. 1064-1074. DOI : 10.18653/v1/P16-1101 (cité page 81).

M. MACARY, M. TAHON, Y. ESTÈVE et A. ROUSSEAU. “AlloSat : A new call center French corpus for satisfaction and frustration analysis”. *Language Resources and Evaluation Conference (LREC)* (2020), p. 1590-1597. URL : https://aclanthology.org/2020.lrec-1.197 (cité page 100).

J. MAKHOUL, F. KUBALA, R. SCHWARTZ et R. WEISCHEDEL. “Performance measures for information extraction”. *CoRR* (2007). URL : https://api.semanticscholar.org/CorpusID:15827348 (cité page 62).

R. MALOUF. “A comparison of algorithms for maximum entropy parameter estimation”. *Computational Natural Language Learning (CoNLL)* (2002). URL : https://api.semanticscholar.org/CorpusID:6249194 (cité page 76).

J. MARKEL et A. GRAY. “Linear prediction of speech”. (1976) (cité page 91).

L. MARTIN, B. MULLER, P. ORTIZ SUAREZ, Y. DUPONT, L. ROMARY, E. VILLEMONTE DE LA CLERGERIE, D. SEDDAH et B. SAGOT. “CamemBERT : A tasty French Language Model”. *Association for Computational Linguistics (ACL)* (2019). DOI : 10.18653/v1/2020.acl-main.645 (cité pages 78, 183).

S. MASKEY et J. HIRSCHBERG. “Automatic summarization of broadcast news using structural features”. *Interspeech* (2003). URL : https://api.semanticscholar.org/CorpusID:2438969 (cité pages 19, 65).

A. MASMOUDI, M. KHMEKHEM, Y. ESTÈVE, L. BELGUITH et N. HABASH. “A corpus and phonetic dictionary for Tunisian Arabic Speech Recognition”. *Language Resources and Evaluation (LREC)* (2014), p. 306-310. URL : http://www.lrec-conf.org/proceedings/lrec2014/pdf/454_Paper.pdf (cité page 140).

E. MATUSOV, S. KANTHAK et H. NEY. “On the integration of Speech Recognition and statistical Machine Translation”. *Interspeech* (2005). DOI : 10.21437/Interspeech.2005-726 (cité page 87).

A. MCCALLUM et W. LI. “Early results for Named Entity Recognition with Conditional Random Fields, feature induction and web-enhanced lexicons”. *North American chapter of the Association*

for Computational Linguistics (NAACL) (2003), p. 188-191. URL : https://aclanthology.org/W03-0430 (cité page 77).

W. MCCULLOCH et W. PITTS. “A logical calculus of the ideas immanent in nervous activity”. *Bulletin of Mathematical Biophysics* 5.4 (1943), p. 115-133. DOI : 10.1007/BF02478259 (cité page 31).

S. MDHAFFAR, F. BOUGARES, R. DE MORI, S. ZAIEM, M. RAVANELLI et Y. ESTÈVE. “TARIC-SLU : A Tunisian benchmark dataset for Spoken Language Understanding”. *Language Resources and Evaluation (LREC) and Computational Linguistics (COLING* (2024). URL : https://aclanthology.org/2024.lrec-main.1357/ (cité pages 23, 63, 116, 132, 135, 138, 141, 156, 157, 174).

S. MDHAFFAR, A. LAURENT et Y. ESTÈVE. “Le corpus PASTEL pour le traitement automatique de cours magistraux”. *Traitement Automatique du Langage Natuel (TALN)* (2018), p. 419-426. URL : https://aclanthology.org/2018.jeptalnrecital-court.25 (cité pages 19, 65).

H. MENG et K. SIU. “Semiautomatic acquisition of semantic structures for understanding domain-specific natural language queries”. *Transactions on Knowledge and Data Engineering* 14 (2002), p. 172-181. DOI : 10.1109/69.979980 (cité pages 21, 85).

G. MESNIL, Y. DAUPHIN, K. YAO et al. “Using Recurrent Neural Networks for slot filling in Spoken Language Understanding”. *Transactions on Audio, Speech and Language Processing (TASLP)* 23.3 (2015), p. 530-539. DOI : 10.1109/TASLP.2014.2383614 (cité pages 19, 63, 77).

G. MESNIL, X. HE, L. DENG et Y. BENGIO. “Investigation of Recurrent Neural Network architectures and learning methods for Spoken Language Understanding”. *Interspeech* (2013). DOI : 10.21437/Interspeech.2013-596 (cité page 77).

M-J. MEURS. “Approche stochastique bayésienne de la composition sémantique pour les modules de compréhension automatique de la parole dans les systèmes de dialogue humain-machine”. (2009) (cité page 68).

Y. MIAO et F. METZE. “Improving low-resource CD-DNN-HMM using dropout and multilingual DNN training”. *Interspeech* (2013). DOI : 10.21437/Interspeech.2013-526 (cité page 85).

T. MIKOLOV, K. CHEN, G. CORRADO et J. DEAN. “Efficient estimation of word representations in vector space”. *International Conference on Learning Representations (ICLR)* (2013). DOI : 10.48550/arXiv.1301.3781 (cité pages 72, 77).

T. MIKOLOV, M. KARAFIÁT, L. BURGET, J. ERNOCKÝ et S. KHUDANPUR. “Recurrent Neural Network based Language Model”. *Interspeech* (2010). DOI : 10.21437/Interspeech.2010-343 (cité page 78).

T. MIKOLOV, S. KOMBRINK, L. BURGET, J. ERNOCKÝ et S. KHUDANPUR. “Extensions of Recurrent Neural Network Language Model”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2011), p. 5528-5531. DOI : 10.1109/ICASSP.2011.5947611 (cité page 78).

T. MIKOLOV, I. SUTSKEVER, K. CHEN, G. CORRADO et J. DEAN. “Distributed representations of words and phrases and their compositionality”. *Advances in Neural Information Processing Systems (NeurIPS)* (2013). DOI : 10.48550/arXiv.1310.4546 (cité page 72).

M. MINSKY et S. PAPERT. “Perceptrons : An introduction to computational geometry”. *MIT Press* (1969) (cité page 32).

M. MITCHELL et D. KRAKAUER. “The debate over understanding in AIs Large Language Models”. *Proceedings of the National Academy of Sciences (PNAS)* 120 (2022). URL : 10.1073/pnas.2215907120 (cité page 79).

M. MOHRI et Nederhof J. “Regular approximation of context-free grammars through transformation”. *Springer Science* 17 (2001), p. 153-163. DOI : 10.1007/978-94-015-9719-7_6 (cité page 74).

N. MORITZ, T. HORI et J. LE. “Streaming Automatic Speech Recognition with the Transformer model”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2020), p. 6074-6078. DOI : 10.1109/ICASSP40776.2020.9054476 (cité page 72).

A. MOUMEN et T. PARCOLLET. “Stabilising and accelerating Light Gated Recurrent Units for Automatic Speech Recognition”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2023). DOI : 10.48550/arXiv.2302.10144 (cité pages 47, 136, 180).

M. MÜLLER, S. CHOUDHARY, C. CHUNG, A. MOUCHTARIS et S. KUNZMANN. “In pursuit of Babel - Multilingual and-to-and Spoken Language Understanding”. *Automatic Speech Recognition and Understanding (ASRU)* (2021), p. 1042-1049. DOI : 10.1109/ASRU51503.2021.9688263 (cité pages 21, 83, 85).

G. MURRAY, G. CARENINI et R. NG. “Interpretation and transformation for abstracting conversations”. *North American chapter of the Association for Computational Linguistics (NAACL)* (2010), p. 894-902. URL : https://aclanthology.org/N10-1132 (cité pages 19, 65).

V. NAIR et G. HINTON. “Rectified linear units improve restricted Boltzmann machines”. *International Conference on Machine Learning (ICML)* (2010), p. 807-814. URL : https://dl.acm.org/doi/10.5555/3104322.3104425 (cité page 33).

L. NARAYANA et S. KOPPARAPU. “Choice of Mel filter bank in computing MFCC of a resampled speech”. *CoRR* (2014). DOI : 10.48550/arXiv.1410.6903 (cité page 94).

A. NARAYANAN, A. MISRA, K. SIM et al. “Toward domain-invariant Speech Recognition via large scale training”. *Spoken Language Technology Workshop (SLT)* (2018), p. 441-447. DOI : 10.48550/arXiv.1808.05312 (cité page 103).

Y. NESTEROV. “Gradient methods for minimizing composite functions”. *Mathematical Programming* 140 (2013), p. 125-161. DOI : 10.48550/arXiv.2203.07318 (cité page 39).

H. NEY. “Speech translation : Coupling of recognition and translation”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (1999), p. 517-520. DOI : 10.1109/ICASSP.1999.758176 (cité page 87).

J. NIEKRASZ et J. MOORE. “Participant subjectivity and involvement as a basis for discourse segmentation”. *Special Interest Group on Discourse and Dialogue (SIGDIAL)* (2009). DOI : 10.3115/1708376.1708384 (cité page 65).

NLLB-TEAM. “No Language Left Behind : Scaling human-centered Machine Translation”. *ArXiv* (2022). DOI : 10.48550/arXiv.2207.04672 (cité pages 87, 88, 109, 181).

D. NOUVEL, Ehrmann M. et S. ROSSET. “Les entités nommées pour le traitement automatique des langues”. *ISTE Editions* (2015) (cité pages 19, 62, 82).

E. PALOGIANNIDI, I. GKINIS, G. MASTRAPAS, P. MIZERA et T. STAFYLAKIS. “End-to-End architectures for ASR-free Spoken Language Understanding”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2020), p. 7974-7978. DOI : 10.1109/ICASSP40776.2020.9054314 (cité page 82).

S. PAN et Q. YANG. “A survey on transfer learning”. *Transactions on Knowledge and Data Engineering* 22.10 (2010), p. 1345-1359. DOI : 10.1109/tkde.2009.191 (cité page 43).

T. PARCOLLET, H. NGUYEN, S. EVAIN et al. “LeBenchmark 2.0 : A standardized, replicable and enhanced framework for Self-Supervised representations of French speech”. *Computer Science and Language* 86 (2024). DOI : 10.48550/arXiv.2309.05472 (cité page 101).

T. PARCOLLET et M. RAVANELLI. “The energy and carbon footprint of training end-to-end speech recognizers”. *Interspeech* (2021). DOI : 10.21437/interspeech.2021-456 (cité page 83).

D. PARK, W. CHAN, Y. ZHANG, C. CHIU, B. ZOPH, E. CUBUK et Q. LE. “SpecAugment : A simple data augmentation method for Automatic Speech Recognition”. *Interspeech* (2019). DOI : 10.21437/Interspeech.2019-2680 (cité page 71).

S. PASCUAL, M. RAVANELLI, J. SERRÀ, A. BONAFONTE et Y. BENGIO. “Learning problem-agnostic speech representations from multiple Self-Supervised tasks”. *CoRR* (2019). DOI : 10.48550/arXiv.1904.03416 (cité page 73).

R. PASSONNEAU et D. LITMAN. “Discourse segmentation by human and automated means”. *Computational Linguistics (COLING)* 23.1 (1997), p. 103-139. URL : https://aclanthology.org/J97-1005 (cité page 65).

V. PEDDINTI, D. POVEY et S. KHUDANPUR. “A Time Delay Neural Network architecture for efficient modeling of long temporal contexts”. *Interspeech* (2015), p. 3214-3218. DOI : 10.21437/Interspeech.2015-647 (cité page 48).

V. PELLOIN, N. CAMELIN, A. LAURENT, R. DE MORI, A. CAUBRIERE, Y. ESTÈVE et S. MEIGNIER. “End2End acoustic to semantic transduction”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2021), p. 7448-7452. DOI : 10.1109/ICASSP39728.2021.9413581 (cité pages 82, 126).

V. PELLOIN, F. DARY, N. HERVÉ, B. FAVRE, N. CAMELIN, A. LAURENT et L. BESACIER. “ASR-generated text for Language Model pre-training applied to speech tasks”. *Interspeech* (2022). DOI : 10.48550/arXiv.2207.01893 (cité page 78).

J. PENNINGTON, R. SOCHER et C. MANNING. “GloVe : Global Vectors for word representation”. *Empirical Methods in Natural Language Processing (EMNLP)* (2014), p. 1532-1543. DOI : 10.3115/v1/D14-1162 (cité pages 72, 77).

M. PETERS, M. NEUMANN, M. IYYER, M. GARDNER, C. CLARK, K. LEE et L. ZETTLEMOYER. “Deep contextualized word representations”. *North American chapter of the Association for Computational Linguistics (NAACL)* 1 (2018), p. 2227-2237. DOI : 10.18653/v1/N18-1202 (cité page 78).

N. PHAM, T. NGUYEN, J. NIEHUES, M. MÜLLER et A. WAIBEL. “Very deep self-attention networks for end-to-end Speech Recognition”. *CoRR* (2019). DOI : 10.48550/arXiv.1904.13377 (cité page 72).

V. PRATAP, A. TJANDRA, B. SHI et al. “Scaling speech technology to 1,000+ languages”. *Journal of Machine Learning Research (JMLR)* 25 (2024). DOI : 10.48550/arXiv.2305.13516 (cité page 73).

V. PRATAP, Q. XU, A. SRIRAM, G. SYNNAEVE et R. COLLOBERT. “MLS : A Large-Scale Multilingual dataset for speech research”. *Interspeech* (2020). DOI : 10.48550/arXiv.2012.03411 (cité pages 100, 101, 103).

P. PRICE, M. OSTENDORF, S. SHATTUCK-HUFNAGEL et C. FONG. “The use of prosody in Syntactic Disambiguation”. *Acoustical Society of America (ASA)* 90.6 (1991), p. 2956-70. DOI : 10.1121/1.401770 (cité pages 20, 81).

R. PRICE. “End-to-end Spoken Language Understanding without matched language speech model pretraining data”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2020), p. 7979-7983. DOI : 10.1109/ICASSP40776.2020.9054573 (cité pages 21, 86).

N. QIAN. “On the momentum term in gradient descent learning algorithms”. *Neural Networks* 12.1 (1999), p. 145-151. DOI : 10.1016/S0893-6080(98)00116-6 (cité page 38).

Y. QIAN, R. UBALE, V. RAMANARYANAN, P. LANGE, D. SUENDERMANN-OEFT, K. EVANINI et E. TSUPRUN. “Exploring ASR-free end-to-end modeling to improve Spoken Language Understanding in a cloud-based dialog system”. *Automatic Speech Recognition and Understanding (ASRU)* (2017), p. 569-576. DOI : 10.1109/ASRU.2017.8268987 (cité page 82).

L. QIN, Q. CHEN, T. XIE, Q. LI, J. LOU, W. CHE et M. KAN. “GL-CLeF : A global-local contrastive learning framework for cross-lingual Spoken Language Understanding”. *Association for Computational Linguistics (ACL)* (2022), p. 2677-2686. DOI : 10.18653/v1/2022.acl-long.191 (cité page 85).

L. RABINER. “A tutorial on Hidden Markov Models and selected applications in Speech Recognition”. *IEEE* 77.2 (1989), p. 257-286. DOI : 10.1109/5.18626 (cité page 69).

L. RABINER. “Automatic Speech Recognition - A brief history of the technology development”. (2005). URL : https://api.semanticscholar.org/CorpusID:12721778 (cité pages 19, 64).

A. RADFORD, J. KIM, T. XU, G. BROCKMAN, C. MCLEAVEY et I. SUTSKEVER. “Robust speech recognition via large-scale weak supervision”. *International Conference on Machine Learning (ICML)* 1182 (2023), p. 28492-28518. DOI : 10.48550/arXiv.2212.04356 (cité pages 20, 73, 79, 103, 181).

A. RADFORD et K. NARASIMHAN. “Improving language understanding by Generative Pre-Training”. (2018). URL : https://api.semanticscholar.org/CorpusID:49313245 (cité page 79).

A. RADFORD, J. WU, R. CHILD, D. LUAN, D. AMODEI et I. SUTSKEVER. “Language Models are unsupervised multitask learners”. (2019). URL : https://api.semanticscholar.org/CorpusID:160025533 (cité pages 79, 105).

L. RAMSHAW et M. MARCUS. “Text chunking using transformation-based learning”. *Association for Computational Linguistics (ACL)* (1995). DOI : 10.48550/arXiv.cmp-lg/9505040 (cité page 66).

A. RASTOGI, X. ZANG, S. SUNKARA, R. GUPTA et P. KHAITAN. “Towards scalable multi-domain conversational agents : The schema-guided dialogue dataset”. *Association for the Advancement of Artificial Intelligence (AAAI)* (2020). DOI : 10.48550/arXiv.1909.05855 (cité page 64).

M. RAVANELLI, P. BRAKEL, M. OMOLOGO et Y. BENGIO. “Light Gated Recurrent Units for Speech Recognition”. *Transactions on Emerging Topics in Computational Intelligence* 2.2 (2018), p. 92-102. DOI : 10.48550/arXiv.1803.10225 (cité page 47).

M. RAVANELLI, T. PARCOLLET, P. VANHARN PLANTINGA et al. “SpeechBrain : A general-purpose speech toolkit”. *CoRR* (2021). DOI : 10.48550/arXiv.2106.04624 (cité pages 22, 116, 129, 132, 179).

C. RAYMOND. “Décodage conceptuel : Co-articulation des processus de transcription et compréhension dans les systèmes de dialogue”. (2005) (cité page 74).

C. RAYMOND. “Robust tree-structured Named Entities Recognition from speech”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2013), p. 8475-8479. DOI : 10.1109/ICASSP.2013.6639319 (cité page 68).

C. RAYMOND et G. RICCARDI. “Generative and discriminative algorithms for Spoken Language Understanding”. *Interspeech* (2007). DOI : 10.21437/Interspeech.2007-448 (cité pages 20, 79, 125).

N. REIMERS et I. GUREVYCH. “Sentence-BERT : Sentence embeddings using siamese BERT-networks”. *Empirical Methods in Natural Language Processing (EMNLP) and International Conference on Natural Language Processing (ICNLP)* (2019), p. 3982-3992. DOI : 10.48550/arXiv.1908.10084 (cité pages 88, 109).

F. RINGEVAL, A. SONDEREGGER, J. SAUER et D. LALANNE. “Introducing the RECOLA multimodal corpus of remote collaborative and affective interactions”. *Automatic Face and Gesture Recognition (FG)* (2013). DOI : 10.1109/fg.2013.6553805 (cité page 100).

F. ROSENBLATT. “The perceptron : A probabilistic model for information storage and organization in the brain”. *Psychological Review* 65.6 (1958), p. 386-408. DOI : 10.1037/H0042519 (cité page 31).

S. RUDER. “An overview of gradient descent optimization algorithms”. *CoRR* (2016). DOI : 10.48550/arXiv.1609.04747 (cité page 40).

S. RUDER, N. CONSTANT, J. BOTHA et al. “XTREME-R : Towards more challenging and nuanced multilingual evaluation”. *Empirical Methods in Natural Language Processing (EMNLP)* (2021). DOI : 10.18653/v1/2F2021.emnlp-main.802 (cité pages 20, 84).

D. RUMELHART, G. HINTON et R. WILLIAMS. “Learning representations by back-propagating errors”. *Nature* 323 (1986), p. 533-536. DOI : 10.1038/323533a0 (cité pages 38, 40, 44).

E. SALESKY, M. WIESNER, J. BREMERMAN, R. CATTONI, M. NEGRI, M. TURCHI, D. OARD et M. POST. “The multilingual TEDx corpus for Speech Recognition and Translation”. *Interspeech* (2021). DOI : 10.48550/arXiv.2102.01757 (cité page 100).

S. SAMSON JUAN. “Exploiting resources from closely-related languages for Automatic Speech Recognition in low-resource languages from Malaysia”. (2015) (cité pages 23, 86).

R. SARIKAYA. “Rapid bootstrapping of statistical spoken dialogue systems”. *Speech Communication (SPECOM)* 50.7 (2008), p. 580-593. DOI : 10.1016/j.specom.2008.03.011 (cité pages 21, 85).

R. SARIKAYA, G. HINTON et B. RAMABHADRAN. “Deep belief nets for natural language call-routing”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2011), p. 5680-5683. DOI : 10.1109/ICASSP.2011.5947649 (cité pages 73, 77).

M. SAXON, S. CHOUDHARY, J. MCKENNA et A. MOUCHTARIS. “End-to-end Spoken Language Understanding for generalized voice assistants”. *Interspeech* (2021). DOI : 10.48550/arXiv.2106.09009 (cité page 64).

S. SCHNEIDER, A. BAEVSKI, R. COLLOBERT et M. AULI. “wav2vec : Unsupervised pre-training for Speech Recognition”. *Interspeech* (2019), p. 3465-3469. DOI : 10.48550/arXiv.1904.05862 (cité pages 20, 72, 95).

T. SCHULTZ. “Multilingual and crosslingual Speech Recognition”. *Broadcast News Transcription and Understanding Workshop (BNTUW)* (1998). DOI : 10.5445/IR/2F44598 (cité page 84).

T. SCHULTZ et A. BLACK. “Challenges with rapid adaptation of speech translation systems to new language pairs”. *International Conference on Acoustics Speech and Signal Processing Proceedings (ICASSP)* 5 (2006). DOI : 10.1109/ICASSP.2006.1661500 (cité pages 23, 85, 86).

S. SCHUSTER, S. GUPTA, R. SHAH et M. LEWIS. “Cross-lingual transfer learning for multilingual task oriented dialog”. *North American chapter of the Association for Computational Linguistics (NAACL)* (2018). DOI : 10.18653/v1/2FN19-1380 (cité pages 21, 86).

R. SCHWARTZ, S. MILLER, D. STALLARD et J. MAKHOUL. “Language understanding using hidden understanding models”. *International Conference on Spoken Language Processing (ICSLP)* (1996). DOI : 10.1109/ICSLP.1996.607771 (cité page 76).

H. SCHWENK. “Continuous space Language Models”. *Computer Speech and Language* 21.3 (2007), p. 492-518. DOI : 10.1016/j.csl.2006.09.003 (cité pages 77, 78).

H. SCHWENK et M. DOUZE. “Learning joint multilingual sentence representations with neural Machine Translation”. *Representation Learning for NLP (RepL4NLP)* (2017), p. 157-167. DOI : 10.18653/v1/W17-2619 (cité page 88).

H. SCHWENK, G. WENZEK, S. EDUNOV, E. GRAVE, A. JOULIN et A. FAN. “CCMatrix : Mining Billions of high-quality parallel sentences on the Web”. *Association for Computational Linguistics (ACL) and International Conference on Natural Language Processing (ICNLP)* (2021), p. 6490-6500. DOI : 10.48550/arXiv.1911.04944 (cité pages 106, 109).

F. SEIDE, G. LI, X. CHEN et D. YU. “Feature engineering in context-dependent Deep Neural Networks for conversational speech transcription”. *Automatic Speech Recognition and Understanding (ASRU)* (2011), p. 24-29. DOI : 10.1109/ASRU.2011.6163899 (cité page 71).

R. SENNRICH, B. HADDOW et A. BIRCH. “Neural Machine Translation of rare words with subword units”. *Association for Computational Linguistics (ACL)* (2016), p. 1715-1725. DOI : 10.48550/arXiv.1508.07909 (cité page 105).

D. SERDYUK, Y. WANG, C. FUEGEN, A. KUMAR, B. LIU et Y. BENGIO. “Towards end-to-end Spoken Language Understanding”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2018), p. 5754-5758. DOI : 10.1109/ICASSP.2018.8461785 (cité pages 20, 79, 81).

C. SERVAN, N. CAMELIN, C. RAYMOND, F. BÉCHET et R. DE MORI. “On the use of Machine Translation for Spoken Language Understanding portability”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2010), p. 5330-5333. DOI : 10.1109/ICASSP.2010.5494960 (cité pages 21, 86, 117).

K. SEYMORE et R. ROSENFELD. “Using story topics for Language Model adaptation”. (1997) (cité page 65).

P. SHAH, D. HAKKANI-TÜR, G. TÜR, A. RASTOGI, A. BAPNA, N. KENNARD et L. HECK. “Building a conversational agent overnight with dialogue self-play”. *CoRR* (2018). DOI : 10.48550/arXiv.1801.04871 (cité pages 63, 118).

C. SHANNON. “A mathematical theory of communication”. *The Bell System Technical Journal* 27.3 (1948), p. 379-423. DOI : 10.1002/j.1538-7305.1948.tb01338.x (cité page 70).

C. SHANNON. “Prediction and entropy of printed English”. *The Bell System Technical Journal* 30.1 (1951), p. 50-64. DOI : 10.1002/j.1538-7305.1951.tb01366.x (cité page 71).

Y. SHI, K. YAO, H. CHEN, Y. PAN, M. HWANG et B. PENG. “Contextual Spoken Language Understanding using Recurrent Neural Networks”. *International Conference on Acoustics, Speech and*

Signal Processing (ICASSP) (2015), p. 5271-5275. DOI : 10.1109/ICASSP.2015.7178977 (cité page 77).

M. SHOEYBI, M. PATWARY, R. PURI, P. LEGRESLEY, J. CASPER et B. CATANZARO. "Megatron-LM : Training multi-Billion parameter Language Models using model parallelism". *CoRR* (2019). DOI : 10.48550/arXiv.1909.08053 (cité page 79).

E. SHRIBERG. "Spontaneous speech : How people really talk and why engineers should care". *Interspeech* (2005), p. 1781-1784. DOI : 10.21437/Interspeech.2005-3 (cité pages 20, 81).

S. SI, W. MA, H. GAO et al. "SpokenWOZ : A large-scale speech-text benchmark for spoken task-oriented dialogue agents". *Advances in Neural Information Processing Systems (NeurIPS)* (2024). DOI : 10.48550/arXiv.2305.13040 (cité page 63).

E. SIMONNET, N. CAMELIN, P. DELÉGLISE et Y. ESTÈVE. "Exploring the use of attention-based Recurrent Neural Networks For Spoken Language Understanding". *Neural Information Processing Systems (NeurIPS)* (2015). URL : https://hal.science/hal-01433202 (cité page 78).

E. SIMONNET, S. GHANNAY, N. CAMELIN et Y. ESTÈVE. "Simulating ASR errors for training SLU systems". *Language Resources and Evaluation (LREC)* (2018). URL : https://aclanthology.org/L18-1499 (cité pages 78, 81, 126).

E. SIMONNET, S. GHANNAY, N. CAMELIN, Y. ESTÈVE et R. DE MORI. "ASR error management for improving Spoken Language Understanding". *Interspeech* (2017). DOI : 10.21437/Interspeech.2017-1178 (cité pages 81, 126).

S. STEPHANIE. "TINA : A natural language system for spoken language applications". *Computational Linguistics (COLING)* 18.1 (1992), p. 61-86. URL : https://aclanthology.org/J92-1004 (cité page 74).

S. STEVENS, J. VOLKMANN et E. NEWMAN. "A scale for the measurement of the psychological magnitude pitch". *Acoustical Society of America (ASA)* 8 (1937), p. 185-190. DOI : 10.1121/1.1915893 (cité page 92).

E. STRUBELL, A. GANESH et A. MCCALLUM. "Energy and policy considerations for Deep Learning in NLP". *CoRR* (2019). DOI : 10.18653/v1/2FP19-1355 (cité page 83).

M. SUNDERMEYER, R. SCHLÜTER et H. NEY. "LSTM neural networks for Language Modeling". *Interspeech* (2012), p. 194-197. DOI : 10.21437/Interspeech.2012-65 (cité page 78).

B. SUNDHEIM. "Overview of results of the MUC-6 evaluation". *Message Understanding Conference (MUC)* (1995). URL : https://aclanthology.org/M95-1002 (cité page 61).

P. SWIETOJANSKI, A. GHOSHAL et S. RENALS. “Unsupervised cross-lingual knowledge transfer in DNN-based LVCSR”. *Spoken Language Technology (SLT)* (2012), p. 246-251. DOI : 10.1109/SLT.2012.6424230 (cité page 85).

R. TAORI, I. GULRAJANI, T. ZHANG, Y. DUBOIS, X. LI, C. GUESTRIN, P. LIANG et T. HASHIMOTO. “Alpaca : A strong, replicable instruction-following model”. (2021). URL : https://crfm.stanford.edu/2023/03/13/alpaca.html (cité page 79).

N. TOMASHENKO, A. CAUBRIERE et Y. ESTÈVE. “Investigating adaptation and transfer learning for end-to-end Spoken Language Understanding from speech”. *Interspeech* (2019). DOI : 10.21437/interspeech.2019-2158 (cité pages 21, 81, 86).

H. TOUVRON, T. LAVRIL, G. IZACARD et al. “LLaMA : Open and efficient foundation Language Models”. *CoRR* (2023). DOI : 10.48550/arXiv.2302.13971 (cité page 79).

T. TRAN, S. TOSHNIWAL, M. BANSAL, K. GIMPEL, K. LIVESCU et M. OSTENDORF. “Parsing Speech : A neural approach to integrating lexical and acoustic-prosodic information”. *North American chapter of the Association for Computational Linguistics (NAACL)* (2017). DOI : 10.18653/v1/N18-1007 (cité pages 20, 81).

G. TÜR et R. DE MORI. “Spoken Language Understanding : Systems for extracting semantic information from speech”. *Wiley* (2011) (cité pages 19, 63, 64).

G. TÜR, D. HAKKANI-TÜR et L. HECK. “What is left to be understood in ATIS ?” : *2010 IEEE Spoken Language Technology Workshop* (2010), p. 19-24. DOI : 10.1109/SLT.2010.5700816 (cité page 63).

J. VALK et T. ALUMÄE. “VoxLingua107 : A dataset for Spoken Language Recognition”. *Spoken Language Technology (SLT)* (2021), p. 652-658. DOI : 10.48550/arXiv.2011.12998 (cité page 103).

C. VAN RIJSBERGEN. “Foundation of evaluation”. *Journal of Documentation* 30.4 (1974), p. 365-373. DOI : 10.1108/eb026584 (cité pages 64, 127, 184).

V. VAPNIK. “The nature of statistical learning theory”. *Statistics for Engineering and Information Science* (2000). DOI : 10.1007/978-1-4757-3264-1 (cité page 75).

V. VAPNIK. “Estimation of dependences based on empirical data”. *Information, Science and Statistics* (2006). DOI : 10.2307/2988246 (cité page 75).

A. VASWANI, N. SHAZEER, N. PARMAR, J. USZKOREIT, L. JONES, A. GOMEZ, L. KAISER et I. POLOSUKHIN. “Attention is all you need”. *Advances in Neural Information Processing Systems*

(NeurIPS) 30 (2017), p. 6000-6010. DOI : 10.48550/arXiv.1706.03762 (cité pages 20, 49, 51, 52, 78, 88, 106, 109, 180).

N. VU, D. IMSENG, D. POVEY, P. MOTLICEK, T. SCHULTZ et H. BOURLARD. "Multilingual Deep Neural Network based acoustic modeling for rapid language adaptation". *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2014), p. 7639-7643. DOI : 10.1109/ICASSP.2014.6855086 (cité pages 21, 85).

V. VUKOTIC, C. RAYMOND et G. GRAVIER. "Is it time to switch to word embedding and Recurrent Neural Networks for Spoken Language Understanding ?" : *Interspeech* (2015). DOI : 10.21437/Interspeech.2015-41 (cité pages 21, 77, 78, 119).

A. WAIBEL, T. HANAZAWA, G. HINTON, K. SHIKANO et K. LANG. "Phoneme recognition using Time-Delay Neural Networks". *Transactions on Acoustics, Speech and Signal Processing (TASSP)* 37.3 (1989), p. 328-339. DOI : 10.1109/29.21701 (cité page 71).

A. WAIBEL, T. SCHULTZ, S. VOGEL, C. FUGEN, M. HONAL, M. KOLSS, J. REICHERT et S. STÜKER. "Towards language portability in statistical speech translation". *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* 3 (2004). DOI : 10.1109/ICASSP.2004.1326657 (cité pages 23, 86).

C. WANG, M. RIVIERE, A. LEE et al. "VoxPopuli : A large-scale multilingual speech corpus for representation learning, Semi-Supervised Learning and interpretation". *Association for Computational Linguistics (ACL) and International Conference on Natural Language Processing (ICNLP)* (2021), p. 993-1003. DOI : 10.48550/arXiv.2101.00390 (cité pages 101, 103, 181).

C. WANG, A. WU, J. GU et J. PINO. "CoVoST 2 and massively multilingual speech translation". *Interspeech* (2021), p. 2247-2251. DOI : 10.48550/arXiv.2007.10310 (cité page 100).

P. WANG, L. WEI, Y. CAO, J. XIE et Z. NIE. "Large-scale unsupervised pre-training for end-to-end Spoken Language Understanding". *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2020), p. 7999-8003. DOI : 10.1109/ICASSP40776.2020.9053163 (cité page 83).

Y. WANG, A. ACERO et C. CHELBA. "Is Word Error Rate a good indicator for Spoken Language Understanding accuracy". *Automatic Speech Recognition and Understanding (ASRU)* (2003), p. 577-582. DOI : 10.1109/ASRU.2003.1318504 (cité pages 20, 81).

Y. WANG, A. ACERO, M. MAHAJAN et J. LEE. "Combining statistical and knowledge-based Spoken Language Understanding in conditional models". *Computational Linguistics (COLING) and Asso-*

ciation for Computational Linguistics (ACL) (2006), p. 882-889. URL : https://aclanthology.org/P06-2113 (cité page 68).

Y. WANG, A. MOHAMED, D. LE et al. "Transformer-based acoustic modeling for hybrid Speech Recognition". *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2020), p. 6874-6878. DOI : 10.1109/ICASSP40776.2020.9054345 (cité page 72).

Y. WANG, L. TANG et T. HE. "Attention-based CNN-BLSTM networks for joint intent detection and slot filling". *Chinese Computational Linguistics (CCL)* (2018). DOI : 10.1007/978-3-030-01716-3_21 (cité page 78).

Z. WANG, U. TOPKARA, T. SCHULTZ et A. WAIBEL. "Towards universal Speech Recognition". *International Conference on Multimodal Interfaces (ICMI)* (2002), p. 247-252. DOI : 10.1109/ICMI.2002.1167001 (cité pages 21, 84).

R. WATROUS et L. SHASTRI. "Learning phonetic features using connectionist networks". *Acoustical Society of America (ASA)* 81 (1987), p. 93-94. DOI : 10.1121/1.2024481 (cité page 72).

J. WEI, M. BOSMA, V. ZHAO et al. "Finetuned Language Models are zero-shot learners". *CoRR* (2021). DOI : 10.48550/arXiv.2109.01652 (cité page 79).

J. WEIZENBAUM. "ELIZAa computer program for the study of natural language communication between man and machine". *Communications of the ACM* 9.1 (1966), p. 36-45. DOI : 10.1145/365153.365168 (cité page 73).

W. WOODS. "Whats in a link : Foundations for semantic networks". *Representation and Understanding* (1975), p. 35-82. DOI : 10.1016/B978-0-12-108550-6.50007-0 (cité pages 19, 60, 63).

S. WU, A. CONNEAU, H. LI, L. ZETTLEMOYER et V. STOYANOV. "Emerging cross-lingual structure in pretrained Language Models". *Association for Computational Linguistics (ACL)* (2019). DOI : 10.18653/v1/2020.acl-main.536 (cité page 86).

B. XIE et R. PASSONNEAU. "Graph structured semantic representation and learning for financial news". *Florida Artificial Intelligence Research Society Conference (FLAIRS)* (2015). URL : https://cdn.aaai.org/ocs/10415/10415-46112-1-PB.pdf (cité page 67).

P. XU et R. SARIKAYA. "Convolutional Neural Network based triangular CRF for joint intent detection and slot filling". *Automatic Speech Recognition and Understanding (ASRU)* (2013), p. 78-83. DOI : 10.1109/ASRU.2013.6707709 (cité pages 64, 77).

P. XU et R. SARIKAYA. “Contextual domain classification in Spoken Language Understanding systems using Recurrent Neural Network”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2014), p. 136-140. DOI : 10.1109/ICASSP.2014.6853573 (cité page 77).

L. XUE, N. CONSTANT, A. ROBERTS, M. KALE, R. AL-RFOU, A. SIDDHANT, A. BARUA et C. RAFFEL. “mT5 : A massively multilingual pre-trained text-to-text Transformer”. *North American chapter of the Association for Computational Linguistics (NAACL)* (2020). DOI : 10.18653/V1/2F2021.NAACL-MAIN.41 (cité pages 20, 84).

S. YANG, P. CHI, Y. CHUANG et al. “SUPERB : Speech processing Universal PERformance Benchmark”. *Interspeech* (2021). DOI : 10.21437/interspeech.2021-1775 (cité page 83).

Z. YANG, Y. YANG, D. CER, J. LAW et E. DARVE. “Universal sentence representation learning with conditional Masked Language Model”. *Empirical Methods in Natural Language Processing (EMNLP)* (2021), p. 6216-6228. DOI : 10.48550/arXiv.2012.14388 (cité page 106).

K. YAO, B. PENG, Y. ZHANG, D. YU, G. ZWEIG et Y. SHI. “Spoken Language Understanding using Long Short-Term Memory neural networks”. *Spoken Language Technology (SLT)* (2014), p. 189-194. DOI : 10.1109/SLT.2014.7078572 (cité page 77).

K. YAO, G. ZWEIG, M. HWANG, Y. SHI et D. YU. “Recurrent Neural Networks for language understanding”. *Interspeech* (2013). DOI : 10.13140/2.1.2755.3285 (cité page 77).

G. YENDURI, M. RAMALINGAM, G. SELVI et al. “GPT (Generative Pre-Trained Transformer) A comprehensive review on enabling technologies, potential applications, emerging challenges, and future directions”. *Access* 12 (2023), p. 54608-54649. DOI : 10.48550/arXiv.2305.10435 (cité page 79).

M. ZANON BOITO, F. BOUGARES, F. BARBIER, S. GAHBICHE, L. BARRAULT, M. ROUVIER et Y. ESTÈVE. “Speech resources in the Tamasheq LInguage”. *Language Resources and Evaluation Conference (LREC)* (2022), p. 2066-2071. DOI : 10.48550/arXiv.2201.05051 (cité page 135).

M. ZEILER. “ADADELTA : An adaptive learning rate method”. *CoRR* (2012). DOI : 10.48550/arXiv.1212.5701 (cité pages 42, 136).

K. ZHANG, H. XU, J. TANG et J. LI. “Keyword extraction using Support Vector Machine”. *Adances in Web-Age Information Management (WAIM)* (2006), p. 85-96. DOI : 10.1007/11775300_8 (cité page 75).

L. ZHANG et H. WANG. “Using bidirectional Transformer-CRF for Spoken Language Understanding”. *Natural Language Processing and Chinese Computing (NLPCC)* (2019). DOI : 10.1007/978-3-030-32233-5_11 (cité page 78).

X. ZHANG et L. HE. “End-to-end cross-lingual Spoken Language Understanding model with multilingual pretraining”. *Interspeech* (2021). DOI : 10.21437/interspeech.2021-818 (cité pages 21, 85).

Y. ZHANG, M. PEZESHKI, P. BRAKEL, S. ZHANG, C. LAURENT, Y. BENGIO et A. COURVILLE. “Towards end-to-end Speech Recognition with deep Convolutional Neural Networks”. *CoRR* (2016). DOI : 10.48550/arXiv.1701.02720 (cité pages 20, 72, 81).

G. ZHENG, Y. XIAO, K. GONG, P. ZHOU, X. LIANG et L. LIN. “Wav-BERT : Cooperative acoustic and linguistic representation learning for low-resource Speech Recognition”. *Empirical Methods in Natural Language Processing (EMNLP)* (2021). DOI : 10.18653/v1/2F2021.findings-emnlp.236 (cité page 83).

S. ZHU et K. YU. “Encoder-decoder with focus-mechanism for sequence labelling based Spoken Language Understanding”. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2017), p. 5675-5679. DOI : 10.1109/ICASSP.2017.7953243 (cité page 78).