Modèles faiblement supervisés pour la documentation automatique des langues

Shu Okabe

université PARIS-SACLAY

Modèles faiblement supervisés pour la documentation automatique des langues

Weakly Supervised Models for Computational Language Documentation

Thèse de doctorat de l'université Paris-Saclay

École doctorale n°580 : sciences et technologies de l'information et de la communication (STIC)
Spécialité de doctorat : Informatique
Graduate School : Informatique et sciences du numérique.
Référent : Faculté des sciences d'Orsay

Thèse préparée dans l'unité de recherche **Laboratoire interdisciplinaire des sciences du numérique (Université Paris-Saclay, CNRS)**, sous la direction de **François YVON**, Directeur de recherche

Thèse soutenue à Paris-Saclay, le 19 décembre 2023, par

Shu OKABE

Composition du jury

Membres du jury avec voix délibérative

Agata SAVARY Professeure, LISN, CNRS, Université Paris-Saclay	Présidente
Claire GARDENT Directrice de recherche, LORIA, CNRS, Université de Lorraine	Rapporteur & Examinatrice
Alexis NASR Professeur, LIS, CNRS, Aix-Marseille Université	Rapporteur & Examinateur
Roland KUHN Agent de recherches principal, Conseil national de recherches du Canada	Examinateur
François PELLEGRINO Directeur de recherche, DDL, CNRS, Université Lyon 2	Examinateur

THESE DE DOCTORAT

NNT : 2023UPASG091

université PARIS-SACLAY

ÉCOLE DOCTORALE
Sciences et technologies de l'information et de la communication (STIC)

Titre : Modèles faiblement supervisés pour la documentation automatique des langues
Mots clés : documentation automatique des langues ; segmentation en mots ; modèle bayésien non paramétrique ; génération de gloses interlinéaires ; supervision faible ; linguistique de terrain

Résumé : Face à la menace d'extinction de la moitié des langues parlées aujourd'hui d'ici la fin du siècle, la documentation des langues est un domaine de la linguistique notamment consacré à la collecte, annotation et archivage de données. Dans ce contexte, la documentation automatique des langues vise à outiller les linguistes pour faciliter différentes étapes de la documentation, à travers des approches de traitement automatique du langage. Dans le cadre du projet de documentation automatique CLD2025, cette thèse s'intéresse principalement à deux tâches : la segmentation en mots, identifiant les frontières des mots dans une transcription non segmentée d'une phrase enregistrée, ainsi que la génération de gloses interlinéaires, prédisant des annotations linguistiques pour chaque unité de la phrase. Pour la première, nous améliorons les performances des modèles bayésiens non paramétriques utilisés jusque là à travers une supervision faible, en nous appuyant sur des ressources disponibles de manière réaliste lors de la documentation, comme des phrases déjà segmentées ou des lexiques. Comme nous observons toujours une tendance de sur-segmentation dans nos modèles, nous introduisons un second niveau de segmentation : les morphèmes. Nos expériences avec divers types de modèles de segmentation à deux niveaux indiquent une qualité de segmentation sensiblement meilleure ; nous constatons, par ailleurs, les limites des approches uniquement statistiques pour différencier les mots des morphèmes. La seconde tâche concerne la génération de gloses, soit grammaticales, soit lexicales. Comme ces dernières ne peuvent pas être prédites en se basant seulement sur les données d'entraînement, notre modèle statistique d'étiquetage de séquences fait moduler, pour chaque phrase, les étiquettes possibles et propose une approche compétitive avec les modèles neuronaux les plus récents.

Title: Weakly Supervised Models for Computational Language Documentation
Keywords: Computational language documentation; word segmentation; Bayesian non-parametric model; interlinear gloss generation; weak supervision; field linguistics

Abstract: In the wake of the threat of extinction of half of the languages spoken today by the end of the century, language documentation is a field of linguistics notably dedicated to the recording, annotation, and archiving of data. In this context, computational language documentation aims to devise tools for linguists to ease several documentation steps through natural language processing approaches. As part of the CLD2025 computational language documentation project, this thesis focuses mainly on two tasks: word segmentation to identify word boundaries in an unsegmented transcription of a recorded sentence and automatic interlinear glossing to predict linguistic annotations for each sentence unit. For the first task, we improve the performance of the Bayesian non-parametric models used until now through weak supervision. For this purpose, we leverage realistically available resources during documentation, such as already-segmented sentences or dictionaries. Since we still observe an over-segmenting tendency in our models, we introduce a second segmentation level: the morphemes. Our experiments with various types of two-level segmentation models indicate a slight improvement in the segmentation quality. However, we also face limitations in differentiating words from morphemes, using statistical cues only. The second task concerns the generation of either grammatical or lexical glosses. As the latter cannot be predicted using training data solely, our statistical sequence-labelling model adapts the set of possible labels for each sentence and provides a competitive alternative to the most recent neural models.

Remerciements

Tout d'abord, je tiens à remercier mon directeur de thèse, François. Merci pour m'avoir fait confiance il y a un peu plus de trois ans, pour m'avoir accompagné et encouragé tout au long de cette thèse, face aux obstacles, pour m'avoir guidé et éclairci, pour vos conseils bienveillants, pour avoir pris le temps d'identifier et d'expliquer ce qui m'empêchait d'avancer, pour ce cadre de travail et pour être si disponible, et ce, malgré votre emploi du temps, pour un appel, une relecture ou une réunion. Je n'ai pas pu demander meilleur encadrement. À travers ces mots, j'espère traduire, autant que possible, toute ma gratitude envers vous. Merci, infiniment.

Je remercie également Agata SAVARY, Claire GARDENT, Alexis NASR, Roland KUHN et François PELLEGRINO, pour avoir accepté d'être membres de mon jury de thèse ; merci pour cet honneur. Merci pour vos commentaires, remarques et questions enrichissantes sur ma thèse.

Un remerciement à Laurent en particulier pour m'avoir encadré au début de ma thèse et pour les conseils par la suite sur mes travaux. Merci aussi à Gilles pour avoir suivi mon évolution pendant la thèse en tant que responsable du projet CLD2025.

Je vais maintenant remercier le laboratoire, le personnel et toutes les personnes qui ont contribué à rendre ce parcours plus convivial, surtout après une période initiale difficile. Merci à Laurence et Olga [1] pour avoir accompagné mes missions. Merci à Sophie pour m'avoir aidé tant de fois.

Meric à Caio pour tous ces conseils sur la recherche et le monde de la recherche. Merci à Guillaume pour cet éclairage sur le japhug et plus encore. Merci à Thomas pour toute ton aide et ta disponibilité pour répondre à mes questions, lorsque j'étais perdu. Merci à Lucas pour m'avoir suivi et écouté pendant ces trois années.

Merci à l'IUT, enseignants et élèves, de m'avoir formé pendant ces missions d'enseignement. Un remerciement particulier à Benjamin, pour avoir été un guide dans cette aventure en période de pandémie, merci pour ces conseils pédagogiques et pour avoir toujours été là quand je rencontrais des problèmes.

Je remercie également les doctorants en général, ceux du département STL et en particulier ceux de l'ex-TLP, avec qui j'ai partagé ce deuxième étage. Merci à vous pour ces discussions : Hugues, Marc, Camille, Juan, Simon, Léo, Alina, Anh Khoa, Minh Quang, Lisa, Mathilde et d'autres encore. Merci pour ces déjeuners au CESFO et ces pauses à la cafétéria.

Merci Aina et Syrielle pour m'avoir accueilli dans votre bureau et pour ces quelques fois où nous avons pu y travailler ensemble. J'espère que l'on a pu rattraper le temps volé par la suite.

À Aman, pour avoir été mon binôme pour ce cours. À Maxime, pour ces nombreux feux d'artifices pour colorer nos pauses digestives à travers des finesses (ou des prises de tête). À François, pour ces conseils et ces encouragements sur ma dernière ligne droite. À Yajing, pour toutes ces histoires rocambolesques que je ne pourrai oublier. À Francesca, pour avoir réalisé une œuvre unique. À Nicolas, même si je ne devrais te remercier qu'en période de conférence. À Dávid, pour avoir mangé italien et apprécié ces parkings canadiens. À Sofiya, pour ce récit de ton séjour au Japon. À Tom, pour ce pique-nique. Merci à Léa-Marie, pour avoir supporté mes plaintes parfois beaucoup trop longues et pour ces discussions d'après-midi. À Élise, pour m'avoir appris que l'on disait tarte flambée. À Lufei, pour m'avoir appris ce qu'est un brevet, ce qui est très différent des patients. À Rémi, pour avoir partagé ces moments dans le même bureau ou dans le RER. À Jitao, pour avoir toujours été de bon conseil et je me souviendrai de ces discussions dans le couloir entre nos deux bureaux.

1. Par la suite, les listes suivront un ordre chronologique ou alphabétique.

À ceux qui ont commencé en même temps que moi. À Théo, pour m'avoir fait découvrir des sports marins dont je ne soupçonnais pas l'existence. À Natalia, pour tous ces rires que j'ai pu vivre, toutes ces histoires que tu racontes si bien et toutes ces idées de cadeaux. À Paul, pour avoir été le pilier de notre étage, toujours présent, et qui m'a motivé à venir plus souvent. Et bien sûr merci Alban, je suis heureux d'avoir pu passer ces années de thèses à tes côtés ; ces souvenirs de conférences, surtout la première en présentiel, seront marqués à jamais, pour cette vue sur tout Dublin au crépuscule, au soleil écarlate, ou pour cette course à l'aéroport, pour ces quelques parties de jeux de société ou ces courses virtuelles, où même assisté, je perdais.

À ceux que j'ai rencontrés lors des conférences. Merci Antoine et Hee-soo pour ces moments mémorables à TALN sous le soleil provençal ou parisien, sur un bateau ou un manège.

Je remercie également les personnes qui m'ont guidé vers le chemin de la recherche. Merci à Arnaud pour ces conseils visionnaires. Merci à Loïc pour avoir initié un parcours que je suis. Merci à Julia pour avoir évoqué l'idée d'une thèse, à un moment où je n'en avais pas conscience. Merci à Frédéric et Lucia pour votre encadrement et vos conseils lors de cette première expérience dans le monde de la recherche ; vous m'avez donné l'envie de continuer.

Je vais maintenant mentionner mes amis, vous qui m'avez soutenu (ou supporté) ces trois ans. Merci Éric pour tous ces voyages méditerranéens pour aller où le ciel est bleu, où les repas sont succulents et où les chants grecs nous transportent. Je n'oublierai pas le paradis sucré, ces paysages splendides et ces monuments inoubliables. Merci de m'avoir accompagné dans tous ces moments. Merci Antoine pour ces moments autour d'un (des) thé(s) et des films ou des séries, à suivre les aventures tantôt d'un détective, tantôt d'espions, pour ces invitations aux représentations théâtrales toujours comiques et pour ce séjour londonien. Pour vous deux, merci pour ces souvenirs de nos jours insouciants, immortalisés dans des chefs-d'œuvre du septième art. Merci aussi Manon de nous accompagner à ces dîners où je vous impose (presque) le lieu et les plats.

Merci Diane et Estelle pour m'avoir encouragé tout au long de la thèse à travers ces repas faits maison, ces onigiris à Bastille, ces dîners dans ce cirque et cette escapade dans la Renaissance.

À mes amis du lycée, même éparpillés, pour ces retrouvailles. Merci à Arnaud, Adélaïde et Emmanuel-Paul pour m'avoir offert des moyens pour m'évader du quotidien et pour me demander si je trouvais.

À cette team parc floral, Ignacio, Juliette et Anastasia, merci pour ces retrouvailles autour de ramens ou de crêpes ou ce week-end de vélo où j'ai apprécié un certain banc dans un parc. À Christophe pour ces discussions sur la recherche autour de dîners et vins.

À mes amis loups-garous, Aurélien, Idriss, Hugo et Édith, qui constatez mon retour sur ce plateau chéri, merci pour ces messages quotidiens hilarants. Merci aux plantes. Merci Cédric pour ces quelques (rares) fois où je me suis invité à tes séances à la bibliothèque et une mention particulière pour toute ton aide vers la fin de ma thèse, jusqu'aux dernières minutes de la soutenance. Merci Juliette pour ces appels tardifs la nuit pour nous tenir au courant des deux côtés de la Manche et pour être la héroïne d'une série incroyablement captivante. Merci Valentin pour le soleil que tu as apporté, les rires que nous avons partagés et surtout le dépaysement que tu nous as montré depuis les sommets (à base carrée ?) de la terre (et une mention spéciale pour m'avoir indiqué comment faire cette carte du monde). Et Suzanne, merci pour m'avoir accompagné tout au long de cette thèse à travers des encouragements dans des moments de doutes. Tant de moments partagés à Paris ou ailleurs, à voyager dans le temps par les tableaux ou la musique. À ces plaintes quotidiennes que tu supportes, à ces souvenirs du RER B, à cette passion pour rester sur un plateau qui nous a tant formé.

Je vais clore en remerciant ma famille. Merci à mes proches au Japon pour m'avoir continuellement encouragé malgré la distance et les circonstances. Et en particulier, un immense merci à mes parents. Pour m'avoir toujours soutenu dans mes choix quoiqu'il arrive et pour m'avoir donné les moyens de réussir. Pour avoir toujours été un ancre dans ma vie.

Et pour finir véritablement, je dirai : *thank your for the music, for giving it to me*.

Table des matières

Table des figures

Liste des tableaux

Glossaire

AER Taux d'erreur d'alignement, abréviation de l'anglais *Alignment Error Rate*.

AG Abréviation de l'anglais *Adaptor Grammar*.

BF F-score sur les frontières, abréviation de l'anglais *Boundary F1-score*.

BP Précision sur les frontières, abréviation de l'anglais *Boundary Precision*.

BR Rappel sur les frontières, abréviation de l'anglais *Boundary Recall*.

CFG Grammaire non contextuelle, abréviation de l'anglais *Context-Free Grammar*.

CNN Réseau de neurones convolutif, abréviation de l'anglais *Convolutional Neural Network*.

CRF Champ aléatoire conditionnel, abréviation de l'anglais *Conditional Random Field*.

CRP Processus du restaurant chinois, abréviation de l'anglais *Chinese Restaurant Process*.

HMM Modèle de Markov caché, abréviation de l'anglais *Hidden Markov Model*.

IGT Texte avec gloses interlinéaires, abréviation de l'anglais *Interlinear Glossed Text*.

IPA Alphabet phonétique international, abréviation de l'anglais *International Phonetic Alphabet*.

LF F-score sur les types, abréviation de l'anglais *Lexicon F1-score*.

LP Précision sur les types, abréviation de l'anglais *Lexicon Precision*.

LR Rappel sur les types, abréviation de l'anglais *Lexicon Recall*.

LSTM Abréviation de l'anglais *Long Short-Term Memory*.

MDL Longueur de description minimale, abréviation de l'anglais *Minimum Description Length*.

OCR Reconnaissance optique de caractères, abréviation de l'anglais *Optical Character Recognition*.

PCFG Grammaire non contextuelle probabiliste, abréviation de l'anglais *Probabilistic Context-Free Grammar*.

PoS Partie du discours, abréviation de l'anglais *Part of Speech*.

RNN Réseau de neurones récurrent, abréviation de l'anglais *Recurrent Neural Network*.

TAL Traitement automatique des langues.

TL Longueur moyenne des types, abréviation de l'anglais *Type Length*.

TTR Rapport type-occurrence, abréviation de l'anglais *Type-Token Ratio*.

WF F-score sur les occurrences, abréviation de l'anglais *Word F1-score*.

WL Longueur moyenne des occurrences, abréviation de l'anglais *Word Length*.

WP Précision sur les occurrences, abréviation de l'anglais *Word Precision*.

WR Rappel sur les occurrences, abréviation de l'anglais *Word Recall*.

Chapitre 1

Introduction

Per sogni e per chimere
e per castelli in aria
l'anima ho milionaria.

Premier tableau, *La Bohème*, Puccini

1.1 Contexte

D'ici la fin du siècle, la moitié des langues parlées aujourd'hui risquent de disparaître (Austin et Sallabank, 2011). À ce titre, (Hale et al., 1992) constitue une véritable prise de conscience de la communauté linguistique sur cette menace, bien que d'autres publications antérieures aient également alerté sur le sujet ; selon eux, la part des langues qui seront fortement en danger voire disparues s'élève plutôt autour de 90 %. En 1997, Bernard (1997) considère que plus de 90 % des langues sont parlées par 5 % de la population mondiale, soulignant la très forte disparité dans le nombre de locuteurs des langues. Les estimations plus optimistes de (Crystal, 2000) et de (Nettle et Romaine, 2000) indiquent une proportion plus basse de langues en danger, « seulement » 50 %.

Plus récemment, plusieurs taxonomies ont été proposées pour mesurer le niveau de risque d'extinction, comme l'EGIDS (« *Expanded Graded Intergenerational Disruption Scale* ») de (Lewis et Simons, 2010). La plus complète selon (Austin et Sallabank, 2011) est celle de l'UNESCO (UNESCO, 2003), qui a notamment été employée pour l'Atlas des langues en danger dans le monde (UNESCO, 2010). Cette classification a été mise à jour et est désormais utilisée dans l'Atlas mondial des langues [1] (*World Atlas of Languages* en anglais) de l'UNESCO. Elle définit une échelle de six degrés de vitalité, allant de « sans risque » (*safe* en anglais) à « en situation critique » (*critically endangered*), pour les plus de 7 000 langues parlées à l'heure actuelle. Un peu plus de 1 000 d'entre elles sont considérées comme étant peu, voire non risquées ; le reste (plus de 80 %) est en danger à des stades variés. Par ailleurs, selon l'édition actuelle de l'*Ethnologue* [2] (Eberhard et al., 2023), 43 % des 7 168 langues qui y sont répertoriées sont estimées comme étant en danger de disparition. Bien que les proportions changent en fonction des sources, même récentes, le constat est sans appel.

Les langues peuvent être en danger à cause de divers facteurs que Austin et Sallabank (2011) classifient en quatre catégories : les catastrophes naturelles, les guerres, les politiques d'assimilation et la domination (par exemple, culturelle ou économique). L'extinction d'une langue affecte avant tout la communauté, qui perd son identité ethnique et culturelle (Bernard, 1992). Puis, les

1. https://en.wal.unesco.org/.
2. https://www.ethnologue.com/.

conséquences impactent certes directement la linguistique, qui pourrait voir la majeure partie de son domaine d'étude disparaître, comme l'alerte Krauss (1992), mais signifient également la disparition d'un patrimoine culturel immatériel (comme le définit l'UNESCO). Il s'agit notamment d'une perte des coutumes, des pratiques et des connaissances humaines, en particulier sur l'environnement local ; Seifart et al. (2018) mentionnent par exemple les travaux de (Fabricant et Farnsworth, 2001), indiquant que 80 % des substances pharmaceutiques extraites à partir des plantes proviennent de médecines traditionnelles.

Notons que cette thèse s'achève au début de la décennie 2022-2032, qui a été déclarée comme étant la « Décennie internationale des langues autochtones » [3] par l'UNESCO, dont l'objectif est de sensibiliser sur l'état des langues autochtones et d'encourager les initiatives pour leur préservation ou leur revitalisation.

Suite à l'alarme donnée par (Hale et al., 1992) sur la vitalité des langues dans le monde mais également sur le domaine de la linguistique en lui-même, la réponse principale a été de renforcer les efforts de documentation. La documentation des langues est un domaine de la linguistique qui vise à collecter, annoter et archiver des données d'une langue (Woodbury, 2011). Elle mène à la production de corpus (des enregistrements, des vidéos, des transcriptions, des textes), mais également de connaissances linguistiques (des grammaires, des dictionnaires). Des initiatives de documentation ont été entreprises pour couvrir différentes langues du globe, comme le rappelle Woodbury (2011), avec des projets comme DOBES [4] (*Dokumentation bedrohter Sprachen* en allemand), archivé désormais par le Max Planck Institute for Psycholinguistics. Selon (Seifart et al., 2018), parmi les plus de 3 000 langues considérées comme étant en danger, la proportion de langues documentées est passée de 46 % avant 1992, année de publication de (Hale et al., 1992), à 61 % en 2016.

Notons que Himmelmann (1998) puis Woodbury (2003) distingue la documentation des langues de leur description : la première s'intéresse à « la collecte, la transcription et la traduction de données premières », là où la seconde consiste davantage en une analyse linguistique (avec la constitution d'une grammaire ou d'un dictionnaire par exemple) ; la définition que nous considérons englobe ces deux aspects.

Afin de faciliter les travaux de documentation, les linguistes ont déjà recours à de nouvelles technologies, une pratique plus ou moins harmonisée, avec des enregistreurs numériques comme AIKUMA (Bird et al., 2014) (ou des variantes), en suivant par exemple la méthodologie de la documentation de langue orale basique (*Basic Oral Language Documentation* en anglais) (Bird, 2010), et des outils d'annotations comme Praat (Boersma et Weenink, 1992–2022), FLEx [5] (Rogers, 2010) ou ELAN (Wittenburg et al., 2006), comme illustré en figure 1.1. Bettinson et Bird (2017) proposent également une boîte à outils d'applications Web dédiées à la documentation sur le terrain, modulables selon les besoins des linguistes et les contraintes sur place (comme la connexion Internet).

Toutefois, pour le moment, la plupart des étapes de documentation se fait principalement de manière manuelle, ce qui s'avère être très coûteux en temps. De fait, pour la première étape de transcription des enregistrements, Seifart et al. (2018) estiment que 40 à 100 heures sont nécessaires pour transcrire une heure d'enregistrement ; Brinckmann (2009) l'évalue même à 200 heures. L'étape de transcription représente alors un premier goulot d'étranglement (Brinck-

3. https://fr.idil2022-2032.org/.
4. https://dobes.mpi.nl/.
5. https://software.sil.org/fieldworks/.

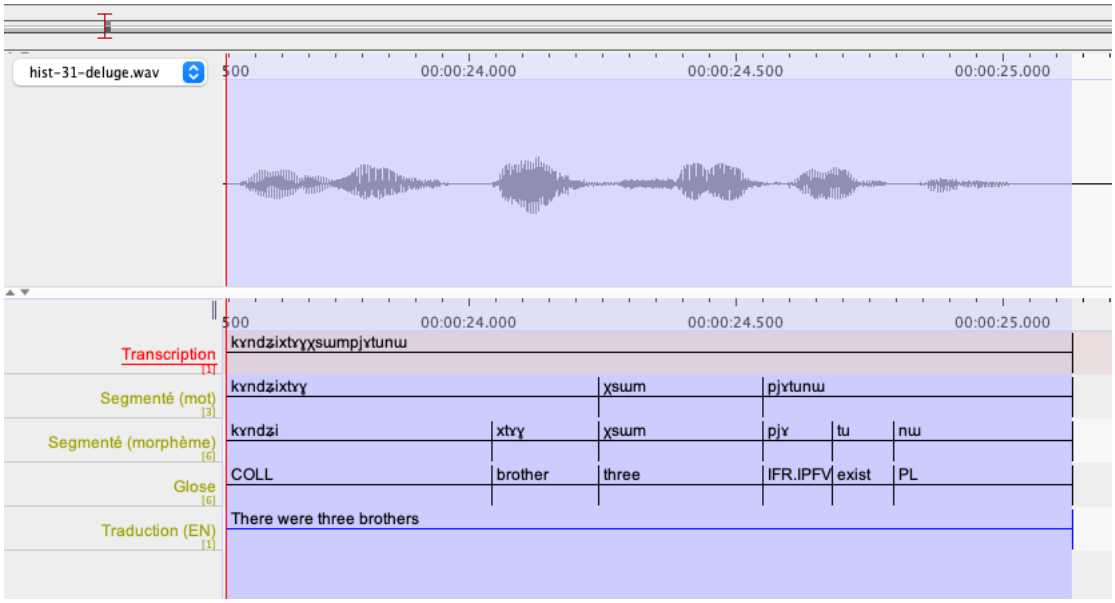


FIGURE 1.1 – Capture d'écran de l'outil ELAN (Wittenburg et al., 2006) avec les strates d'annotations linguistiques que nous étudions dans cette thèse.

mann, 2009). Puis survient le deuxième goulot d'étranglement, celui des annotations linguistiques, qui sont elles aussi coûteuses en temps et nécessitent d'avoir une expertise linguistique.

C'est pourquoi, comme le soulignent Seifart et al. (2018), la quantité de données collectées (des enregistrements) est significativement plus importante que celle des données annotées (transcrites, traduites voire analysées). Si les données ne sont pas annotées, en particulier non transcrites, elles seront malheureusement moins pertinentes pour les documentations futures selon (Seifart et al., 2018) ; elles sont alors destinées à ce que Himmelmann (2006) qualifie de « cimetières de données » (*data graveyards* en anglais).

Parmi les trois solutions que proposent Seifart et al. (2018) pour traiter le goulot d'étranglement notamment de la transcription, l'une consiste en l'utilisation d'approches de traitement automatique des langues (TAL) pour la transcription de la parole.

En effet, pour répondre à ces contraintes de temps qui limitent la production d'annotations, des initiatives ont été entreprises en utilisant des méthodes de TAL lors de la documentation : il s'agit de la documentation automatique des langues (ou *computational language documentation* en anglais). Elle a pour objectif de faciliter et d'accélérer différentes étapes de l'annotation en automatisant partiellement, à travers une collaboration interdisciplinaire entre les linguistes d'une part et les informaticiens d'autre part. Pour le linguiste, il s'agit d'un pré-traitement des données qui lui permet notamment de se concentrer sur les parties les plus intéressantes, en diminuant la part des tâches répétitives ou les erreurs de saisies (Adams et al., 2018).

Parmi les premiers travaux de TAL pour les langues en danger, Bird et Chiang (2012) explorent la traduction automatique pour quinze langues en danger en Papouasie-Nouvelle-Guinée. Puis, l'automatisation s'est en premier lieu effectuée pour la tâche de transcription, qui a entre autres bénéficié des travaux sur les langues peu dotées comme étudiés par (Besacier et al., 2014). En effet, les enregistrements de la langue sur le terrain constituent la première et principale ressource recueillie lors de la documentation des langues. À ce titre, différents travaux de reconnaissance de la parole ou de transcription automatique, en fonction de la quantité de données à disposition, ont été entrepris sur des langues en cours de documentation (Adams et al., 2017, 2018; Jimer-

son et Prud'hommeaux, 2018). En particulier, Adams et al. (2018) ont conçu et publié leur outil de transcription de phonèmes, Persephone [6]. De même, Foley et al. (2018) présentent un outil de transcription à destination des linguistes, Elpis [7], basé sur le système de reconnaissance de la parole Kaldi (Povey et al., 2011). Enfin, nous pouvons mentionner Adams et al. (2021) qui intègrent un système neuronal, ESPnet (Watanabe et al., 2018), dans Elpis, afin de proposer aux linguistes un autre modèle de transcription. Une approche analogue est d'étiqueter des segments de l'enregistrement avec les mots correspondants de la traduction (Duong et al., 2016; Anastasopoulos et al., 2017) ou d'utiliser cette traduction comme ressource supplémentaire pour la transcription (Anastasopoulos et Chiang, 2018).

Il existe également des prototypes de systèmes de documentation des langues intégrant des modèles de TAL, permettant à terme une transcription automatique et une génération d'annotations linguistiques notamment morphosyntaxiques, comme celui de (Neubig et al., 2018).

Le domaine de la documentation automatique des langues suscite ainsi un intérêt croissant des deux communautés et fait également l'objet de thèses récentes (Adams, 2017; Anastasopoulos, 2019; Godard, 2019; Zanon Boito, 2021; Moeller et al., 2021), pour la transcription mais également pour les autres étapes de la documentation, que nous aborderons ici.

C'est dans ce contexte que la présente thèse a été effectuée dans le cadre du projet franco-allemand « La documentation computationnelle des langues à l'horizon 2025 – CLD2025 » [8], qui est le successeur du projet de documentation automatique des langues *Breaking the Unwritten Language Barrier* (BULB) (Adda et al., 2016) ; nos travaux constituent donc une suite à la thèse de (Godard, 2019). L'objectif initial du projet CLD2025 était alors d'étendre et d'améliorer les méthodes de TAL pour la documentation des langues, au vu des résultats prometteurs obtenus dans BULB, à travers une intégration plus poussée dans les outils d'annotation linguistique. Par une collaboration interdisciplinaire entre des laboratoires de linguistique et d'informatique, l'idée était également de rendre standard le recours à l'automatisation dans les pratiques de documentation des langues.

Nous traitons principalement deux tâches intermédiaires de la documentation ; la première est la segmentation de séquences. Une fois les enregistrements transcrits automatiquement, notamment à l'aide des outils de transcription phonétique, nous obtenons une chaîne de caractères non segmentée. Il est donc nécessaire de la segmenter en mots, tels que les linguistes les définissent. Cette tâche monolingue et textuelle a déjà été abordée dans le cadre du projet de documentation automatique des langues BULB (Godard et al., 2016, 2018d,c), principalement à l'aide d'approches non supervisées. Ensuite, la phrase doit également être segmentée en morphèmes, les plus petites unités significatives dans la langue.

La seconde est la génération automatique de gloses, à savoir des annotations linguistiques explicitant soit le rôle grammatical, soit la signification lexicale d'un morphème. Les gloses sont aussi coûteuses à produire, car elles nécessitent une compréhension de la langue ainsi qu'une maîtrise linguistique : elles sont donc obtenues à travers une analyse manuelle.

Jusque là, les méthodes déployées pour la segmentation des langues peu dotées étaient principalement non supervisées, du fait de l'insuffisance de la taille des données pour une approche supervisée. De même, la phrase de la langue source était la seule à être utilisée pour la génération des gloses, la traduction correspondante n'étant pas prise en compte. Or, dans le cadre de la documentation, des corpus sont constitués, avec éventuellement des grammaires ou des lexiques

6. https://github.com/persephone-tools/persephone.
7. *Endangered Language Pipeline and Inference System* ; https://github.com/CoEDL/elpis/.
8. ANR-19-CE38-0015 ; https://anr.fr/Projet-ANR-19-CE38-0015.

en complément, qui n'ont pas été utilisés jusque là, comme le constate Bird (2020). En effet, la situation « zéro ressource » (comme dans (Dunbar et al., 2017)) est rare dans les faits et ne sert ni les linguistes en documentation, ni les informaticiens en TAL ; cela revient entre autres à ignorer les travaux linguistiques antérieurs (Bird, 2020).

Nous nous intéressons donc à la question centrale de l'utilisation des ressources disponibles pour améliorer la qualité des modèles à travers une supervision faible dans le cadre de la documentation automatique des langues. En outre, nous essayons également de concevoir des approches qui parviennent à utiliser efficacement les données, même de petite taille.

1.2 Contributions

Nous listons ci-dessous les principales contributions de la thèse :

— un ensemble de méthodes de supervision faible pour des modèles bayésiens non paramétriques de segmentation en mots à travers des ressources existantes lors de la documentation ;
— une première étape d'apprentissage incrémental de segmentation en mots, en vue d'une intégration dans une plateforme d'annotation linguistique ;
— une étude des différentes stratégies de segmentation simultanée en mots et morphèmes à l'aide des modèles précédents ;
— une réimplémentation en Python d'un modèle bayésien non paramétrique, avec des extensions adaptées à nos expériences de segmentation à un ou deux niveaux, avec la possibilité d'utiliser une supervision faible ;
— une analyse des corpus utilisés provenant de la documentation des langues, d'un point de vue des statistiques de fréquence des unités ;
— une comparaison de la tâche de génération automatique de gloses avec la tâche d'étiquetage en parties du discours ;
— une réutilisation pour la génération de gloses d'un modèle d'étiquetage de séquences permettant de moduler dynamiquement les étiquettes proposées et supervisé par des alignements automatiques ;
— un pré-entraînement du modèle précédent à travers des données glosées d'autres langues.

1.3 Structure

La thèse s'articule autour des chapitres suivants :

Le chapitre 2 présente principalement les deux tâches abordées, la segmentation de séquences et la génération automatique de gloses, à travers leur définition ainsi que l'état de l'art associé. Pour la première, nous nous concentrons davantage sur la segmentation en mots à l'aide de modèles bayésiens non paramétriques et mentionnerons brièvement la segmentation en morphèmes. Quant à la seconde, après avoir exposé l'intérêt des gloses pour la documentation mais aussi pour le TAL, nous détaillerons les différents défis et techniques utilisées. Nous décrivons également la documentation automatique des langues ainsi que les domaines connexes de la préservation et de la revitalisation.

Le chapitre 3 se focalise sur les différentes ressources linguistiques accessibles, en mettant l'accent sur celles que nous avons étudiées dans nos travaux. En effet, pour les langues en cours de documentation en particulier, peu de données sont disponibles en ligne et il est nécessaire de

s'appuyer sur les travaux des linguistes à travers, entre autres, des corpus de documentation, des grammaires ou des archives.

Le chapitre 4 s'intéresse à la prise en charge de différentes formes de supervision faible pour améliorer la segmentation en mots, en se basant sur des ressources linguistiques dans les projets de documentation, accessibles de manière réaliste. De plus, nous expérimentons avec un cadre d'apprentissage incrémental, en vue d'une intégration dans une plateforme d'annotation, où un expert de la langue corrigerait progressivement les phrases segmentées. Enfin, nous effectuons une ébauche d'analyse quant à la nature des unités obtenues.

Le chapitre 5 introduit un deuxième niveau de segmentation, les morphèmes, afin de permettre une meilleure distinction entre ces unités. Nous concevons différents types de modèles en tenant compte de la relation entre les mots et les morphèmes. Nous vérifierons également les distributions des unités dans les corpus étudiés, afin de vérifier les hypothèses sous-jacentes du modèle.

Le chapitre 6 aborde une autre étape de la documentation automatique, la génération de gloses, à l'aide d'un modèle statistique d'étiquetage de séquences permettant de restreindre l'ensemble des étiquettes possibles pour une phrase donnée. Celui-ci est supervisé par des alignements automatiques entre les gloses lexicales et la traduction.

Enfin, le chapitre 7 clôt cette thèse en récapitulant les principales conclusions observées dans nos expériences et analyses subséquentes. Après avoir esquissé les limites de nos approches, nous suggérons également les perspectives pour les deux tâches traitées en vue d'une meilleure automatisation de la documentation des langues.

1.4 Publications

Les travaux effectués dans le cadre de la thèse ont mené aux publications répertoriées ci-dessous :

— Shu Okabe, François Yvon et Laurent Besacier. 2021. Segmentation en mots faiblement supervisée pour la documentation automatique des langues. In *Journées du Groupement de Recherche Linguistique Informatique, Formelle et de Terrain (LIFT)*, Grenoble, France. CNRS. (Okabe et al., 2021) ;

— Shu Okabe, Laurent Besacier et François Yvon. 2022. Weakly supervised word segmentation for computational language documentation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, pages 7385–7398, Dublin, Ireland. Association for Computational Linguistics. (Okabe et al., 2022) ;

— Shu Okabe et François Yvon. 2022. Modèle-s bayés-ien-s pour la segment-ation à deux niveau-x faible-ment super-vis-é-e (Bayesian models for weakly supervised two-level segmentation). In *Actes de la 29e Conférence sur le Traitement Automatique des Langues Naturelles. Volume 1 : conférence principale*, pages 174–182, Avignon, France. ATALA. (Okabe et Yvon, 2022a) ;

— Shu Okabe et François Yvon. 2022. Vers la génération automatique de gloses pour la documentation automatique des langues. In *Journées Jointes des Groupements de Recherche Linguistique Informatique, Formelle et de Terrain (LIFT) et Traitement Automatique des Langues (TAL)*, pages 198–203, Marseille, France. CNRS. (Okabe et Yvon, 2022b) ;

— Shu Okabe et François Yvon. 2023. Joint word and morpheme segmentation with Bayesian non-parametric models. In *Findings of the Association for Computational Linguis-*

tics : EACL 2023, pages 640–654, Dubrovnik, Croatia. Association for Computational Linguistics. (Okabe et Yvon, 2023a) ;

— Shu Okabe et François Yvon. 2023. Production automatique de gloses interlinéaires à travers un modèle probabiliste exploitant des alignements. In *Actes de CORIA-TALN 2023. Actes de la 30e Conférence sur le Traitement Automatique des Langues Naturelles (TALN), volume 1 : travaux de recherche originaux – articles longs*, pages 262–274, Paris, France. ATALA. (Okabe et Yvon, 2023c) ;

— Shu Okabe et François Yvon. 2023. LISN @ SIGMORPHON 2023 shared task on interlinear glossing. In *Proceedings of the 20th SIGMORPHON workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 202–208, Toronto, Canada. Association for Computational Linguistics. (Okabe et Yvon, 2023b) ;

— Shu Okabe et François Yvon. Towards multilingual interlinear morphological glossing. In *Findings of the Association for Computational Linguistics : EMNLP 2023*, pages 5958–5971, Singapore. Association for Computational Linguistics. (Okabe et Yvon, 2023d).

Chapitre 2

Un aperçu de la documentation automatique des langues

2.1 Introduction

Dans ce chapitre, nous commençons par donner une vue d'ensemble sur la documentation automatique des langues, avec les différentes initiatives de TAL, mais aussi en évoquant les disciplines connexes de préservation et revitalisation des langues. Puis, nous présentons les deux tâches principalement au cœur de cette thèse : la segmentation de séquences ainsi que la génération de gloses, qui interviennent toutes deux après la transcription des enregistrements, lors de la documentation automatique des langues. Notons que les présentations se feront sous le prisme du TAL, tout au long de cette thèse.

Par ailleurs, nous mentionnons également des travaux sur des langues peu dotées, comme le définit par exemple (Berment, 2004) ; en effet, elles observent des défis similaires d'un point de vue du TAL, comme les contraintes dues aux tailles de données ou la présence numérique. Néanmoins, les langues peu dotées en TAL désignent également des langues avec plusieurs millions de locuteurs (donc avec un meilleur degré de vitalité) et qui disposent de ressources en bien plus grande quantité, ce qui contraste avec les langues en danger.

De fait, Joshi et al. (2020) observent six catégories de langues étudiées en TAL selon la quantité de ressources disponibles, qui sont soit étiquetées, dans des archives linguistiques et dépôts de données, soit non étiquetées, en l'occurrence dans Wikipédia. Chacune de ces classes bénéficie différemment des dernières avancées de TAL : la troisième plus dotée (« les étoiles montantes ») dispose par exemple d'une quantité suffisante de données non étiquetées, facilitant alors l'utilisation des modèles pré-entraînés, ce qui ne sera pas le cas des deux dernières catégories [1], qui disposent de trop peu de données.

2.1.1 Les différentes étapes de la documentation

La figure 2.1 présente les principales étapes de la documentation des langues à travers l'exemple d'une phrase. Dans la plupart des cas, les langues étudiées sont orales : les linguistes de terrain effectuent alors en premier lieu des enregistrements des locuteurs et assemblent un corpus audio (**S0**). Il est primordial de préciser ici que l'enregistrement d'une phrase clairement délimitée comme en **S0** est le résultat de plusieurs traitements effectués en amont sur l'enregistrement fait sur le terrain. En effet, il est nécessaire de le nettoyer en séparant notamment le bruit ambiant de la voix et de découper le fichier en phrases.

1. Les langues que nous étudions (section 3.2) font partie de ces deux dernières catégories ou ne sont pas présentes dans cette taxonomie ; https://microsoft.github.io/linguisticdiversity/assets/lang2tax.txt.

S0	Enregistrement	fichier audio		
S1	Phrase non segmentée	kɤndʑixtɤɣχsɯmpjɤtunɯ		
S2	Segmentation en mots	kɤndʑixtɤɣ	χsɯm	pjɤtunɯ
S3	Segmentation en mots et morphèmes	kɤndʑi–xtɤɣ	χsɯm	pjɤ–tu–nɯ
S4	Glose	COLL–brother	three	IFR.IPFV–exist–PL
S5	Traduction (EN)	*There were three brothers*		

FIGURE 2.1 – Les principales étapes de la documentation des langues. Phrase en japhug extraite de (Jacques, 2021).

L'étape suivante consiste en la transcription de la phrase, souvent de manière (presque) phonétique, en se basant également sur l'alphabet phonétique international. Le choix dépend par exemple de certaines conventions existantes pour la famille de langues et des choix des linguistes. Il faut noter que le format que nous présentons ici est adapté pour ce type de transcriptions. Deux choix de transcription sont alors possibles. Dans le premier cas, que nous considérons dans cette thèse, une automatisation partielle est tout d'abord effectuée par des outils de reconnaissance vocale ; ces derniers, faute de ressources, produisent des transcriptions phonétiques et non segmentées (**S1**), qui nécessitent d'être corrigées par la suite. Sinon, dans le cas manuel classique, le linguiste (ou l'annotateur) transcrit directement à partir de l'enregistrement, vers une phrase segmentée en mots (**S2**), sous forme phonétique ou alphabétique. Nous pouvons remarquer que les mots de la phrase sont alors segmentés, séparés ici par un espace (« »).

Ensuite, les mots sont segmentés en morphèmes (**S3**), les plus petites unités porteuses de sens dans la langue (en français, le mot « langues » serait segmenté en « langue–s »), et séparés par des tirets (« – »)[2]. Nous soulignons ici que la séparation de morphèmes peut être indiquée par d'autres symboles dans certains cas spécifiques, comme le « = » pour les clitiques ou le « ~ » pour le redoublement. Dans cette thèse, nous uniformisons les séparations de morphèmes par des tirets.

La phrase est par la suite glosée (**S4**) : les gloses sont des annotations linguistiques. Pour chaque morphème source, il y a, en effet, soit une indication de son rôle grammatical (comme PL pour pluriel), soit sa signification dans la langue de documentation (par exemple, brother). Une description plus détaillée des gloses se trouve à la section 2.3 ci-dessous. Nous pouvons aussi noter qu'une glose peut comporter plusieurs informations, séparées par un point (« . »), comme IFR.IPFV (pour inférentiel et imperfectif respectivement) dans l'exemple, quand il est impossible de décomposer la forme sonore pour chacune de ses fonctions. Il existe de même des combinaisons de gloses lexicales et grammaticales, comme be.3SG.IPFV pour le mot « était » en français, étant donné qu'il n'est pas possible de segmenter en unités plus petites.

Enfin, la phrase est accompagnée de sa traduction (**S5**) dans une langue plus dotée, la langue de documentation (ici, l'anglais). Notons que la documentation peut également se faire à travers la langue majoritaire locale, autre que l'anglais, comme nous le verrons dans quelques corpus étudiés. Les traductions représentent l'autre strate habituellement présente dans les corpus de documentation, du fait de certaines consignes (Wittenburg et al., 2002), en étant notamment enregistrées à la suite des exemples dans la langue étudiée, lorsque les informateurs sont bilingues. Par ailleurs, le style de traduction semble varier en fonction des annotateurs ; si nous pouvons observer des tournures proches de l'expression en langue source, les traductions plus idiomatiques semblent plus répandues du fait des différences linguistiques.

2. Notons que cette représentation est adaptée aux langues ayant une morphologie plutôt concaténative.

Il est nécessaire de souligner que d'autres strates d'annotations, plus ou moins fréquemment utilisées, existent (voir (von Prince et Nordhoff, 2020) pour une description plus détaillée), comme les étiquettes en parties du discours (ou, en anglais, *Part-of-Speech*, PoS) ; la figure 2.1 ne présente que celles que nous avons étudiées dans le cadre de cette thèse.

Pour la suite, nous adoptons les conventions suivantes : les mots sources sont laissés tels quels (χsɯm), les gloses grammaticales sont en petites capitales (PL) tandis que les gloses lexicales sont dans une police différente (brother). Enfin, les mots de la traduction dans la langue de documentation sont en italique (*brothers*).

2.1.2 Les initiatives en place d'un point de vue TAL

Si les initiatives de documentation entreprises par le domaine de la linguistique sont multiples (comme nous le verrons en section 3.1.3, à travers quelques archives en ligne), nous nous intéressons ici aux différents travaux effectués en coopération avec la communauté TAL, au-delà de notre projet CLD2025. Plus particulièrement, nous nous concentrons sur l'automatisation intervenant lors de la création de ressources pour les langues en danger.

Le projet BULB Le projet franco-allemand *Breaking the Unwritten Language Barrier* (ou BULB) [3] (Adda et al., 2016) est un exemple de documentation automatique des langues, à travers la collaboration entre la linguistique de terrain et le domaine du TAL, où trois langues bantoues, à tradition orale, ont été documentées.

Tout d'abord, plusieurs dizaines d'heures d'enregistrement avec leur traduction ont été collectées sur place à l'aide de l'application mobile LIG-Aikuma [4] (Blachon et al., 2016), une version augmentée de Aikuma (Bird et al., 2014), qui est un outil d'enregistrement adapté à la documentation. Par rapport à l'implémentation originale qui proposait déjà certaines fonctionnalités supplémentaires comme l'enregistrement des traductions en suivant (Hanke et Bird, 2013), ou les répétitions (*respeaking* en anglais, qui consiste en la répétition plus lente de l'énoncé à oral afin de faciliter la transcription ; (Woodbury, 2003)), LIG-Aikuma permet également d'effectuer des élicitations, à savoir une lecture d'un texte ou un commentaire de photo et de vidéo, et présente une meilleure ergonomie. Le corpus constitué lors du projet est présenté en section 3.1.1.

Ensuite, des étapes comme la transcription automatique ou la segmentation en mots sont traitées. La transcription a notamment été effectuée de manière phonétique (voir section 2.2.1). Puis, les phrases ainsi transcrites ont été segmentées en utilisant différents modèles bayésiens non paramétriques (Godard, 2019), comme nous le verrons en section 2.2. Par ailleurs, l'enregistrement a été aligné avec les mots de la traduction de manière automatique. Enfin, le troisième pilier du projet est consacré à l'implémentation d'outils afin de faciliter les tâches d'annotations pour le linguiste. Notons également qu'un objectif complémentaire du projet est d'expérimenter un procédé permettant d'étendre les modèles TAL aux langues à tradition orale en général.

Le projet AGGREGATION Le projet AGGREGATION [5] de l'université de Washington a pour objectif d'assister la documentation des langues à travers une extraction de grammaires. Pour ce faire, les données au format structuré IGT (phrase source, annotations linguistiques et traduction

3. ANR-14-CE35-002 ; https://www.bulb-project.org/.
4. https://lig-aikuma.imag.fr/.
5. ***A**utomatic **G**eneration of **GR**ammars for **E**ndangered Languages from **G**losses **A**nd **T**ypological **I**nformati**ON*** ; https://depts.washington.edu/uwcl/aggregation/.

correspondante ; voir section 2.3.1) servent de pont entre la langue d'étude et une langue plus dotée, afin de permettre par exemple une projection de structure d'une langue à une autre, à la manière de (Xia et Lewis, 2007). Comme les phrases sous ce format se trouvent pour le moment principalement dans les grammaires et autres documents linguistiques, elles sont extraites à partir des PDF afin de constituer une base de données massive d'IGT, appelée ODIN (Lewis, 2006; Lewis et Xia, 2010) (voir section 3.1.2).

Projets connexes Un projet connexe à la documentation automatique des langues, est DoReCo (*DOcumentation REference COrpus*) [6] (Paschen et al., 2020) qui a abouti à la constitution d'un corpus multilingue (51 langues peu dotées de différentes familles linguistiques) à partir d'enregistrements extraits de dépôts déjà existants de documentation des langues, sous un format standardisé. Chaque fichier audio est alors aligné avec sa transcription au niveau des mots, avec une vérification manuelle. À travers cette harmonisation, l'idée est de favoriser les études entre langues de phénomènes comme les variations dans l'allongement phonétique (par exemple, aux positions finales dans Paschen et al. (2022)) ou la densité d'information dans les morphèmes. Ce projet est complémentaire à notre projet CLD2025, dans la mesure où il interviendrait à l'étape succédant aux annotations linguistiques (éventuellement pré-générées automatiquement).

Un autre projet, cette fois toujours en cours, est Autogramm [7] (Kahane et al., 2023) qui vise notamment à automatiser la création de treebanks pour des langues en cours de documentation, à travers la collaboration avec les linguistes. L'objectif à terme est de pouvoir extraire automatiquement des grammaires à partir de ces treebanks (ou des corpus). En suivant ce procédé, comme les annotations suivront alors les mêmes conventions, en particulier *Surface Syntactic UD* (SUD) (Gerdes et al., 2018, 2019), les règles de grammaire seront comparables entre les langues et les corpus, afin de faciliter le travail de typologie quantitative.

Les ateliers et défis partagés de TAL Il existe également des initiatives dans la communauté TAL à travers des ateliers (*workshops*) ou des défis partagés (*shared tasks*), sur des thèmes certes parfois plus généraux que la documentation, mais qui rejoignent néanmoins des problématiques communes aux langues peu dotées. Tout d'abord, l'atelier Field Matters [8], qui a lieu durant des conférences internationales de TAL (deux éditions à ce jour et une troisième en 2024), s'intéresse au rapprochement entre la linguistique de terrain, comme son nom l'indique, et le domaine du TAL, afin de faciliter les tâches de recueil et d'annotations notamment. De plus, en France, nous pouvons mentionner le groupement de recherche LIFT [9] (Linguistique Informatique, Formelle et de Terrain), qui vise à encourager les collaborations entre ces trois disciplines, ce qui inclut notamment les initiatives de documentation automatique des langues. Au-delà des journées de recherche tenues annuellement, des écoles thématiques ainsi qu'un séminaire mensuel en ligne (ILFC) sont également organisés. Enfin, le *Workshop on Language Technology for Language Documentation and Revitalization* (Neubig et al., 2020) de l'université Carnegie-Mellon a été un atelier en 2019, afin de rapprocher les chercheurs en TAL de l'université avec des linguistes en documentation ainsi que des membres des communautés parlant les langues étudiées. Quatre axes de recherche y ont été discutés, en recourant aux avancées récentes en TAL : les technologies de la parole pour la transcription, le traitement des dictionnaires pour une meilleure valorisation, la création d'outil

6. ANR-18-FRAL-0010-01 ; https://doreco.info/.
7. ANR-21-CE38-0017 ; https://autogramm.github.io/.
8. https://field-matters.github.io/.
9. https://gdr-lift.loria.fr/.

de recherche dans les corpus pour faciliter l'enseignement de la langue ainsi que l'utilisation des réseaux sociaux pour la documentation et la revitalisation.

Les conférences internationales de TAL voient également des ateliers focalisés sur les langues en danger comme l'atelier ComputEL [10] (« *Computational Methods for Endangered Languages* »), qui a eu lieu lors de la conférence ACL pour l'édition 2022. De plus, récemment, le groupe d'intérêt SIGUL [11] (*Special Interest Group on Under-Resourced Languages*) dédié aux langues peu dotées a organisé un atelier (Melero et al., 2022), où ont été également publiés des articles sur des langues en danger. Par ailleurs, nous pouvons mentionner le thème spécial sur la diversité des langues lors de la conférence ACL 2022 (« *Language Diversity: from Low-Resource to Endangered Languages* »). Ces cadres permettent ainsi de favoriser la publication de travaux spécifiquement pour des langues peu dotées ou en danger, comme par exemple un analyseur morphologique pour des langues avec une morphologie très riche (polysynthétique) (Lane et Bird, 2020). Cela met alors en lumière d'autres types de défis et approches en TAL, en adéquation avec les particularités des langues.

Enfin, il existe des organisations dans le domaine du TAL pour certains groupes de langues sous-dotées comme Masakhane [12] pour les langues africaines ou AI4Bharat [13] pour les langues du sous-continent indien. L'objectif est d'étendre les modèles de TAL vers ces langues qui disposent certes de moins de ressources comparativement aux autres langues fréquemment traitées en TAL, mais qui sont néanmoins parlées par des millions de locuteurs.

Apprentissage actif En complément, nous présentons également une manière d'intégrer les approches TAL dans le processus de documentation des langues, qui est de recourir à un apprentissage actif : grâce à l'interaction (notamment des corrections d'erreurs) entre les annotateurs et le modèle, ce dernier peut être amélioré progressivement. Palmer et al. (2009) et Baldridge et Palmer (2009) expérimentent cette configuration pour la tâche de génération de gloses. Deux linguistes, l'un expert de la langue et l'autre non, effectuent des annotations linguistiques sur le corpus suivant différentes configurations d'automatisation : soit l'annotateur a accès à la prédiction du modèle, accompagnée des probabilités associées, soit les étiquettes précédemment utilisées pour le morphème sont affichées par ordre de fréquence (un système reproduisant celui des outils d'annotation actuels). De plus, l'exemple à annoter est choisi de trois manières : de façon aléatoire, séquentielle (c'est-à-dire, la phrase suivante dans le corpus) ou selon la difficulté pour le modèle (l'idée ici est d'avoir un meilleur apprentissage). Le coût est mesuré par le temps nécessaire à l'annotation, ce qui permet d'englober également le coût financier. Bien que l'expérience compte seulement deux annotateurs, elle donne une idée de la complexité et de l'impact de la tâche : la configuration optimale dépend par exemple fortement de l'expertise de l'annotateur.

Par ailleurs, pour le domaine acoustique, Le Ferrand et al. (2020) proposent une méthodologie incorporant des locuteurs natifs lors de la transcription de mots dans les langues très peu dotées. Une application mobile permet de vérifier manuellement si les transcriptions écrites correspondent à l'enregistrement. À travers ce processus itératif, le modèle de reconnaissance de termes parlés (*Spoken Term Detection*, en anglais) voit ses performances renforcées, malgré un jeu de données de taille restreinte.

10. https://computel-workshop.org/
11. https://www.sigul.eu/.
12. https://www.masakhane.io/.
13. https://ai4bharat.iitm.ac.in/.

2.1.3 La préservation et la revitalisation des langues

Comme nous avons pu le voir avec l'initiative jointe de documentation et de revitalisation présentée précédemment (Neubig et al., 2020) ou avec les plateformes de ressources à vocation multiple comme l'*Endangered Languages Project*[14] par exemple, la documentation des langues et la préservation ou la revitalisation des langues partagent de nombreux points communs, à commencer par la volonté de répondre aux menaces pesant sur les langues en danger. Cependant, il faut les distinguer, malgré leurs proximités ; en effet, comme l'indique Yamada (2011), la linguistique s'intéresse davantage à des thématiques qui ne sont pas nécessairement les plus adaptées pour la revitalisation. Par ailleurs, s'il existe une méthodologie séquentielle qui considère la documentation, puis la description et ensuite la revitalisation, Evans (2008) recommande plutôt une approche simultanée de documentation et de description, à laquelle Taylor-Adams (2019) ajoute la revitalisation, car toutes trois complémentaires. Nous présentons ici les deux domaines brièvement, mis en perspective avec la documentation ou l'utilisation d'outils de TAL.

La préservation ou la maintenance des langues désigne communément les efforts visant à maintenir l'effectif des locuteurs ; la langue est encore relativement vitale, bien que des signes de déclin puissent être parfois visibles. D'autre part, le terme de revitalisation est principalement employé lorsque la langue est en danger, pour désigner les actions entreprises afin d'inverser la chute du nombre de locuteurs et d'encourager l'utilisation de la langue (Hinton, 2011). Elle peut être effectuée à travers différents moyens comme le recueil de la langue auprès des locuteurs natifs ou l'apprentissage de la langue par les générations suivantes ; il s'agit donc en grande partie d'enseigner la langue à ceux qui ne la parlent pas et d'inciter les locuteurs natifs ainsi que les apprenants à l'utiliser davantage, selon (Hinton, 2011). Pérez Báez et al. (2018) effectuent à ce titre une revue approfondie de la littérature de revitalisation de la langue.

Des travaux parviennent à concilier la documentation aux efforts de revitalisation en coopération avec les locuteurs et les enseignants. Yamada (2011) présente par exemple une approche où le corpus de documentation est constitué en ayant aussi pour objectif de servir de matériel pédagogique pour l'enseignement du kali'na (langue caribe ; `car`). Taylor-Adams (2019) souligne également les bénéfices mutuels potentiels d'une documentation effectuée conjointement à une revitalisation. En effet, après avoir consulté dix enseignants de langue dans le cadre de leur activité de revitalisation, les divergences entre leurs attentes et les ressources de documentation semblent apparentes : par exemple, les grammaires ne sont pas nécessairement adaptées à l'enseignement pour des non linguistes pour le moment.

Par ailleurs, Pérez Báez et al. (2019) présentent les résultats de leur enquête concernant la revitalisation de la langue. Ils observent notamment que les projets actuels se concentrent principalement sur l'enseignement de la langue (par rapport à d'autres objectifs comme la transmission entre générations ou la diffusion de la langue).

C'est pourquoi la revitalisation semble avant tout bénéficier d'outils informatiques dédiés à l'enseignement, en favorisant un format plus pédagogique. En ce sens, Littell et al. (2018), qui recensent les technologies développées pour les différentes langues autochtones au Canada, catégorisent notamment les outils numériques d'apprentissage de langue comme étant disponibles pour beaucoup des langues étudiées, comparativement à d'autres outils encore balbutiants (la traduction automatique ou la reconnaissance de la parole). Neubig et al. (2020) conçoivent par exemple un outil de recherche dans les corpus (en utilisant entre autres des plongements lexicaux) ou une première version d'agent conversationnel pour traduire des mots et faciliter l'utilisation ainsi que

14. `https://www.endangeredlanguages.com/`.

l'apprentissage de la langue. Ní Chiaráin et al. (2022) ont créé une plateforme intégrée pour l'apprentissage du gaélique irlandais (`gle`), en recourant à des méthodes de TAL comme la synthèse vocale. D'autres approches pédagogiques récentes existent telles que des jeux interactifs, à nouveau pour le gaélique irlandais (Xu et al., 2022).

Par ailleurs, Mehta et al. (2020) étudient le déploiement d'outils TAL pour le gondi (`gon`), une langue parlée en Inde. Pour la communauté, l'objectif a été d'obtenir des ressources utilisables au quotidien comme un dictionnaire disponible sur le téléphone ou des livres traduits pour enfants, là où d'un point de vue TAL, ces travaux constituent aussi une première étape vers l'obtention de données sur lesquelles entraîner des modèles de reconnaissance de la parole ou de traduction automatique, afin de pouvoir proposer ces outils ultérieurement.

Enfin, nous pouvons également mentionner les initiatives visant à mieux intégrer la langue dans les systèmes de TAL à terme, comme (James et al., 2022) qui crée un outil de détection de langue et d'alternance codique (*code switching* en anglais) pour le maori de Nouvelle-Zélande (`mri`), une langue en cours de revitalisation.

2.2 La segmentation de séquences

Cette section s'intéresse à la tâche de segmentation des séquences, principalement des phrases en mots, mais également des mots en morphèmes. Après l'avoir définie, nous présentons les différentes méthodes bayésiennes non paramétriques, majoritairement non supervisées, utilisées dans le cadre de la documentation automatique des langues. En complément, nous mentionnons les approches neuronales de segmentation à partir de données textuelles, mais également acoustiques. Enfin, en prévision de la tâche de segmentation simultanée en mots et en morphèmes du chapitre 5, nous exposons succinctement la tâche connexe de segmentation en morphèmes.

2.2.1 La segmentation en mots pour la documentation

L'objectif de la segmentation en mots est d'identifier les frontières des mots d'une langue donnée dans une séquence de symboles non segmentée, correspondant à un énoncé, comme la strate **S1** dans l'exemple 2.1. Soulignons ici que les mots indiquent les unités qui sont communément considérées comme mots, notamment par les linguistes dans le cadre de la documentation des langues. Dans notre cas, cette tâche se concentre donc sur la reproduction de cette convention. Nous utiliserons plutôt le terme de segments pour qualifier les unités obtenues par les modèles de segmentation, qui peuvent correspondre ou non aux mots dans la langue.

Parmi les différentes manières de représenter cette tâche, nous considérons la suivante : nous attribuons à chaque caractère c_i de la séquence, une variable binaire b_i indiquant la présence ($b_i = 1$) ou l'absence ($b_i = 0$) de frontières après celle-ci, comme illustré en figure 2.2. Puisque la chaîne de caractères à segmenter correspond à une phrase, la dernière position a nécessairement pour valeur $b_i = 1$, car la fin d'une phrase est également une fin de mot. En regroupant les plages de caractères délimitées par les positions où $b_i = 1$, nous pouvons retrouver les segments.

Provenance de la transcription Dans le cadre du projet BULB (Adda et al., 2016), dont le projet CLD2025 est le successeur, la transcription s'effectue en trois temps à partir de l'enregistrement de la phrase :

1. identification des frontières des phones uniquement, avec des réseaux *Long short-term memory* (LSTM) bidirectionnels (Franke et al., 2016) ;

i	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20
c_i	k	ʏ	n	d	ʑ	i	x	t	ʏ	ɣ	χ	s	ɯ	m	p	j	ʏ	t	u	n	ɯ
b_i	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	1

FIGURE 2.2 – Représentation de la phrase de l'exemple 2.1 « kʏndʑixtʏɣ χsɯm pjʏtunɯ », segmentée à travers la variable de frontières $\boldsymbol{b}$.

2. un étiquetage de chaque segment à travers les caractéristiques articulatoires (Müller et al., 2017b) ;
3. un regroupement de ces segments vers des phones, en s'appuyant sur les caractéristiques extraites (Müller et al., 2017a).

La transcription obtenue est donc non segmentée et correspond à une représentation phonétique. Par conséquent, elle peut contenir du bruit, un scénario étudié dans (Godard et al., 2018a) par exemple. Toutefois, notre approche se fait dans un cadre plus favorable, où la transcription est considérée comme correcte et nécessite seulement d'être segmentée, comme dans (Godard et al., 2016, 2018b) notamment.

La tâche de segmentation en mots diffère de la tokenisation, courante en TAL. Si la segmentation en unités BPE (*byte pair encoding*, en anglais) (Sennrich et al., 2016) ou le modèle SentencePiece (Kudo, 2018) permettent effectivement d'obtenir des séquences de sous-mots pour une phrase, ces segments ne correspondent pas nécessairement à des sous-unités pertinentes de la langue et ne constituent pas une fin en soi.

2.2.2 Approches antérieures

L'un des premiers à s'intéresser à une segmentation de séquences est (Harris, 1955), qui repose sur la différence dans la facilité de prédiction à une frontière entre morphèmes par rapport à une position au sein d'un morphème (par exemple, il est plus facile de prédire la lettre succédant au « r » qu'au « i » dans « écri**r**e », « écri**v**ain », « écri**t**ure »). D'autres modèles existent comme le modèle de (Bimbot et al., 1995) reposant sur le modèle de langue multigramme de (Deligne et Bimbot, 1995), où une phrase est notamment considérée comme une séquence d'unités dont la longueur maximale est fixée.

Une autre approche de segmentation repose sur le principe de longueur de description minimale (*Minimum Description Length* en anglais, MDL) (Rissanen, 1989), qui suppose que l'hypothèse optimale expliquant des données est celle qui permet de les compresser le plus. Les hypothèses correspondent ici à une méthode d'encodage des données vers des bits, ce qui permet de quantifier et d'évaluer les différentes méthodes. En notant $\mathcal{D}$ les données et L la fonction de longueur (ou longueur de description), traduisant la compression, l'hypothèse optimale H^* parmi l'ensemble des hypothèses $\mathcal{H}$ est obtenue par l'équation (2.1) :

$$H^* = \arg\min_{H \in \mathcal{H}} (L(D|H) + L(H)) \tag{2.1}$$

Ici, $L(D|H)$ indique la longueur associée aux données lorsqu'elle est expliquée par l'hypothèse H, donc il s'agit de l'encodage, là où $L(H)$ indique le code correspondant à cette hypothèse. Intuitivement, pour la segmentation en mots, il s'agit de trouver un compromis d'encodage entre le niveau le plus fin des caractères, où le code pour chaque caractère sera petit ($L(H)$), mais l'encodage des phrases sera long ($L(D|H)$), et celui des mots, où l'encodage sera plus court mais le code sera

plus long. Ce modèle a été et est toujours utilisé pour les différents niveaux de segmentation ; en particulier au niveau des morphèmes, avec Morfessor (Creutz et Lagus, 2002, 2007), une approche non supervisée fréquemment utilisée comme modèle de référence (voir section 2.2.8).

D'autres modèles de segmentation de séquences sont également présentés dans (Godard, 2019).

2.2.3 Le modèle bayésien non paramétrique dpseg

Le modèle de segmentation en mots bayésien non paramétrique [15] `dpseg` (Goldwater et al., 2009) a initialement été conçu dans le cadre des travaux visant à comprendre l'acquisition de la langue chez les enfants en bas âge. Au lieu d'essayer de reproduire les mécanismes à l'œuvre pour détecter les mots à partir d'informations statistiques comme (Saffran et al., 1996) qui s'appuie notamment sur l'évolution de la probabilité de transition entre symboles, l'idée est de mettre en évidence les hypothèses à prendre en compte pour effectuer une segmentation en mots : soit les mots sont des unités indépendantes, comme dans le modèle unigramme, soit il existe une dépendance entre les unités, représentée par le modèle bigramme. Nous présentons ici principalement le modèle unigramme, que nous utilisons dans nos travaux.

L'une des forces de ce modèle réside dans la possibilité de modéliser une distribution en loi de puissance, plus particulièrement la loi de Zipf (Zipf, 1935), qui apparaît dans les langues naturelles. En effet, celle-ci est visible en classant la fréquence des mots dans un corpus : il existe une petite quantité de mots très fréquents pour un grand nombre de mots rares.

La présentation dans cette section s'appuie sur (Goldwater, 2006) et (Goldwater et al., 2009).

***Chinese Restaurant Process* (CRP)** Afin de comprendre le processus du restaurant chinois, son appellation nous invite à visualiser un restaurant, avec un nombre théoriquement infini de tables. Dès lors qu'un client entre, deux choix se présentent à lui : soit il s'assied à une table déjà occupée par d'autres clients, soit il s'installe à une nouvelle table et reste, pour le moment, seul. Toutefois, il faut supposer que la probabilité de s'attabler à une table déjà existante croît en fonction du nombre de clients qu'il y rejoint et que les chances de le voir s'installer à une table vide sont régies par un paramètre de concentration α (≥ 0). Par cette définition, ce processus modélise le phénomène des « plus riches devenant plus riches » (*rich-get-richer*, en anglais), comme les tables les plus remplies ont tendance à attirer plus de nouveaux clients.

En s'appuyant sur cette analogie, formellement, le processus du restaurant chinois (CRP) est défini par l'équation (2.2) ci-dessous, caractérisant comment le client i choisit sa table z_i, en tenant compte de la disposition $\boldsymbol{z_{-i}}$ des clients antérieurs à leur table respective dans le restaurant :

$$P(z_i = k|\boldsymbol{z_{-i}}) = \begin{cases} \dfrac{n_k(\boldsymbol{z_{-i}})}{i-1+\alpha} & \text{, si } 1 \leq k \leq K(\boldsymbol{z_{-i}}) \\ \dfrac{\alpha}{i-1+\alpha} & \text{, si } k = K(\boldsymbol{z_{-i}})+1 \end{cases} \tag{2.2}$$

k indique l'indice de la table choisie, avec $K(\boldsymbol{z_{-i}})$ le nombre total de tables (non vides) à l'arrivée du client i et $n_k(\boldsymbol{z_{-i}})$ représente le nombre de clients déjà assis à la table d'indice k. L'équation (2.2) peut alors se lire comme suit :

— soit le client s'installe à une table déjà occupée (donc $k \in [1, K(\boldsymbol{z_{-i}})]$), avec une probabilité dépendant du nombre de ses convives $n_k(\boldsymbol{z_{-i}})$;

15. L'expression « non paramétrique » signifie ici non pas l'absence de paramètres mais que le modèle peut moduler le nombre de paramètres.

— soit il préfère une nouvelle table (indexée par $k = K(\boldsymbol{z_{-i}}) + 1$), en fonction du paramètre de concentration α.

Ce dernier permet de définir la propension du client à privilégier la solitude initiale plutôt que la compagnie des autres. De fait, plus ce paramètre est élevé, plus le client aura tendance à s'installer à une nouvelle table et donc plutôt lisser la répartition des individus dans le restaurant. À l'inverse, lorsqu'il est plus bas, nous observerons une plus forte *concentration* des clients sur un petit nombre de tables, accentuant l'effet *rich-get-richer*.

Notons par ailleurs que dans l'équation (2.2), le dénominateur $(i-1+\alpha)$ permet bien d'obtenir une probabilité, car le nombre de clients à l'arrivée du client i, $(i-1)$, est bien égal à la somme des personnes assises à chaque table du restaurant, $\sum_{k=1}^{K(\boldsymbol{z_{-i}})} n_k(\boldsymbol{z_{-i}})$.

Processus de Dirichlet Un processus de Dirichlet est défini par deux paramètres : une distribution de base G_0 et un paramètre de concentration α. Ce processus peut être considéré comme un modèle à « deux étages » fondé sur les CRP, comme le montre Goldwater et al. (2009) dans son annexe (A.1.2.). En effet, la distribution de base G_0 sert à générer les étiquettes associées aux tables et est donc, pour cette raison, nommée *générateur*. Les clients s'assoient alors suivant le CRP à des tables étiquetées par le générateur, adaptant ainsi leur distribution dans le restaurant : le CRP est donc appelé *adapteur* par Goldwater et al. (2009).

Dans le cadre de la segmentation en mots, les étiquettes des tables correspondent aux *types* de mots, tandis que les clients représentent leurs *occurrences*. Soulignons qu'une même étiquette peut être générée à plusieurs reprises pour être affectée à des tables différentes, comme nous pouvons le voir à la figure 2.3 pour le type « je ».

L'utilisation du processus de Dirichlet convient alors à la tâche de segmentation en mots, dans la mesure où elle permet de reproduire la distribution en loi de puissance, typiquement observée dans les distributions de mots dans les langues, aussi connue sous le nom de loi de Zipf (Zipf, 1935).

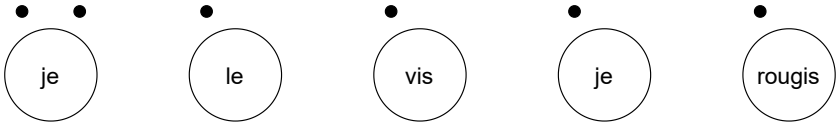


FIGURE 2.3 – Représentation d'un restaurant avec des clients (*occurrences* de mots) assis à des tables étiquetées par des *types* de mots. Illustration adaptée de (Goldwater et al., 2009).

Le modèle `dpseg` Le modèle de segmentation non supervisé `dpseg` (de l'anglais ***Dirichlet Process Segmentation***) (Goldwater et al., 2009) repose, comme son nom l'indique, sur un processus de Dirichlet ; dans le cas simple *unigramme*, celui-ci est défini par un paramètre de concentration α et une distribution de base P_0.

La distribution de base caractérise la probabilité de générer un nouveau mot w_i, constitué de la séquence de L caractères $c_1, ..., c_L$, à travers l'équation (2.3) ci-dessous, si le mot est généré à nouveau ; sinon, la probabilité d'un mot est déterminée par l'équation (2.4).

$$P_0(w_i) = P(w_i = c_1, ..., c_L | w_i \text{ nouveau}) = p_\#(1-p_\#)^{L-1} * \prod_{j=1}^{L} P_c(c_j) \tag{2.3}$$

$$P(w_i = l | w_i \text{ non nouveau}) = \frac{n_l}{n} \tag{2.4}$$

où $p_\#$ est la probabilité d'avoir une frontière de mot après un caractère et P_c indique la probabilité d'un caractère dans la langue. L'équation (2.3), en ce sens, se décompose en deux parties : le premier facteur représente un modèle de longueur, tandis que le second constitue un simple modèle unigramme de caractères. En effet, $p_\#(1-p_\#)^{L-1}$ traduit la probabilité d'engendrer un mot de longueur L avec une loi géométrique de paramètre $p_\#$, car la probabilité de ne pas avoir de frontière après un caractère est de $(1-p_\#)$. L'équation (2.4) correspond à la fréquence relative du mot l (de fréquence n_l) par rapport à tous les mots (n).

Les équations (2.2), (2.3) et (2.4) permettent de caractériser le processus de Dirichlet, vu comme un CRP à « deux étages ». La probabilité d'un mot w dans un agencement h^- du restaurant (donc de la segmentation du texte intégral, hormis la position considérée) peut s'écrire comme ci-dessous (les étapes intermédiaires sont en annexe A.1) :

$$P(w_i = w|\boldsymbol{w_{-i}}, \alpha) = \frac{n_w^{(\boldsymbol{w_{-i}})} + \alpha P_0(w)}{i - 1 + \alpha} \tag{2.5}$$

où $n_w^{(\boldsymbol{w_{-i}})}$ indique la fréquence du mot w dans le reste du texte (dénoté par $(\boldsymbol{w_{-i}})$).

Ainsi, l'équation (2.5) peut également être vue comme une combinaison d'un système de cache, où le mot est choisi parmi un lexique de mots connus ($n_w^{(\boldsymbol{w_{-i}})}$), et d'un terme caractérisant la génération du mot ($\alpha P_0(w)$). De fait, le paramètre de concentration contrôle alors directement la propension du modèle à générer de nouveaux mots (Goldwater et al., 2006). En particulier, Goldwater et al. (2011) montrent que l'utilisation d'un CRP répartit les unités de manière logarithmique aux tables du restaurant.

Pour traiter les cas des mots en fin de phrase, leur proportion est modélisée par un paramètre dédié, $p_\$$. Comme sa valeur est inconnue, nous avons recours à une probabilité a priori d'une distribution Beta($\frac{\rho}{2}$) symétrique. Le paramètre ρ correspond alors à la probabilité de finir une phrase. En suivant l'implémentation de Goldwater et al. (2009), nous utilisons $\rho = 2$.

Selon (Goldwater et al., 2009), cette configuration et son résultat sont déjà documentés dans la littérature baysésienne (Bernardo et Smith, 1994; Gelman et al., 2004) ; la probabilité que le i-ème mot soit une fin de phrase ($u_i = 1$) est alors définie par l'équation (2.6) :

$$P(u_i = 1|\boldsymbol{u_{-i}}, \rho) = \int P(u_i = 1|p_\$)P(p_\$|\boldsymbol{u_{-i}}, \rho)dp_\$ = \frac{n_\$ + \frac{\rho}{2}}{i - 1 + \rho} \tag{2.6}$$

avec $n_\$$ représentant le nombre total de phrases.

Inférence de dpseg Deux hypothèses sont alors considérées pour chaque position dans le texte : soit une frontière est présente après le caractère (hypothèse h_2 ; $b_i = 1$), soit ce n'est pas le cas (hypothèse h_1 ; $b_i = 0$). En comparant les probabilités de chacune de ces hypothèses, à savoir celle d'avoir deux mots de part et d'autre de la position actuelle (h_2) ou un seul mot (h_1), il est possible d'effectuer un tirage de la valeur de b_i. Par exemple, dans la figure 2.2, si l'on se situe à la position 12, l'hypothèse h_2 signifie une séquence « χsɯ m », là où h_1 correspond à « χsɯm ». Les détails des calculs sont à nouveau en annexe A.1.

En pratique, l'inférence repose sur un échantillonnage de Gibbs (Geman et Geman, 1984), une méthode de Monte-Carlo par chaînes de Markov. Étant donné que dans le cas de `dpseg`, la distribution a posteriori (pour l'hypothèse considérée conditionnellement aux données) est difficile à calculer directement, ce procédé permet de l'approcher, à partir d'une initialisation choisie des frontières, par itérations successives. À chaque position, la variable est tirée conditionnellement

aux autres variables (ici, l'état des frontières dans le reste du texte), comme dans l'algorithme 2.1. En effet, après une période au début, dite de *burn-in*, l'échantillonneur converge vers la distribution a posteriori, en théorie, quelle que soit l'initialisation utilisée pour les valeurs de frontières.

Algorithme 2.1 : Algorithme d'échantillonnage de Gibbs dans dpseg

```
pour n = 1...N faire                    /* N : nombre total d'itérations */
    pour i = 1...I faire                /* chaque caractère du texte */
        si b_i = 1 alors                /* présence d'une frontière */
            Enlever les clients w_2 et w_3 du restaurant associé au modèle;
        sinon                           /* pas de frontière */
            Enlever le client w_1 du restaurant;
        fin
        Calculer les probabilités P(b_i = 0|h^-) et P(b_i = 1|h^-);
        Tirer la nouvelle valeur de b_i par rapport à ces deux hypothèses;
        si b_i = 1 alors
            Ajouter les clients w_2 et w_3 au restaurant;
        sinon
            Ajouter le client w_1 au restaurant;
        fin
    fin
fin
```

Un des points faibles de l'échantillonnage de Gibbs réside dans sa vitesse de convergence, car les modifications de chaque itération sont locales. Afin de l'accélérer, dpseg utilise le recuit simulé (Aarts et Korst, 1989), qui permet notamment d'augmenter la probabilité de l'hypothèse la moins probable. Cette méthode introduit un paramètre de température τ, qui décroît au fil des itérations et qui modifie ici la probabilité de tirage P pour $\tilde{P}(x) = P(x)^{1/\tau}$. Le système explore alors davantage l'espace des possibilités au début, pour progressivement converger avec la baisse de température. Concrètement, Goldwater et al. (2009) utilisent dix paliers réguliers de température, variant de 0,1 à 1,0, ce qui implique que les probabilités calculées à chaque position sont mises à la puissance 0,1 jusqu'à 1,0.

Modèle bigramme Le modèle unigramme de dpseg repose notamment sur l'hypothèse simpliste d'indépendance entre les mots. Dans le cadre des travaux sur l'apprentissage du langage, Goldwater et al. (2009) observent que cette hypothèse conduit à une sous-segmentation des locutions (comme « parce que »). Pour remédier à ce phénomène, Goldwater (2006) conçoit également une version bigramme du modèle dpseg. Il s'agit alors du modèle de processus de Dirichlet hiérarchique (*Hierarchical Dirichlet Process*) (Teh et al., 2006) appliqué à la segmentation. En effet, la probabilité d'un mot w_i dépend désormais du mot précédent $w_{i-1} = l$, qui possède son propre restaurant où les clients suivent un processus de Dirichlet qui lui est spécifique (distribution H_l). Les étiquettes des restaurants bigrammes sont également répartis en suivant un processus de Dirichlet unique (distribution G) ; il s'agit ici du modèle unigramme avec une distribution de base P_0 représentant la probabilité de générer un mot.

Formellement, en reprenant les notations de (Goldwater et al., 2009), ce modèle peut être décrit par les distributions présentées en (2.7).

$$\begin{aligned} w_i|w_{i-1} = l, H_l &\sim H_l && \forall l \\ H_l|\alpha_2, G &\sim DP(\alpha_2, G) && \forall l \\ G|\alpha_1, P_0 &\sim DP(\alpha_1, P_0) \end{aligned} \tag{2.7}$$

L'inférence dans ce modèle se fait également avec un échantillonnage de Gibbs.

Si le modèle unigramme ne parvient pas à segmenter les locutions du fait de l'hypothèse d'indépendance des unités dans (Goldwater et al., 2009), le modèle bigramme ne subit pas ce problème et obtient des segmentations améliorées. Ce constat explique l'utilisation du modèle bigramme notamment dans (Godard et al., 2016) pour la segmentation de langues en cours de documentation. Toutefois, notons que dans (Godard, 2019), le modèle `dpseg` unigramme atteint de meilleurs résultats que son équivalent bigramme sur le même corpus.

2.2.4 Extensions du modèle dpseg

Processus de Pitman-Yor Nous caractérisons désormais le processus de Pitman-Yor (Pitman et Yor, 1997) avec l'équation (2.8), en conservant les notations vues en section 2.2.3.

$$P(z_i = k|\boldsymbol{z_{-i}}) = \begin{cases} \dfrac{n_k(\boldsymbol{z_{-i}}) - d}{i - 1 + \alpha} & \text{, si } 1 \leq k \leq K(\boldsymbol{z_{-i}}) \\ \dfrac{K(\boldsymbol{z_{-i}}) * d + \alpha}{i - 1 + \alpha} & \text{, si } k = K(\boldsymbol{z_{-i}}) + 1 \end{cases} \tag{2.8}$$

Le nouveau paramètre introduit ici, d, est appelé paramètre d'escompte et vérifie $0 \leq d < 1$ et $\alpha > -d$. Intuitivement, avec l'analogie du restaurant, nous comprenons que ce processus décourage le client d'aller vers une table déjà occupée et favorise l'installation à une nouvelle table. Pour la segmentation en mots, cela se traduit par une répartition plus lisse et une génération d'un nombre plus grand de nouveaux types. Notons, par ailleurs, que le processus de Dirichlet n'est qu'un cas particulier des processus de Pitman-Yor : si $d = 0$, l'équation (2.8) est identique à celle des processus du restaurant chinois avec l'équation (2.2).

Nous nous intéressons au processus de Pitman-Yor pour son aptitude à mieux reproduire une distribution en loi de puissance. En effet, comme Teh (2006b) le constate, il permet d'identifier davantage de types de mots que le processus de Dirichlet : pour un nombre de mot N tendant vers l'infini, le premier croît avec une vitesse de $\mathcal{O}(\alpha N^d))$, contre $\mathcal{O}(\alpha \ln(N))$ pour le second.

Teh (2006a,b) étend alors le processus de Dirichlet hiérarchique (Teh et al., 2006) en remplaçant le processus de Dirichlet par un processus de Pitman-Yor, pour définir un modèle de langue de Pitman-Yor hiérarchique. Ce modèle obtient notamment de meilleurs résultats, en comparant les différentes valeurs du paramètre d'escompte, $d > 0$ et $d = 0$. Cette extension est également utilisée pour la segmentation en morphèmes par (Goldwater et al., 2011).

Extensions basées sur le processus de Pitman-Yor En se basant sur les processus de Pitman-Yor décrits précédemment, Mochihashi et al. (2009) proposent une extension du modèle, notamment afin de répondre aux limites identifiées pour les modèles de Goldwater (2006), comme la lenteur de la convergence de l'échantillonneur de Gibbs ou la restriction à des dépendances bigrammes. Tout d'abord, l'approche repose sur l'utilisation d'un modèle hiérarchique fondé sur le processus de Pitman-Yor, non seulement pour les mots, mais également pour les caractères, à

travers la distribution de base, d'où son appellation de modèle imbriqué. De cette manière, de plus grandes dépendances, au-delà du bigramme, peuvent être modélisées, tandis que la génération des types de mots est améliorée. Cette dernière est également accompagnée d'une correction de la longueur selon la loi de Poisson, à la manière de Xu et al. (2008), car les mots longs obtenaient de trop faibles probabilités sinon. En outre, l'inférence est accélérée grâce à l'utilisation d'un échantillonnage de Gibbs bloqué : le tirage des variables de frontières $\boldsymbol{b}$ se fait non plus position par position mais à l'échelle de la phrase entière, à travers un algorithme forward-backward. Ce choix permet notamment de traiter plus facilement les dépendances plus longues que les bigrammes, bien qu'elles n'aient été explorées que jusqu'aux trigrammes dans ce travail. Cette approche parvient à atteindre plus rapidement de meilleurs résultats sur les mêmes données que Goldwater (2006). Par ailleurs, ce modèle est implémenté de manière plus efficace en utilisant un échantillonnage de Gibbs bloqué parallélisé dans (Neubig, 2014), ce qui accélère encore le calcul.

Uchiumi et al. (2015) présentent une extension fondée sur le modèle de (Mochihashi et al., 2009), afin de prédire de manière conjointe les frontières des mots avec leur étiquette PoS. L'idée est de pouvoir différencier les mots entre eux à travers leur catégorie et de modéliser également dans la phrase, les successions de PoS. Ces étiquettes PoS sont alors introduites comme variable latente du modèle. Celui-ci effectue donc tout d'abord un tirage de l'étiquette PoS, puis d'un mot sachant sa partie du discours. La différence avec le modèle original réside alors dans la dépendance vis-à-vis de la PoS dans le processus de Pitman-Yor pour les mots ainsi que dans la distribution Pitman-Yor (distincte) pour les étiquettes PoS elles-mêmes. Cet apprentissage simultané permet alors d'améliorer la qualité de segmentation par rapport à la segmentation en mots générique. Notons que Uchiumi et al. (2015) expérimentent également une configuration semi-supervisée en utilisant des phrases entièrement segmentées et étiquetées.

Enfin, Löser et Allauzen (2016) s'inscrivent dans cette continuité : bien que leur modèle soit originellement destiné à la segmentation morphologique, il est aisément transposable pour la segmentation de mots. Ici, il ne s'agit plus de PoS mais simplement de classes d'unités, en partant du constat que les affixes ou les racines apparaissent dans des contextes différents. Ce modèle génère tout d'abord une variable latente de classe de mots, puis un mot sachant sa classe, comme dans (Uchiumi et al., 2015). Ces deux étapes reposent également sur des processus Pitman-Yor distincts, afin d'avoir un nombre restreint de classes et de mots. Il reprend aussi une structure hiérarchique, avec notamment un modèle de caractères trigramme.

Trois de ces différents modèles sont comparés dans (Godard et al., 2016) puis (Godard, 2019) avec le modèle bigramme de `dpseg`, sur le corpus mboshi de BULB (Adda et al., 2016) entre autres. Malgré les améliorations sophistiquées qu'ils intègrent, les résultats sont soit sensiblement meilleurs que `dpseg` (pour (Löser et Allauzen, 2016)), soit dégradés (pour (Neubig, 2014) et une implémentation de (Mochihashi et al., 2009)). `dpseg` apparaît donc comme un modèle simple et stable pour la segmentation en mots de ce corpus. Il est intéressant de noter que tous les modèles sauf celui de (Löser et Allauzen, 2016) présentent une tendance à sur-segmenter et génèrent trop peu de types.

2.2.5 *Adaptor Grammar*

Le modèle des *Adaptor Grammars* (AG) (Johnson et al., 2007), amélioré par la suite dans (Johnson et Goldwater, 2009), est une extension des grammaires non contextuelles probabilistes (*Probabilistic Context-Free Grammars* en anglais ; PCFG). Le formalisme des PCFG est lui-même

une extension des grammaires non contextuelles (*Context-Free Grammars* en anglais ; CFG). Formellement, une grammaire non contextuelle G est définie par :

1. $\mathcal{T}$ pour l'ensemble fini des symboles terminaux ;
2. $\mathcal{N}$ pour l'ensemble fini des symboles *non* terminaux ;
3. $\mathcal{R}$ pour l'ensemble des règles de production de la forme $a \to b$, où $a \in \mathcal{N}$ et $b \in (\mathcal{T} \cup \mathcal{N})^*$, où E^* dénote la fermeture de Kleene de l'ensemble E ;
4. $\mathcal{S} \in \mathcal{N}$ pour le symbole de départ de la structure.

La version probabiliste introduit un ensemble supplémentaire de paramètres θ affectant à chaque règle de production de la grammaire une probabilité, de sorte que, pour chaque non-terminal a, la somme cumulée des règles de tête égale à a soit de 1. Le calcul de la probabilité d'un arbre se fait alors en évaluant récursivement tous les sous-arbres.

Les PCFG reposent toutefois sur une hypothèse d'indépendance, en particulier entre un non-terminal a et son nœud parent ou ses nœuds frères. Les AG ajoutent alors la possibilité d'*adapter* la distribution des arbres générés pour un symbole non terminal, en utilisant un mécanisme de cache, à la manière de la distinction entre générateurs et adapteurs vus en section 2.2.3.

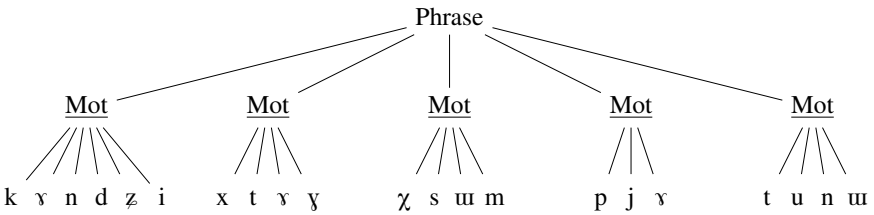


FIGURE 2.4 – Représentation simplifiée d'une segmentation (fausse) en mots obtenue avec l'*Adaptor Grammar* pour la phrase de l'exemple 2.1 : « kɤndʑixtɤɣ χsɯm pjɤtunɯ ».

En se restreignant à la tâche de segmentation de séquences, l'AG permet donc d'apprendre les différentes structures de formation des sous-unités, comme les mots ou les syllabes, avec une distribution reflétant une loi de puissance, en choisissant un adapteur approprié. Notons que récemment, ce modèle est plutôt utilisé pour la segmentation en morphèmes, en transposant les strates, comme nous le décrivons en section 2.2.8. La figure 2.4 présente une segmentation possible en mots, obtenue avec la grammaire suivante pour la phrase de l'exemple 2.1. Les terminaux correspondent aux caractères et dans cette représentation, seuls certains non terminaux sont adaptés (soulignés ci-dessous) :

$$\begin{aligned} \underline{\text{Mot}} &\to \text{Caractères} \\ \text{Caractères} &\to \text{Caractère} \\ \text{Caractères} &\to \text{Caractères Caractère} \end{aligned}$$

Il est alors possible d'exprimer, par exemple, le modèle de processus de Dirichlet pour la segmentation des mots sous forme d'AG. La représentation ci-dessus reprend l'illustration dédiée de (Johnson et al., 2007). Cette structure indique que l'entité Mot est composée d'entités Caractères ; ces dernières sont constituées soit d'un unique Caractère, soit d'une autre entité Caractères suivie d'un Caractère. Les deux dernières lignes peuvent être raccourcies par la notation : Caractères $\to$ Caractère$^+$, ce qui nous permet d'obtenir la représentation aplatie ci-dessus. En adaptant l'entité Mot (indiqué par le soulignement) avec un processus de Dirichlet, nous retrouvons donc le modèle de segmentation basé sur ce processus (en ajoutant la règle Phrase $\to$ Mot$^+$), illustré en

figure 2.4 : un Mot peut être généré à partir du cache constitué des sous-arbres précédents (de manière analogue aux clients déjà assis dans un restaurant) ou être généré à nouveau (comme un nouveau client). La probabilité pour chaque production « Mot → Caractères » est alors modifiée en exploitant ce cache, ce qui rend les mots déjà produits par le passé plus probables.

En pratique, Johnson (2008b) utilise l'AG avec un adapteur fondé sur les processus de Dirichlet, pour segmenter en mots un corpus (et analyser morphologiquement des verbes) en anglais. Différentes grammaires sont testées : en introduisant des règles sur la structure d'une syllabe (qui est constituée d'une attaque et d'une rime, elle-même composée d'un noyau et d'une coda), entre les Mots et les Caractères, ou en représentant le niveau des Locutions, un regroupement intermédiaire entre les Mots et les Phrases. Cette dernière grammaire rejoint l'idée du modèle bigramme de `dpseg` d'une dépendance entre les mots ; bien que l'implémentation diffère, elle parvient à obtenir des résultats proches de (Goldwater et al., 2006).

Johnson (2008a) effectue une analyse similaire, mais cette fois sur le sesotho (code ISO 639-3 : `sot`), une langue bantoue, à la morphologie plus complexe que l'anglais, afin d'évaluer la validité du modèle sur une langue aux règles et structures différentes de l'anglais. En effet, Johnson (2008a) étudie des configurations (grammaires) supplémentaires, de la simple grammaire unigramme à celles avec plusieurs niveaux de segmentation (mots, morphèmes et syllabes). Par ailleurs, des connaissances linguistiques du sesotho ont été intégrées dans la grammaire, notamment le patron structurel des mots en racine et affixes (trois préfixes et un suffixe au plus), améliorant particulièrement la qualité de segmentation.

D'autres modifications ont été également étudiées, pour l'anglais. Johnson et Goldwater (2009) observent entre autres les bénéfices d'une structure imbriquée de Locutions (trois niveaux de Locutions précisément), permettant de prendre en compte différents types de dépendances entre les mots. Johnson et al. (2014) étendent cette structure en représentant séparément dans la grammaire les mots outils, ce qui améliore aussi nettement la segmentation.

Dans le cadre de la documentation des langues, ce modèle peut donc, grâce à la collaboration avec des linguistes, intégrer des connaissances linguistiques, notamment la structure des syllabes (patrons Consonnes Voyelles) comme dans (Godard et al., 2018b). Les règles sont adaptées à la langue, à des degrés variables, pour les niveaux des locutions (mots consécutifs), des mots (groupements de morphèmes), des syllabes (suites de caractères) et des caractères. Les résultats indiquent une amélioration quand des règles spécifiques à la langue sont utilisées. De plus, une telle approche permet d'assister les linguistes à travers un point de vue complémentaire : par exemple, le modèle de segmentation obtient de meilleurs scores lorsque les consonnes complexes (constituées de deux ou trois caractères) sont explicitement représentées dans la grammaire, ce qui peut être un argument supplémentaire pour les intégrer comme consonnes à part entière de la phonologie. De même, l'AG est parvenu à apprendre les principaux affixes dans les langues étudiées, sans nécessiter de supervision.

Par ailleurs, mentionnons que l'*Adaptor Grammar* ainsi que `dpseg` sont proposés dans la boîte à outils Wordseg [16] (Bernard et al., 2020), visant à uniformiser l'utilisation en pratique de différents modèles de segmentation en mots non supervisée.

2.2.6 Méthodes neuronales

Comme elles nécessitent une quantité de données conséquente, les approches neuronales de segmentation en mots concernent davantage les langues plus dotées qui ne présentent pas de mar-

16. https://github.com/bootphon/wordseg.

queur visible de frontières de mots ; c'est le cas de certaines langues asiatiques comme le chinois (Sun et Deng, 2018; Wang et Zheng, 2022), le japonais (Kitagawa et Komachi, 2018) ou le thaï (Seeha et al., 2020), pour lesquelles nous donnons quelques exemples avec ou sans supervision. Notons au passage que ces langues ont également été étudiées avec des approches statistiques, comme (Magistry, 2012) ou (Magistry et Sagot, 2012), à travers la variation de l'entropie de branchement pour le chinois mandarin (Tanaka-Ishii, 2005; Jin et Tanaka-Ishii, 2006) ou pour le japonais et le chinois mandarin ((Mochihashi et al., 2009), présenté en section 2.2.4). Néanmoins, il existe aussi des modèles neuronaux pour la segmentation en mots de langues très peu dotées.

Aligner pour segmenter Dans le cadre de la documentation, les traductions dans une langue plus dotée sont également disponibles. Ce constat rend possible une approche combinant l'alignement et la segmentation : en effet, dans ces modèles, les caractères de la phrase dans la langue à documenter sont chacun alignés avec des mots de la traduction. Les segments seront alors reconstitués en fusionnant la succession des caractères alignés au même mot. De plus, cette approche présente l'avantage de générer des liens d'alignement entre les phrases des deux langues.

Si cette méthode existe pour des paires de langues mieux dotées (comme dans (Stahlberg et al., 2012)), c'est ainsi que procèdent Boito et al. (2017) sur le corpus mboshi-français collecté dans le cadre du projet BULB (Adda et al., 2016). En entraînant un modèle de traduction neuronal entre ces deux langues, la matrice d'attention calculée peut être utilisée afin d'en extraire des alignements entre les caractères mboshi et les mots français. Parmi les différentes configurations testées, l'architecture « inversée » (c'est-à-dire, traduisant du français vers le mboshi) en appliquant un lissage dans les alignements obtient les meilleurs scores. De plus, cette méthode permet d'obtenir une distribution des segments plus proche de la référence que `dpseg`. Cette architecture peut également être semi-supervisée ; ici, les cent mots les plus fréquents ont été donnés en entraînement, améliorant alors la qualité de segmentation.

Godard et al. (2019) étendent cette approche en introduisant deux modifications dans la fonction de perte, afin de contrôler la longueur des unités segmentées. La première méthode intègre un biais proportionnel à la longueur du mot à aligner dans l'attention, en partant du constat que les mots alignés dans les deux langues tendent à avoir des longueurs comparables. L'autre méthode modifie la fonction de perte afin d'obtenir des nombres d'unités comparables entre les deux phrases. Si la première dégrade les résultats par rapport à la version sans altération, la seconde obtient de meilleurs résultats, en régulant davantage le nombre de mots par phrases, ce qui réduit sa tendance à sur-segmenter. Malheureusement, ce modèle reste moins performant que l'AG (Godard et al., 2018b) sur le même jeu de données. Toutefois, à travers ses liens d'alignement, elle permet également d'obtenir un lexique bilingue dont la qualité n'est pas évaluée dans ce travail.

Utiliser un pré-entraînement multilingue Downey et al. (2022a) utilisent un modèle neuronal non supervisé de segmentation de séquences, un modèle de langue segmental masqué (*Masked Segmental Language Model* en anglais) (Downey et al., 2021). Il s'agit d'une variante du modèle de langue segmental (*Segmental Language Model* en anglais) (Sun et Deng, 2018; Kawakami et al., 2019), lui-même conçu pour la tâche de segmentation du chinois mandarin et fondé sur des LSTM. La nouveauté de l'approche masquée réside dans l'utilisation d'un transformeur à la place d'un LSTM en encodeur ainsi que d'un masque d'attention sur le segment courant. Cette méthode obtient de meilleures segmentations que sa version de base sur le chinois mandarin notamment.

En vue d'une extension aux langues peu dotées, Downey et al. (2022a) effectuent un pré-entraînement de ce modèle sur un corpus monolingue ou multilingue de dix langues d'Amérique latine (AmericasNLP 2021, Mager et al. (2021)), pour segmenter en mots et morphèmes (les frontières ne sont en revanche pas différenciées) des phrases non segmentées d'une langue appartenant à une autre famille linguistique, le k'iche' (ISO 639-3 : quc) (Tyers et Henderson, 2021). Ce faisant, des caractéristiques générales sur les langues semblent avoir été modélisées, aboutissant à de meilleurs résultats de segmentation, même à partir de quelques centaines de phrases de supervision. De plus, le corpus intégral de pré-entraînement reste certes plus petit en taille que ceux employés habituellement, mais les langues présentes sont plus proches de celle étudiée, sans pour autant appartenir à la même famille linguistique : elles sont morphologiquement riches, parlées dans des pays hispanophones ou lusophones et quelques-unes sont rapprochées par des liens historiques et géographiques. Cette approche apparaît alors comme un moyen d'utiliser un pré-entraînement multilingue pour des langues très peu dotées.

2.2.7 De l'enregistrement audio vers la transcription segmentée

Comme les systèmes standards de reconnaissance de la parole nécessitent un modèle de langue efficace (et donc une quantité de données adaptée pour l'entraînement) pour une transcription orthographique, ils ne sont pas envisageables, pour le moment, dans le cadre de la documentation automatique des langues. C'est pourquoi les méthodes employées s'appuient davantage sur des systèmes de transcription phonétique puis de segmentation ; les récentes avancées des modèles pré-entraînés permettent néanmoins de proposer une solution intégrée vers une transcription segmentée en mots.

Transcrire puis segmenter Une première approche, comme celle adoptée pour le projet BULB (Adda et al., 2016) (et CLD2025) ou utilisée lors de la piste 2 sur la découverte de termes parlés (en anglais, *spoken term discovery*) du *Zero Resource Speech Challenge* [17] (Dunbar et al., 2017), est d'aborder la tâche en cascade : par exemple, une transcription phonétique depuis l'enregistrement, puis une segmentation en mots. C'est ce que font notamment Godard et al. (2018a) sur le corpus mboshi, à travers le procédé en trois étapes décrit en section 2.2.1 (Franke et al., 2016; Müller et al., 2017b,a) pour obtenir une transcription automatique non segmentée, puis en utilisant `dpseg` pour segmenter cette séquence de phonèmes. L'écart significatif avec les performances du même modèle de segmentation sur une transcription de référence indique néanmoins la part de bruit du module de transcription, mais aussi le manque de robustesse d'une approche comme `dpseg`.

Dans la continuité des approches conjointes d'alignement et de segmentation décrites en section 2.2.6, Godard et al. (2018c) étendent l'approche à une segmentation à partir de données acoustiques : l'enregistrement de la phrase est tout d'abord converti grâce à un système de découverte d'unités acoustiques (*Acoustic Unit Discovery* en anglais), en l'occurrence basé sur (Ondel et al., 2016). Puis, à travers un modèle neuronal de traduction automatique, les mots de la traduction sont alignés avec les symboles phonétiques afin d'obtenir une phrase segmentée. Les modèles neuronaux déployés obtiennent aussi des performances moins dégradées que `dpseg` notamment, en comparant les segmentations obtenues à partir de la transcription de référence et celle générée automatiquement.

17. https://zerospeech.com/.

En utilisant la même méthodologie d'alignement entre symboles phonétiques et mots de la traduction (ou d'une transcription), Boito et al. (2019) étendent ce travail en comparant trois architectures neuronales de séquence à séquence pour la segmentation en mots à partir d'enregistrements : les réseaux de neurones récurrents (en anglais, *Recurrent Neural Networks* ou RNN), les réseaux de neurones convolutifs (en anglais, *Convolutional Neural Networks* ou CNN) et les transformeurs (Vaswani et al., 2017). Sur les mêmes données parallèles en mboshi et français, le RNN obtient des résultats meilleurs et plus robustes que les deux autres modèles.

De même, Boito et al. (2022) comparent cinq modèles de transcription (trois bayésiens et deux neuronaux), combinés avec deux modèles de segmentation en mots, à savoir `dpseg` et l'approche neuronale de (Godard et al., 2018c), pour le mboshi (et quatre autres langues). Les expériences montrent que les approches bayésiennes de transcription sont plus robustes dans ces scénarios, là où les modèles neuronaux génèrent des séquences trop longues, ce qui dégrade voire empêche l'utilisation des outils de segmentation. Concernant ces deux derniers, l'approche neuronale reste meilleure dans les situations bruitées, tandis que `dpseg` obtient des scores plus élevés sur les transcriptions de référence.

Approches intégrées pour les langues en cours de documentation Enfin, il existe également des approches neuronales abordant la transcription et la segmentation en mots de manière intégrée, ce qui présente notamment l'avantage d'accéder aux informations acoustiques (comme les pauses ou la prosodie) pour la segmentation. Grâce aux modèles pré-entraînés, la contrainte due à la taille des données peut être contournée pour obtenir un modèle de reconnaissance de la parole, même pour des langues très peu dotées en cours de documentation.

En effet, une première approche est de considérer un modèle de transcription non générique dépendant du locuteur. Cette restriction est pertinente dans le cadre de la documentation parce que les enregistrements sont principalement effectués auprès d'un nombre restreint de locuteurs (par exemple dans (Amith et al., 2021)). Gupta et Boulianne (2020b) considèrent de cette manière le cri (code ISO 639-3 : `cre`), une langue algonquienne. Une deuxième stratégie est de reposer sur les ressources et outils existants pour la langue ; Gupta et Boulianne (2020a) considèrent en effet l'utilisation d'un modèle de langue et d'un analyseur morphologique pour l'inuktikut (code ISO 639-3 : `iku`), une langue inuite polysynthétique (et étudient entre autres l'impact du niveau de segmentation, comme les syllabes et les morphèmes). Enfin, une autre méthode est de recourir à des modèles multilingues ; Gupta et Boulianne (2022) entraînent notamment un modèle sur des données de 12 langues peu dotées et de complexités morphologiques variées et obtiennent de meilleurs résultats que des modèles monolingues. Une amélioration des résultats a aussi été observée grâce à un affinage sur les quelques heures d'entraînement disponibles pour la langue à transcrire. Enfin, Boulianne (2022) constate que dans le cadre de la transcription des enregistrements d'un seul locuteur (donc une approche dépendant du locuteur), l'affinage d'un modèle pré-entraîné multilingue, XLSR-53 (Conneau et al., 2021), constitue la meilleure approche.

Guillaume et al. (2022) conçoivent également une méthode fondée sur un affinage du modèle pré-entraîné multilingue XLSR-53 (Conneau et al., 2021), qui propose une représentation du signal de manière universelle, indépendamment des langues. Puis, à partir de cette représentation, la seconde étape consiste en une reconnaissance de phonèmes, en reposant sur les données annotées dans la langue étudiée (ici, le japhug). Une des nouveautés de leur approche consiste en l'utilisation d'un marqueur spécifique pour les frontières de mots (« »), une pratique courante pour les langues plus dotées, mais non utilisée jusqu'alors dans le cadre de la documentation des langues. Les résultats obtenus sont alors prometteurs, malgré une sensibilité aux méta-paramètres

utilisés : à partir de deux heures de données transcrites, le taux d'erreur (calculé grâce à la distance d'édition entre la référence et la prédiction) au niveau des caractères est de 12,5 %.

De plus, un des points clefs pour évaluer les performances réelles des modèles dans une collaboration transdisciplinaire telle que les projets de documentation automatique des langues, reste l'impact pour les utilisateurs, ici les linguistes. Dans (Guillaume et al., 2022), un linguiste expert du japhug a vérifié les annotations prédites pour quelques phrases (correspondant à un enregistrement d'environ deux minutes) et n'a eu besoin de corriger que 1,9 % des caractères seulement. Les métriques automatiques semblent alors assez pessimistes sur la qualité des transcriptions, du moins pour ce qui concerne leur utilité en pratique.

Ces transcriptions, suffisamment précises pour ne nécessiter que quelques corrections, sont ainsi déjà utiles pour la documentation des langues. En effet, l'accès au signal sonore, et en particulier aux informations orales comme les pauses entre les unités ou la prosodie, constitue un atout majeur pour cette approche de transcription segmentée de bout en bout, rendant alors à terme caducs, les modèles de segmentation textuels (Guillaume et al., 2022).

2.2.8 La segmentation en morphèmes

Comparativement à la segmentation en mots, la segmentation en morphèmes est une tâche répandue en TAL, notamment à travers des défis partagés, comme le Morpho Challenge [18] (Kurimo et al., 2010) tenu presque chaque année entre 2005 et 2010 ou celui de SIGMORPHON (Batsuren et al., 2022a). Au-delà de l'intérêt linguistique, elle permet d'améliorer les performances de tâches en aval, comme la reconnaissance automatique de la parole (Afify et al., 2006) ou la traduction automatique (Clifton et Sarkar, 2011), en particulier pour les langues morphologiquement riches. Notons que cette tâche est initialement définie comme la segmentation des *mots* en morphèmes, plutôt que des phrases, ce qui signifie que la fréquence des mots n'est pas prise en compte et chaque type est d'importance équivalente.

Plusieurs approches ont été proposées pour résoudre cette tâche, comme Morfessor (Creutz et Lagus, 2002, 2007), fondé sur le principe de longueur de description minimale (Rissanen, 1989), présenté en section 2.2.2, des méthodes d'étiquetage de séquences avec les champs aléatoires conditionnels (ou, en anglais, *Conditional Random Fields* ; CRF) (Lafferty et al., 2001), présentés en annexe A.2, ou des modèles neuronaux. De plus, elle a été abordée à travers des approches supervisées mais également non supervisées ; par exemple, Ruokolainen et al. (2016) présentent une revue de littérature sur les cas faiblement et non supervisés. Nous nous focalisons ici sur les travaux concernant soit les méthodes bayésiennes non paramétriques, notamment l'*Adaptor Grammar* (présentée en section 2.2.5), soit les langues peu dotées, dans des situations sans supervision ou faiblement supervisées.

Segmentation canonique ou de surface Nous pouvons distinguer deux manières de segmenter en morphèmes : la segmentation de surface qui identifie des frontières dans la forme de mots telle qu'elle est (comme « étudiais » segmenté en « étudi-ais ») ainsi que la segmentation canonique qui recouvre les unités définies de manière « canonique », comme des lemmes (étudier-ais, pour le même exemple) (Cotterell et al., 2016). La segmentation canonique nécessite principalement une approche supervisée car elle implique d'identifier à la fois les unités canoniques et les segmentations, ce qui rend le problème sous-spécifié pour des approches entièrement non supervisées ; un problème qui ne se pose pas pour la segmentation de surface.

18. http://morpho.aalto.fi/events/morphochallenge/.

Par exemple, Moeng et al. (2021) abordent la segmentation canonique et de surface pour les langues nguni, sous-groupe des langues bantoues, qui sont peu dotées. Pour la segmentation canonique, des méthodes neuronales sont employées comme un LSTM bidirectionnel avec un mécanisme d'attention (Bahdanau et al., 2015) ou un transformeur (Vaswani et al., 2017). Bien que leurs performances soient supérieures à des approches à base de règles notamment, la taille des données d'entraînement semble être le facteur limitant par rapport aux performances sur d'autres langues plus dotées. Pour la segmentation de surface, Moeng et al. (2021) utilisent un CRF ou un modèle combiné de LSTM bidirectionnel et CRF, où la tâche est considérée comme un étiquetage de caractères (avec des étiquettes comme début, milieu ou fin de morphèmes). Dans cette expérience, le modèle CRF simple obtient de meilleurs scores ; ce résultat peut ici aussi être expliqué par la taille des données d'entraînement.

Notons que pour ces langues nguni, Meyer et Buys (2022) développent un modèle de langue qui apprend également à segmenter en morphèmes. Cet apprentissage conjoint permet d'optimiser la segmentation en sous-unités ; ce modèle obtient alors de meilleurs résultats, comparativement aux modèles standards non supervisés de segmentation en sous-unités comme Morfessor, mais aussi comme BPE (Sennrich et al., 2016).

Notons que la segmentation de surface est plus proche de nos tâches, dans la mesure où nous souhaitons segmenter les mots tels qu'ils sont énoncés puis retranscrits (et non tels qu'ils sont analysés).

Segmentation canonique bayésienne Mentionnons tout d'abord une approche bayésienne de segmentation canonique. Naradowski et Goldwater (2009) utilisent la version de `dpseg` pour la segmentation en morphèmes de (Goldwater et al., 2005), où les suffixes et les racines des mots sont considérés comme des unités explicitement distinctes. En prenant en compte, dans le modèle, le contexte à la jonction des morphèmes, donc ici les caractères autour de la frontière entre la racine et le suffixe éventuel, il est possible de représenter les transformations locales, comme l'insertion ou la suppression de caractères. Par exemple, en anglais, la segmentation « state-ing » pour le mot « stating » peut être obtenue avec l'insertion d'un caractère à la frontière, « e », dans le contexte « at-i ». Selon ces expériences, la modélisation de ces transformations permet de mieux prédire les segmentations en morphèmes pour les verbes en anglais.

Segmentation de surface avec une *Adaptor Grammar* Pour la segmentation de surface, Sirts et Goldwater (2013) utilisent notamment l'*Adaptor Grammar* (Johnson et al. (2007) ; voir section 2.2.5) de manière faiblement supervisée afin de mettre à profit une liste de mille mots segmentés en morphèmes. De fait, les segmentations de référence sont utilisées pour contraindre le modèle, en supprimant les segmentations incohérentes au niveau des morphèmes. Une autre stratégie employée est de créer une grammaire binaire générique et de sélectionner la grammaire la plus appropriée pour la langue et les données, en s'appuyant sur le jeu de segmentations de référence, ce qui permet d'éviter une spécification manuelle de la grammaire. Les expériences sur cinq langues morphologiquement variées conduisent à une amélioration significative grâce à l'approche semi-supervisée, par rapport au cas sans supervision.

Dans la continuité, Eskander et al. (2016) expérimentent l'*Adaptor Grammar* avec différentes grammaires de base, en distinguant notamment les préfixes et les suffixes des racines, ainsi qu'une approche faiblement supervisée, cette fois avec une liste d'affixes de la langue. En effet, comme l'AG nécessite de préciser manuellement la grammaire, il est possible d'y indiquer explicitement les affixes de la langue. Ceux-ci peuvent être obtenus soit à travers des descriptions grammaticales,

soit automatiquement en utilisant un AG en amont pour identifier les affixes de la langue. Ce type de supervision faible aboutit à de meilleures performances non seulement par rapport à une situation sans supervision, mais aussi par rapport à Morfessor (Creutz et Lagus, 2007) ou l'approche de Sirts et Goldwater (2013).

Eskander et al. (2019) étendent cette approche non supervisée, pour quatre langues polysynthétiques, dans lesquelles les mots peuvent être composés de nombreux morphèmes, ce qui constitue donc une difficulté supplémentaire. Comme le corpus d'entraînement est de petite taille (500 phrases environ), deux voies supplémentaires d'amélioration ont été étudiées, au-delà de l'utilisation d'une liste d'affixes : un entraînement multilingue en combinant les mots d'entraînement de quatre langues de la même famille et une augmentation des données dans la même langue mais avec des mots provenant d'une source différente. Une étude plus exhaustive sur douze langues morphologiquement variées et disposant de données de supervision de tailles différentes est effectuée dans (Eskander et al., 2020). Le modèle utilisé, MorphAGram [19], y est également publié et permet une configuration non supervisée et faiblement supervisée.

En utilisant la même méthodologie, Eskander et al. (2021) incorporent cette fois des informations spécifiques aux langues grâce aux linguistes, à travers soit des grammaires spécialisées, soit des listes d'affixes. Il s'agit alors d'une forme de supervision faible qui ne nécessite pas de mots segmentés au préalable, mais plutôt des connaissances linguistiques. Leurs expériences sur le japonais et le géorgien montrent une amélioration notable par rapport à la version non supervisée de MorphAGram ainsi qu'à Morfessor. En prévision d'une extension pour des langues peu dotées, la quantité de données pour ces deux langues a été restreinte. Dans ce contexte, Khandagale et al. (2022) appliquent cette approche en s'intéressant à deux langues polysynthétiques. En utilisant une liste d'affixes, la segmentation est ici aussi améliorée pour l'une des langues ; quant à l'autre, les informations de supervision fournies ne semblent pas suffisamment détaillées. Par ailleurs, MorphAGram a également été employé pour la segmentation en morphèmes non supervisée de langues en danger, comme dans (Bear et Cook, 2022) pour le wolastoqey (aussi appelé malécite-passamaquoddy ; code ISO 639-3 : `pqm`), une langue algonquienne parlée au Canada et aux États-Unis.

Segmentation de surface avec un CRF Des approches reposant sur des modèles statistiques tels que les champs aléatoires conditionnels, abordent la tâche comme un étiquetage de caractères : ceux-ci peuvent être en tête, en milieu ou en fin de morphème [20]. Ruokolainen et al. (2013) explorent notamment une approche entièrement supervisée par peu de données ; un travail qui est étendu dans (Ruokolainen et al., 2014), en exploitant les données non annotées par une méthode semi-supervisée. L'idée est de combiner les forces des modèles non supervisés, pour traiter l'entièreté des données non étiquetées, avec les modèles supervisés, entraînés sur des données étiquetées de bien plus petite taille (ici, un écart d'ordre de grandeur d'au moins 100). Les auteurs constatent un effet bénéfique de la supervision faible, avec une amélioration pour les trois langues étudiées par rapport aux modèles précédents (entièrement ou partiellement supervisés).

Segmentation de surface neuronale Au-delà de cet étiquetage des caractères d'un mot, la segmentation en morphèmes peut également être effectuée avec une approche de séquence à séquence. Kann et al. (2018) utilisent une approche neuronale pour quatre langues polysynthétiques

19. `https://github.com/rnd2110/MorphAGram`.
20. D'autres configurations d'étiquetage peuvent être utilisées pour représenter les frontières, en distinguant en plus, par exemple, les morphèmes constitués d'un seul caractère.

de la famille des langues uto-aztèques parlées au Mexique. Le modèle de segmentation est fondé sur une architecture neuronale de séquence à séquence (Bahdanau et al., 2015) en encodant chaque caractère des mots. Comme il n'y a que 500 mots environ pour l'entraînement du modèle, deux stratégies sont déployées afin de mettre à profit les données non annotées dans une langue : l'introduction d'une tâche auxiliaire de copie (c'est-à-dire, recopier l'entrée en sortie) et l'augmentation de données au moyen d'entrées et sorties identiques. Dans ces deux cas, les données supplémentaires sont soit des mots non segmentés de la langue, soit des chaînes de caractères aléatoires. Kann et al. (2018) comparent donc ce système avec différents modèles de base de segmentation morphologique : le même modèle neuronal sans les modifications, deux extensions de Morfessor, la version semi-supervisée de (Kohonen et al., 2010) et FlatCat (Grönroos et al., 2014) et un CRF. Les expériences montrent que le modèle neuronal, même sans utiliser les données non étiquetées, obtient des résultats comparables au CRF, meilleur modèle de base, malgré la quantité de données et la morphologie complexe des langues.

Liu et al. (2021) appliquent ce modèle neuronal de segmentation morphologique, sans amélioration, principalement pour le seneca (see), une langue iroquoienne des États-Unis et du Canada, classifiée par l'*Ethnologue* (Eberhard et al., 2023) comme étant en danger, ainsi que pour les quatre langues étudiées par (Kann et al., 2018). Trois configurations d'entraînement ont été évaluées afin de bénéficier des trois sources de données à disposition : un livre sur les verbes, des transcriptions fournies par les locuteurs ainsi qu'une traduction de la Bible. La première configuration consiste en un entraînement standard sur les données du même corpus, la deuxième sur un changement de domaine (par exemple, utiliser la Bible comme ressource additionnelle pour la segmentation des verbes) en fournissant les données supplémentaires lors de l'entraînement, et la troisième sur un transfert mutlilingue où les données des quatre langues du Mexique sont utilisées simultanément. Par ailleurs, pour le développement, comme différentes sources sont disponibles, le modèle utilise soit le jeu de données dédié du même corpus, soit celui d'une autre source et donc d'un autre domaine ; cette seconde possibilité permet d'entraîner le modèle sur plus de mots du domaine initial, en combinant les données d'entraînement et de développement.

2.3 La génération automatique de gloses

La seconde tâche traitée dans cette thèse est la génération automatique de gloses. Ces dernières sont des annotations linguistiques, fondamentales pour la constitution de corpus de documentation mais aussi pour l'analyse de la langue. Dans cette section, nous définissions tout d'abord cette tâche, moins commune que la segmentation en mots ou en morphèmes, ainsi que ses enjeux, puis décrivons les diverses méthodes utilisées jusque là pour répondre aux besoins des linguistes. Du fait de la nature même des gloses, l'automatisation ne permet cependant qu'un pré-traitement destiné à être révisé manuellement ensuite.

2.3.1 De l'intérêt des gloses

Nous reposons principalement sur la définition de Lehmann (1982) pour les gloses : il s'agit d'informations linguistiques explicatives des morphèmes d'une langue étudiée source en utilisant soit des unités issues de la traduction dans la langue cible de documentation, soit un jeu d'étiquettes et de symboles.

Historiquement, les annotations qui pourraient aussi être qualifiées de gloses, à savoir des commentaires linguistiques permettant la compréhension du texte, existent depuis plusieurs siècles.

L'un des travaux les plus notables, sans être le plus ancien, en terme de couverture mais aussi de qualité est (Dorsey, 1890), décrivant l'omaha (code ISO 639-3 : oma). En dessous de chaque mot de cette langue nord-américaine, est disposée une courte traduction en anglais, allant souvent d'un simple mot (« *rabbit* ») à une locution plus longue (« *went homeward, they say* »).

Une autre façon de gloser est d'indiquer pour chaque morphème une position dans la traduction, comme dans l'exemple 2.5 suivant. Cette disposition est certes concise mais dépend fortement

$$\begin{array}{lll} \text{āmo}_1 & \text{ti}_2\text{-xolochtōn}_3 & \text{ti}_2\text{-ye-z}_4 \\ \text{Tu}_2 & \text{ne}_1\ \text{seras}_4\ \text{pas}_1\ \text{hypocrite}_3 & \end{array}$$

FIGURE 2.5 – Exemple de gloses « indices » en nahuatl classique, extrait de (Launey, 1994).

de la traduction pour pouvoir être réalisable ; si cette dernière est trop éloignée de la phrase source, l'indexation devient difficile.

Le format *Interlinear Glossed Text* (IGT) Aujourd'hui, malgré quelques limites notoires, le format de présentation le plus usité, aussi bien en linguistique que dans le domaine du TAL, est celui nommé en anglais *Interlinear Glossed Text* (IGT), comme décrit par (Lehmann, 1982) pour les gloses interlinéaires morphémiques (en anglais, *Interlinear Morphemic Gloss*, (IMG)). Il s'agit de la structure correspondant aux strates **S3** à **S5** de la figure 2.1 : la phrase dans la langue source segmentée en première ligne, les gloses associées alignées en deuxième ligne ainsi que la traduction dans une langue plus dotée en troisième ligne. Ces trois types d'informations sont en effet systématiquement présentes, agrémentées parfois de strates supplémentaires comme les étiquettes en parties du discours (ou, en anglais, *Part-of-Speech*, PoS), par exemple. Les IGT permettent en particulier d'expliciter le rôle de chaque morphème (par exemple, la version IGT de l'exemple 2.5 spécifie que « z » exprime le futur, tandis que « ye » signifie être).

Parmi les différents types existants, nous nous intéressons donc aux gloses interlinéaires ; celles-ci disposent de conventions pour essayer d'en uniformiser les annotations, la plus répandue étant les *Leipzig Glossing Rules*[21] (Bickel et al., 2008), que nous considérons exclusivement dans cette thèse. Elles définissent non seulement un jeu d'étiquettes de gloses grammaticales fréquemment observées dans les langues, modifiées selon les besoins, mais cadrent également les manières de représenter certains phénomènes (par exemple, la réduplication ou les clitiques). Il faut néanmoins mentionner que d'autres représentations existent ; récemment, les *Generalized Glossing Guidelines* (GGG) (Mortensen et al., 2023) ont notamment essayé de trouver une solution pour les langues à morphologie non-concaténative, où les phénomènes linguistiques sont plus difficilement analysables sous la forme d'opérations de concaténation de segments.

L'importance des gloses L'utilisation des gloses permet tout d'abord de rendre plus accessible des analyses de phénomènes linguistiques dans les phrases de la langue étudiée. C'est ainsi que les grammaires illustrent les phénomènes morphosyntaxiques, comme en figure 2.6. En effet, même sans connaître la langue, avec un matériel linguistique approprié, les lecteurs peuvent comprendre grâce à la ligne intermédiaire que « χsɯm » signifie trois en japhug et que le suffixe « nɯ » marque ici le pluriel.

Comme évoqué en section 2.1.1, les gloses peuvent être réparties en deux catégories : celles qui indiquent le rôle grammatical et dont l'inventaire est fini pour une langue donnée et celles qui

21. B. Fradin propose un inventaire en français à l'adresse : `http://www.llf.cnrs.fr/fr/node/60`.

(149) kɤndʑi-xtɤɣ χsɯm pjɤ-tu-nɯ
COLL-brother three IFR.IPFV-exist-PL
'There were three brothers.' (31-deluge)

FIGURE 2.6 – Phrase de l'exemple 2.1, extrait de la grammaire du japhug de Jacques (2021). Nous gardons ici les conventions typographiques originales.

sont lexicales qui expriment la signification du morphème. Elles sont respectivement représentées dans l'exemple 2.6 en majuscules et en minuscules. Si les secondes sont aisément compréhensibles, les premières explicitent le rôle du morphème à travers des conventions comme les *Leipzig Glossing Rules*. Elles peuvent être composées, comme « 1SG.POSS », constituée de « 1 » pour première personne, « SG » pour singulier et « POSS » pour le possessif. Notons également qu'une composition des deux types de gloses est possible, comme « know.1SG » pour le verbe conjugué « sais ».

Grâce à la double annotation morphosyntaxique et lexicale, les gloses sont donc centrales dans le processus de documentation, de constitution de corpus et de présentation des analyses. Toutefois, ces annotations sont très coûteuses à produire, car il s'agit principalement d'une analyse manuelle, peu automatisée, et nécessitant surtout une expertise linguistique ainsi qu'une compréhension suffisante du fonctionnement de la langue étudiée. C'est pourquoi il est courant d'observer un écart significatif entre la quantité de données collectées sur le terrain et celles qui sont effectivement annotées ; les corpus IGT sont de fait de taille plus petite que les textes parallèles source-cible (Seifart et al., 2018; Zhao et al., 2020).

Si l'annotation de gloses est pour le moment une tâche principalement manuelle pour le linguiste, certains outils d'annotations comme ELAN-CorpA (CNRS-LLACAN, 2023) ou *FieldWorks Language Explorer* (FLEx) (Rogers, 2010) permettent néanmoins de construire un dictionnaire en sauvegardant les associations entre morphèmes sources et gloses observées, afin de les proposer pour les prochaines occurrences. Ce type de suggestion est directement bénéfique pour les morphèmes qui ont toujours la même étiquette par exemple, mais dès lors que se présente une ambiguïté, la prise en compte du contexte devient nécessaire. De plus, ce système ne peut naturellement pas prédire l'étiquette d'une unité jamais observée auparavant.

L'idée de la documentation automatique des langues est alors de permettre de faciliter ce processus par une systématisation partielle. Elle permettrait également d'éviter les incohérences de gloses dues au caractère répétitif de l'annotation.

Applications en TAL Dans le domaine du TAL, les gloses constituent une ressource linguistique principalement utilisée pour l'amélioration de tâches subséquentes, car elles permettent d'établir un pont entre les langues source et cible. L'idée est souvent de transposer les modèles disponibles pour les langues plus dotées (cible) vers celles qui le sont moins (source). Au-delà d'un rôle d'intermédiaire, elles contiennent également de riches informations linguistiques, car elles assurent la compréhension d'une phrase dans une langue avec laquelle le lecteur n'est pas nécessairement familier.

Tout d'abord, les gloses établissent des liens fiables entre les mots de la langue source et ceux de la traduction : les morphèmes sources suivent une correspondance bijective avec chaque glose (**S3** & **S4**) et parmi ces dernières, celles lexicales se retrouvent souvent dans la phrase traduite (Georgi, 2016), s'apparentant alors à une tâche d'alignement dans la même langue (**S4** & **S5**). En se basant sur ce constat, puisque les langues de documentation comptent parmi les mieux re-

présentées en terme de ressources et outils (comme l'anglais par exemple), Xia et Lewis (2007) projettent des structures syntaxiques (comme l'analyse en dépendance) de la langue cible vers la langue source, comme illustré en figure 2.7. Si le procédé de projection en lui-même était déjà courant (Hwa et al., 2002), il n'était pas adapté aux langues peu dotées qui ne bénéficiaient pas d'alignements de bonne qualité, du fait de leur petite quantité de données parallèles. Les données glosées permettent de remédier à ce problème ; en expérimentant pour sept langues variées, provenant du corpus glosé multilingue ODIN (Lewis, 2006; Lewis et Xia, 2010), Xia et Lewis (2007) obtiennent des résultats prometteurs et soulignent qu'il s'agit d'une première étape pour développer des outils de TAL qui nécessiteraient sinon davantage de données. Georgi et al. (2012) présentent une application directe de cette méthode en extrayant des caractéristiques depuis ces projections afin d'améliorer les performances des analyseurs en dépendance lors de la constitution de treebanks.

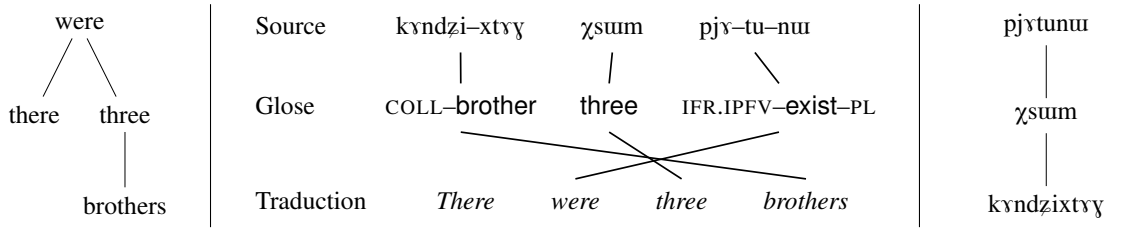


FIGURE 2.7 – Exemple de projection d'analyse en dépendance de l'anglais (gauche) vers le japhug (droite) en alignant à l'aide des gloses (milieu) pour la phrase de l'exemple 2.1.

À travers ce type de projection de structure, il est également possible de répondre à des questions linguistiques ; Lewis et Xia (2008) s'intéressent par exemple à l'ordre des mots et d'autres paramètres typologiques (Greenberg, 1963) pour dix langues. Ils se réfèrent aux règles de grammaire induites (ici des règles de grammaires non contextuelles) et leurs fréquences associées, obtenues à travers les phrases IGT, et mesurent leur correspondance avec les données du *World Atlas of Language Structures* (WALS) de 2005 (Dryer et Haspelmath, 2013). En complément, Lewis et Xia (2008) étudient un peu moins de cent langues avec la même méthodologie mais pour l'ordre des mots uniquement ; au-delà de 40 phrases glosées, pour limiter les effets des cas particuliers, les projections parviennent à une prédiction correcte à 99 %.

Dans cette optique, le projet AGGREGATION [22] de l'université de Washington aborde la systématisation de cette approche pour constituer des grammaires à partir des données IGT pour les langues en cours de documentation. Bender et al. (2013) s'intéressent principalement aux questions de l'ordre des mots et des déclinaisons pour plusieurs langues, là où Bender et al. (2014) puis Zamaraeva et al. (2019) emploient la même méthode pour extraire des règles de grammaire pour le chintang (code ISO 639-3 : ctn), bien documenté. À travers les données glosées, il est également possible de créer des modèles d'analyse morphologique ; Zamaraeva et al. (2017) les utilisent pour prédire la classe des verbes à partir des patrons préfixe-racine obtenus.

En utilisant un procédé similaire, Georgi et al. (2015) conçoivent un modèle d'étiquetage en parties du discours. Une première approche est naturellement d'utiliser les gloses pour transférer les informations en anglais vers les mots sources. Cependant, tous les mots en cible ne sont pas nécessairement alignés et les divergences linguistiques (Dorr, 1994) peuvent causer des erreurs

22. https://depts.washington.edu/uwcl/aggregation/.

si l'on applique l'étiquette en anglais directement à la langue étudiée[23]. Il est alors possible de s'appuyer sur l'autre force des gloses, comme information linguistique, afin de pallier ces problèmes. Elles sont cruciales pour désambigüiser la prédiction finale lorsque plusieurs étiquettes sont envisageables notamment. En comparaison avec une prédiction établie uniquement à partir de la projection, l'approche intégrant les gloses comme ressource complémentaire permet d'améliorer les performances de l'étiqueteur. Cette expérience constitue encore une fois une première étape vers l'extension à d'autres langues très peu dotées, car elle rend accessible un outil sans nécessiter de données massivement parallèles. Ce travail est étendu par le système INTENT[24] (Xia et al., 2015) qui permet d'enrichir les données de la langue source avec les étiquettes PoS, des alignements de mots et des analyses en dépendance.

Enfin, une dernière tâche bénéficiant des gloses est la traduction automatique où elles peuvent également intervenir comme pivot. En effet, les annotations linguistiques contiennent non seulement la décomposition en morphèmes, d'une grande aide pour les langues morphologiquement riches, mais aussi des informations grammaticales. L'approche de Zhou et al. (2019) exploite ces liens entre les langues source et cible à travers une succession d'étapes. La phrase source est en premier lieu analysée morphologiquement et normalisée pour obtenir des gloses, à savoir des lemmes (dans la langue source) et des étiquettes grammaticales suivant les *Leipzig Glossing Rules*. Ensuite, ils s'appuient sur un dictionnaire (soit déjà existant, soit obtenu par alignement de données parallèles conséquentes, comme ici) pour traduire les lemmes. Enfin, un modèle neuronal de traduction automatique multilingue, entraîné sur des corpus glosés de plusieurs dizaines de milliers de phrases comme ODIN, permet d'obtenir à partir de la glose, la traduction en langue cible. Bien que cette approche nécessite un analyseur morphologique (première étape) et d'un dictionnaire fourni (deuxième étape) de la langue, leur modèle en soi n'a été entraîné que sur 865 phrases sources ; là encore, les gloses permettent de mettre à la portée des langues peu dotées, des méthodes qui nécessiteraient sinon une bien plus grande quantité de données.

Ces approches reposent principalement sur la bonne qualité des alignements entre les gloses et la traduction. Ainsi, même pour des langues très peu dotées, où la quantité de données parallèles source-cible ne suffit pas pour entraîner un modèle d'alignement robuste, les données glosées constituent une ressource alternative convaincante pour obtenir les liens entre les mots.

2.3.2 La tâche de génération automatique de gloses

L'objectif est de prédire automatiquement les gloses (strate **S4** dans la figure 2.1) à partir de la phrase source (strates **S2** ou **S3**), et éventuellement de la phrase cible (strate **S5**). En effet, malgré une finalité partagée, les données utilisées en entrée peuvent varier sensiblement. Concernant la phrase de la langue étudiée, si la segmentation en mots est généralement supposée, celle en morphèmes ne l'est pas nécessairement. De fait, elle demande une décomposition supplémentaire : soit la langue dispose d'un analyseur morphologique robuste, soit il s'agit à nouveau d'une annotation manuelle. Ce dernier cas est plus commun dans le cadre de la documentation des langues.

Formellement, en notant $\boldsymbol{x}$ la phrase source et $\boldsymbol{y}$ les gloses interlinéaires associées, la génération de glose a été jusque là modélisée, soit en calculant $p(\boldsymbol{y}|\boldsymbol{x})$, si seule la source est utilisée, soit, avec $\boldsymbol{z}$ la traduction, $p(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{z})$ à l'aide des deux entrées. Les éléments de $\boldsymbol{x}$ et $\boldsymbol{y}$ correspondent

23. Un exemple simple est un concept correspondant à une unité dans une langue (« des ») et à deux avec des étiquettes différentes dans une autre (« of the », en anglais).

24. https://github.com/rgeorgi/intent.

position par position ($\boldsymbol{x} = x_{[1:T]}$ et $\boldsymbol{y} = y_{[1:T]}$, pour T mots ou morphèmes), là où la traduction ne vérifie aucune contrainte de longueur $\boldsymbol{z} = z_{[1:Z]}$.

Dans cette thèse, nous nous plaçons dans la configuration où les phrases sources sont entièrement segmentées en morphèmes et les traductions associées sont disponibles. Les travaux antérieurs se sont parfois intéressés à d'autres situations, comme nous le décrivons dans la section 2.3.3. Néanmoins, dans notre cas, l'accès à la paire de phrases source segmentée et cible est une hypothèse courante pour la tâche, considérée notamment par l'état de l'art (McMillan-Major, 2020; Zhao et al., 2020). De plus, comme évoqué en section 2.1.1, les corpus parallèles représentent habituellement la ressource minimale pour la documentation.

Défi partagé SIGMORPHON 2023 sur la génération automatique de gloses Il faut noter que la tâche de génération automatique de gloses a bénéficié récemment d'un intérêt particulier à travers le défi partagé SIGMORPHON 2023 qui lui est dédié (Ginn et al., 2023) ; il s'agit de la première initiative de ce type. Sept langues géographiquement et morphologiquement variées ont été étudiées, représentant notamment des stades différents dans l'avancement de la documentation. En effet, cela se traduit par la taille des données d'entraînement, allant de très peu de phrases (31), en passant par des langues moyennement dotées (environ 700), jusqu'à celles relativement bien dotées (environ 140 000 phrases). Nous les décrivons plus en détails en section 3.2.4. En pratique, deux pistes étaient proposées, pour couvrir deux cas réalistes vis-à-vis des ressources à disposition, en s'inspirant de la configuration de (Zhao et al., 2020).

La piste fermée (ou piste 1) correspond à une situation où la phrase source est uniquement segmentée en mots ; sans la segmentation en morphèmes, il est donc nécessaire de prédire indirectement le nombre de morphèmes pour chaque mot et éventuellement sa segmentation. Par ailleurs, aucune ressource externe ne pouvait être utilisée.

La piste ouverte (ou piste 2) est un cadre plus favorable, comme il donne accès aux phrases segmentées (canoniquement) en morphèmes ; la correspondance entre chaque morphème et les gloses est alors assurée. En outre, une des langues était également annotée en PoS et des ressources externes pouvaient être utilisées (hormis des données glosées additionnelles dans la langue étudiée).

Pour ces deux pistes, la traduction dans une langue plus dotée (l'anglais ou l'espagnol) était donnée, sauf pour une. La tâche telle que nous la définissons et traitons dans cette thèse correspond ainsi à la piste ouverte du défi partagé.

2.3.3 Méthodes de génération de gloses

S'il est possible de répartir les travaux de génération de gloses selon les modèles utilisés, statistiques comme les champs aléatoires conditionnels (ou, en anglais, *Conditional Random Fields* ; CRF) (Lafferty et al., 2001), présentés en section A.2, ou neuronaux, nous faisons ici le choix de les présenter selon les différents types d'approches, afin de mieux voir les similitudes entre les techniques utilisées et les défis rencontrés.

Étiqueter les gloses telles des parties du discours La première approche notable de génération automatique de gloses a été effectuée par Palmer et al. (2009), dont l'analyse est complétée par Baldridge et Palmer (2009). Cette tâche est considérée comme une variante de l'étiquetage, plus commun, en parties du discours (ou, en anglais, *Part-of-Speech*, PoS), avec la particularité d'avoir un ensemble d'étiquettes plus varié. En effet, la seule différence est que le modèle de classification prédit, pour chaque morphème, soit la glose grammaticale, soit l'étiquette PoS associée.

Toutefois, la question de recherche principale tient ici davantage à l'apport de l'apprentissage actif dans le cadre de la documentation des langues, qu'aux performances du modèle en soi.

Samardžić et al. (2015) utilisent le même principe et traitent la génération de gloses comme un étiquetage en PoS modifié, mais cherchent également à identifier les gloses lexicales. Leur approche se fait en deux temps : un étiqueteur, le modèle de classification de Shen et al. (2007) qui améliore le « perceptron structuré » de Collins et Roark (2004), effectue en premier lieu la prédiction des gloses grammaticales, tandis qu'il affecte aux morphèmes lexicaux la PoS correspondante, afin d'avoir un nombre fini d'étiquettes (environ 200). La seconde étape sera alors d'identifier ces gloses lexicales en se référant à un lexique ; les étiquettes en parties du discours permettent justement de pouvoir lever les ambiguïtés. En effet, toutes les étiquettes possibles pour le morphème considéré sont listées dans le dictionnaire ; le filtrage par le PoS réduit les possibilités à un seul choix pour environ 85 % des cas. Lorsque plusieurs candidats persistent, des heuristiques sont mises en place, comme l'évaluation de la probabilité de la séquence constituée avec les deux étiquettes précédemment prédites.

Si cette dernière méthode s'est avérée performante dans leur cas d'étude, elle nécessite néanmoins des ressources conséquentes : les données d'apprentissage, segmentées en morphèmes et étiquetées en PoS, sont constituées de 955 025 mots, ce qui équivaut à 214 heures d'enregistrement, et chaque morphème du corpus est associé à une entrée du lexique. Il s'agit, en ce sens, d'un cas peu commun en documentation des langues (bien qu'idéal pour le TAL).

Prédire la segmentation en morphèmes et les gloses L'approche de Moeller et Hulden (2018) effectue une prédiction jointe des frontières de morphèmes et des gloses dans un corpus de mots en lezghien. Comme les gloses lexicales constituent un ensemble non fini, elles sont toutes regroupées sous une seule étiquette, « stem », dans les trois modèles qui sont comparés. Le premier est un CRF prédisant simultanément des étiquettes BIO (début (B), interne (I) et hors du morphème (O)) assorti de la glose grammaticale ou de « stem » et repose sur des fonctions caractéristiques, pour la plupart, généralisables à d'autres langues. Le deuxième est une approche en cascade utilisant un CRF pour la segmentation en morphèmes, puis classifiant les morphèmes ainsi obtenus vers les étiquettes de gloses. Enfin, la troisième s'inspire des avancées dans la tâche connexe de prédiction de flexions morphologiques ; il s'agit d'une approche neuronale de séquence à séquence, basée sur un LSTM, prédisant les mêmes étiquettes BIO que le premier modèle. Avec seulement 3 000 mots, préalablement annotés en parties du discours, la méthode utilisant seulement un CRF parvient à prédire correctement plus de 80 % des frontières de morphèmes avec les annotations en gloses. Les principaux défis observés résident dans les allomorphes, c'est-à-dire les étiquettes correspondant à plusieurs morphèmes distincts (ici, uniquement des affixes) ; l'ambiguïté est souvent levée par le contexte et sa position dans le mot.

En utilisant la même méthodologie, Barriga Martínez et al. (2021) comparent pour la langue otomi (code ISO 639-3 : ots) différents modèles statistiques, comme les CRF ou les modèles de Markov cachés (en anglais, *Hidden Markov Models* ou HMM), et neuronaux, tels les réseaux de neurones récurrents (en anglais, *Recurrent Neural Networks* ou RNN) ou les réseaux récurrents à mémoire court et long terme bidirectionnels (bidirectional biLSTM). Les CRF obtiennent de meilleurs résultats par rapport aux modèles neuronaux, eux-mêmes moins bons que le simple HMM. La taille des données semble alors être le principal facteur limitant pour le déploiement des méthodes neuronales vers des langues très peu dotées (ici, un peu plus de 1 000 phrases). Au-delà de ce résultat cohérent avec les constats antérieurs, Barriga Martínez et al. (2021) notent que l'amélioration due à l'utilisation des étiquettes en parties du discours à l'entraînement ne semble

pas décisive, ce qui suggère que cette approche reste viable pour les corpus dépourvus de cette annotation.

Prédire les gloses lexicales La meilleure configuration statistique provient de (McMillan-Major, 2020), qui utilise notamment la traduction comme entrée supplémentaire, afin de permettre la prédiction de gloses lexicales. Tandis qu'un premier CRF traite la phrase dans la langue étudiée (modèle source) en étiquetant chacun des morphèmes, un second considère les mots de la phrase traduite (modèle cible) pour leur associer les gloses lexicales possiblement alignées. Les sorties de ces deux CRFs sont ensuite combinées de manière heuristique, qui choisit notamment en priorité les prédictions étayées par les données, en cas d'incohérence. Ce modèle présente l'avantage de pouvoir traiter également les mots et morphèmes jamais observés à l'entraînement, une limite des approches antérieures, en reposant sur l'hypothèse que les gloses lexicales sont contenues (directement ou indirectement) dans la traduction. Par ailleurs, cette méthodologie ne repose pas sur des caractéristiques propres aux langues étudiées et semble se généraliser aussi bien à travers des langues bien dotées qu'en cours de documentation et morphologiquement variées.

Inspiré par les avancées des modèles de traduction automatique, il existe aussi des modèles neuronaux de génération automatique de gloses. L'idée, évoquée dans des travaux tels que (Samardžić et al., 2015) ou considérée par (Moeller et Hulden, 2018) et (Barriga Martínez et al., 2021), de « traduire » de la langue source vers la « langue » des gloses n'avait pas été retenue. En effet, ses résultats étaient médiocres comparativement aux modèles statistiques, qui souffrent moins du manque des données d'entraînement.

(Zhao et al., 2020) est à ce titre la première approche à surpasser les méthodes statistiques et repose de même sur une utilisation conjointe des phrases source et cible. Leur choix s'est porté sur le transformeur (Vaswani et al., 2017), modifié pour encoder de manière séparée les phrases source et cible, vues comme deux sources. Le décodeur utilise l'information combinée via les matrices d'attention. Néanmoins, l'emploi d'un modèle de séquence à séquence pose des défis spécifiques : la correspondance entre les morphèmes source et les gloses n'est plus garantie. Pour y remédier, une contrainte de longueur minimale de la phrase en sortie est introduite et semble suffisante.

Deux configurations sont évaluées : la phrase dans la langue étudiée est soit seulement segmentée au niveau de mots, un cas plus fréquent mais plus difficile, soit les frontières des morphèmes sont aussi connues, ce qui est plus simple mais nécessite une annotation manuelle ou un analyseur morphologique robuste. Notons ici que cette distinction est à l'origine de la création des deux pistes du défi partagé SIGMORPHON (Ginn et al., 2023) présenté dans la section précédente.

Dans ces deux situations, les méthodes neuronales employant une correction de longueur et, éventuellement, un transfert de connaissances si des données d'une autre langue de la même famille sont disponibles, obtiennent de meilleurs scores que la méthode de (McMillan-Major, 2020). La qualité des prédictions est systématiquement plus élevée quand la segmentation en morphèmes est fournie, ce qui souligne l'importance d'une telle ressource. Par ailleurs, l'apport de la traduction comme source secondaire est également important ; elle contribue à l'amélioration des performances par rapport à une approche ayant recours uniquement à la phrase source, en particulier pour les deux langues avec le plus de données.

Modèles neuronaux du défi partagé SIGMORPHON 2023 Dans le cadre du défi partagé SIGMORPHON sur la génération automatique de gloses (Ginn et al., 2023), plusieurs systèmes neuronaux ont été présentés, que nous décrivons ci-dessous. Le modèle de base (Ginn, 2023) fourni par les organisateurs repose sur le modèle RoBERTa (Liu et al., 2019) avec ses paramètres

par défaut et est utilisé pour étiqueter chaque mot (piste fermée) ou morphème (piste ouverte). Remarquons que ce modèle n'emploie pas la traduction et s'inscrit dans la continuité des méthodes statistiques d'étiquetage de la phrase source. En effet, une approche de séquence à séquence n'a pas été retenue, car une simple insertion ou omission d'unité décale toutes les étiquettes suivantes et dégrade les scores selon la procédure d'évaluation définie. S'agissant d'un modèle de base, plusieurs pistes d'améliorations ont été suggérées ; certaines d'entre elles ont été explorées par les équipes participantes, comme l'utilisation de ressources auxiliaires ou de données augmentées.

Les meilleurs résultats sur toutes les langues de la piste fermée et sur quatre parmi les sept langues de la piste ouverte ont été obtenus par Girrbach (2023) avec deux approches. La première est un modèle de base, recourant au classifieur temporel connexioniste (ou en anglais, *Connectionist Temporal Classification*, CTC), introduit par Graves et al. (2006) pour étiqueter des séquences non segmentées avec un RNN. La seconde, le modèle principal, utilise une estimation du gradient *straight-through* (Bengio et al., 2013), dans le but d'obtenir des représentations intermédiaires interprétables au cours de la tâche. Notons que cette particularité permet aussi d'aborder la tâche d'inflection morphologique, autre défi partagé SIGMORPHON de 2023 (Goldman et al., 2023). Pour la génération de gloses, dans la piste fermée, le modèle neuronal aboutit en particulier à une représentation segmentée en morphèmes, qu'il utilise pour étiqueter la séquence à travers un perceptron multicouche. Dans la piste ouverte, la phrase source est remplacée par sa segmentation de référence. Notons enfin que ces deux approches n'ont besoin que des phrases sources (c'est-à-dire que les traductions sont ignorées) et permettent néanmoins d'atteindre de meilleurs scores que les autres systèmes qui y ont recours.

He et al. (2023) reposent sur deux méthodes principalement : d'une part, une amélioration du modèle de base (Ginn, 2023) en utilisant notamment les poids pré-entraînés de XLM-RoBERTa base (Conneau et al., 2020) et d'autre part, un modèle séquence à séquence pré-entraîné multilingue, ByT5 (Xue et al., 2022) ne nécessitant pas de tokenisation au préalable (car opérant au niveau des caractères), affiné sur les données d'entraînement. Ces deux modèles peuvent, par ailleurs, être aidés par une augmentation des données, ce qui mène à quatre configurations expérimentales possibles. Leur meilleur résultat provient du modèle d'étiquetage de tokens (XLM-R) avec une augmentation de données. Ces dernières sont artificiellement générées en extrayant des patrons de mots contenant un morphème lexical, puis en remplaçant ce dernier par d'autres morphèmes lexicaux. Ce procédé permet notamment de conserver les structures internes des mots dans une phrase, au détriment de son sens.

Coates (2023) se concentre uniquement sur la piste fermée et utilise une architecture LSTM encodeur décodeur. Pour contourner les problèmes rencontrés par les méthodes de séquence à séquence comme (Ginn, 2023), la prédiction des gloses se fait au niveau des mots. Deux modèles reçoivent en entrée des segments de la phrase source, l'une avec une fenêtre d'un seul mot, l'autre de deux mots, pour conserver le contexte proche. La décision finale est prise en combinant leurs prédictions.

Cross et al. (2023) ont également recours à une technique d'augmentation des données, pour une utilisation par des modèles neuronaux. Elle s'appuie sur une fenêtre glissante de tailles variées répétant les mots pour la piste fermée. Pour la piste ouverte, les mots de chaque phrase sont isolés et répétés autant de fois qu'ils contiennent des morphèmes, ce qui permet de représenter successivement ces derniers avec une balise dédiée. Remarquons que du fait de ce fractionnement, le modèle n'a pas accès au contexte du mot au sein de la phrase. La piste fermée est traitée avec un transformeur classique (Vaswani et al., 2017) avec une tokenisation en caractères ou en sous-mots (BPE, (Sennrich et al., 2016)). Pour la seconde, l'approche est divisée en deux temps, à la manière

des méthodes statistiques comme Samardžić et al. (2015) ou Moeller et Hulden (2018). D'abord, pour la prédiction des gloses grammaticales et d'une étiquette unique « stem », un modèle de classification neuronal à propagation avant est utilisé avec un encodeur de caractères BiLSTM ou ByT5 (Xue et al., 2022), plus performant. Puis, pour les gloses lexicales, ils reposent sur un dictionnaire associant à chaque morphème, son étiquette la plus fréquente. La tokenisation en caractères s'est révélée ici meilleure que celle en sous-mots, dans le cadre de la piste fermée, en particulier lorsque les données ne sont pas de grande taille. Quant à la piste ouverte, du fait des coûts de calcul, l'approche reposant sur l'architecture plus simple des BiLSTM est plus performante. Notons que cette participation présente aussi la spécificité de ne pas utiliser la traduction, en faisant le choix de laisser les gloses lexicales inconnues étiquetées comme telles, « UNK ».

Ainsi, nous observons que pour la piste ouverte en particulier, ces méthodes neuronales récentes abordent la tâche plutôt comme un étiquetage d'unités dans la continuité des travaux statistiques antérieurs, surtout du fait des défis rencontrés par les approches de séquence à séquence. De plus, pour satisfaire les besoins des modèles à l'entraînement, l'augmentation des données a été la technique privilégiée.

Pour résumer cette section, le tableau 2.1 récapitule les différents travaux de génération automatique de gloses selon leurs caractéristiques.

modèle	traduction	intégrée	neuronal	lexicales	seq2seq	piste fermée
(Palmer et al., 2009)	✗	✗	✗	✗	✗	✗
(Samardžić et al., 2015)	✗	✗	✗	✓	✗	✗
(Moeller et Hulden, 2018)	✗	✗		✗		✓
(Barriga Martínez et al., 2021)	✗	✗		✗	✗	✓
(McMillan-Major, 2020)	✓	✓	✗	✓	✗	✗
(Zhao et al., 2020)	✓	✓	✓	✓	✓	✓
(Ginn, 2023)	✓	✓	✓	✓	✗	✓
(Girrbach, 2023)	✗	✓	✓	✓	✗	✓
(He et al., 2023)	✓	✓	✓	✓		✗
(Coates, 2023)	✗	✓	✓	✓	✓	✓
(Cross et al., 2023)	✗		✓	✓		✓

TABLE 2.1 – Récapitulatif des particularités des différents travaux de génération automatique de gloses. Nous indiquons ici : l'utilisation d'une *traduction* comme entrée supplémentaire, d'une approche *intégrée* et non en cascade, d'un modèle *neuronal* et non statistique, la prédiction des gloses *lexicales* (avec ou sans dictionnaire), à travers une méthode de séquence à séquence (seq2seq) et si le modèle permet une participation éventuellement à la *piste fermée* du défi partagé, c'est-à-dire utiliser en entrée une phrase source non segmentée en morphèmes. Lorsque plusieurs modèles sont comparés, nous reportons en principe le meilleur d'entre eux, si une segmentation en morphèmes est disponible. Quand une des caractéristiques ne correspond à *aucun* des modèles présentés, nous l'indiquons par une croix (✗) ; la case reste vide sinon.

2.3.4 Limites de l'automatisation

Il faut souligner que les gloses ne sont pas des étiquettes figées, en particulier, dans les langues en cours de documentation : elles sont le produit d'un travail continu. En ce sens, il n'est pas rare de voir les linguistes revenir sur les annotations passées, comme dans l'expérience de Baldridge et Palmer (2009). En effet, dans cette expérience où deux linguistes, l'un expert de la langue et

l'autre non, annotent progressivement des phrases, les étiquettes du second concordaient davantage aux gloses de référence. La raison principale derrière ce constat, surprenant au premier abord, est justement l'évolution de l'analyse de la langue entre les moments où le corpus a été glosé et où l'expérience a eu lieu. L'automatisation ne peut donc être que partielle, afin de faciliter l'annotation, notamment pour les parties répétitives. Elle permettrait ici de réduire la charge manuelle et de limiter par la même occasion les erreurs d'inattention ou d'incohérence qui sont susceptibles de survenir.

En outre, les conventions laissent un certain degré de liberté dans les annotations et sont respectées de manière variable. Par exemple, au sein même des *Leipzig Glossing Rules* (Bickel et al., 2008), il existe des règles optionnelles quant à l'utilisation de certains marqueurs, appliquées selon les préférences de chacun. En effet, en reprenant leur exemple pour la règle 4, le mot français « aux » serait communément étiqueté en anglais comme « to.ART.PL »[25], mais selon la règle optionnelle 4B, il peut aussi être noté « to;ART;PL », pour indiquer que le mot est constitué en soi de plusieurs éléments mais qu'il ne peut pas être segmenté.

Rappelons également que les gloses restent des annotations spécifiques à une langue et à un linguiste dans une moindre mesure ; l'ensemble des étiquettes est de ce fait très dépendant d'un corpus (Palmer et al., 2009). Ces types de variations rendent donc une automatisation systématique d'une base de données à une autre plus difficile, ce qui est flagrant dans les corpus multilingues de gloses tels que ODIN (Lewis, 2006; Lewis et Xia, 2010).

Par ailleurs, les annotation de gloses pour une phrase sont aussi variables en fonction du contexte. Si le but est d'illustrer un phénomène grammatical particulier, les gloses peuvent être détaillées de manière approfondie, là où dans une autre situation, les indications peuvent être plus rudimentaires (Bickel et al., 2008). Dans la version en ligne d'un autre corpus multilingue de gloses, IMTVault[26] (Nordhoff et Krämer, 2022), par exemple, les 14 occurrences du mot « était » en français sont annotées « be.3SG.IPFV »[27], mais aussi « was(IMP) » ou simplement « was ». Notons au passage que IMP signifie impératif selon les *Leipzig Glossing Rules*.

Ainsi, la glose est une annotation qui reflète un travail en cours sur une langue, souvent liée au linguiste et au corpus, à des fins essentiellement explicatives. Cette variabilité inhérente fait que la génération automatique de gloses ne peut se faire qu'en partie et la vérification ultérieure par un linguiste est indispensable. Néanmoins, l'automatisation permet un gain de temps conséquent (Palmer et al., 2009) et une analyse plus systématique et cohérente, moins sujette aux erreurs des annotations manuelles, dues à la nature parfois très répétitive de la tâche et à la nature humaine.

2.4 Conclusion

Nous avons tout d'abord brièvement décrit les différentes initiatives d'intégration de méthodes de TAL dans les étapes de la documentation ainsi que dans la préservation et revitalisation des langues, dans une moindre mesure.

Puis, l'objectif principal de ce chapitre était de situer les deux tâches de segmentation de séquences et de génération de gloses dans le cadre de la documentation automatique des langues, en exposant leurs enjeux ainsi que leurs défis.

La segmentation en mots permet d'identifier les frontières de mots à partir d'une chaîne de symboles non segmentée. L'approche la plus adaptée dans le cadre de la documentation repose

25. ART pour article et PL pour pluriel.
26. Version 1.0.0 ; https://imtvault.org.
27. 3SG pour troisième personne du singulier et IPFV pour imperfectif.

sur les modèles bayésiens non paramétriques non supervisés comme `dpseg` (Goldwater et al., 2009). Notons à ce sujet que nous avons en particulier présenté les travaux s'appuyant sur des transcriptions de référence, sans bruit, ce qui n'est pas nécessairement le cas dans des conditions réelles, lors d'une transcription phonétique à partir d'un enregistrement. Nous avons également abordé la segmentation en morphèmes, plus courante en TAL, qui peut être effectuée de manière canonique ou en surface ; nous nous intéressons aussi à cette dernière dans nos travaux.

Ensuite, l'autre volet de cette thèse consiste à générer automatiquement des annotations linguistiques, morphosyntaxiques et lexicales, les gloses. Celles-ci sont fondamentales dans l'analyse linguistique d'une langue, mais sont coûteuses à produire. En effet, elles sont obtenues en grande partie manuellement pour le moment, comme elles nécessitent une expertise dans la langue à documenter notamment. La génération automatique des gloses en ce sens vise à accélérer également cette étape, afin de proposer un pré-traitement, qui serait par la suite corrigé. Elle permettrait donc de réduire et faciliter le travail d'annotation. Dans le domaine du TAL, il s'agit d'une tâche moins répandue et a entre autres été abordée à travers des méthodes statistiques comme les CRF. Cependant, elle a bénéficié d'une attention plus soutenue avec le défi partagé SIGMORPHON 2023 sur la génération de gloses, où des modèles neuronaux ont été principalement utilisés dans les participations, avec également un aspect multilingue. Enfin, il faut souligner que les gloses sont des analyses linguistiques et qu'elles sont susceptibles d'évoluer au cours de la documentation, comme le notent Baldridge et Palmer (2009).

Nous aborderons dans les chapitres à venir ces deux tâches dans le cadre d'un projet de documentation automatique des langues, étape par étape. Nous supposons la transcription phonétique (non segmentée) déjà effectuée et commençons par la segmentation en mots au chapitre 4. Puis, au chapitre 5, nous explorons la segmentation conjointe en mots et morphèmes, pour essayer de se rapprocher de ces unités de travail pour les linguistes. Enfin, en théorie, comme nous disposons d'une segmentation à deux niveaux, nous pouvons mener la génération automatique des gloses au chapitre 6.

Chapitre 3

Les ressources linguistiques pour la documentation automatique des langues

> Flétrie et sèche, cette fleur
> gardait toujours sa douce odeur.
>
> Acte II, *Carmen*, Bizet

3.1 Les données et corpus

Cette section détaille les différentes sources de données et corpus qui sont utilisées dans les projets de documentation des langues. Nous nous concentrons en particulier sur les ressources les plus adaptées aux tâches de nos travaux, en privilégiant la présence de frontières de mots, et éventuellement de morphèmes, ainsi que des annotations en gloses. Par ailleurs, les ressources ne sont pas présentées dans un ordre chronologique cohérent vis-à-vis de la documentation des langues. En effet, les linguistes commencent par un travail sur le terrain, dont les données, collectées puis annotées, sont éventuellement déposées sur des plateformes (voir section 3.1.3). Puis, les analyses linguistiques peuvent aboutir à une production d'articles ou de livres, comme des grammaires (voir section 3.1.2). Cette section suit néanmoins une autre structure, qui suggère la diffusion et la notoriété des différentes ressources dans le domaine du TAL, pour les tâches que nous traitons.

Rappelons ici que les langues documentées sont principalement orales et sont de fait peu, voire pas du tout présentes sur Internet (c'est le cas des langues amérindiennes ; Mager et al. (2018)) ; les corpus pour ces langues proviennent donc d'autres types de sources.

3.1.1 Projets de documentation des langues

La première source de données que nous présentons correspond naturellement aux projets de documentation *automatique* des langues. À travers les collaborations entre linguistes de terrain et informaticiens, les ressources sont souvent rendues plus accessibles pour les deux parties ; pour le TAL, elles sont donc idéales d'un point de vue pratique avec, entre autres, un accès ouvert, libre, un format défini ou un versionnage.

Nous pouvons illustrer ce cas avec le projet prédécesseur de CLD2025, BULB (Adda et al., 2016), présenté en section 2.1.2. Trois langues bantoues ont été documentées dans ce cadre, le mboshi (C25 selon Guthrie (1948) ; code ISO 639-3 : `mdw`), le myene (B10 ; `mye`) et le basaá (A43 ; `bas`), aboutissant notamment à la création de corpus dans les trois langues (Godard et al., 2018a; Hamlaoui et al., 2018). Des enregistrements de diverses natures ont été effectués sur le terrain

puis retranscrits sous forme textuelle. En outre, pour le mboshi et en basaá, des alignements forcés entre le signal audio et la transcription ont également été calculés.

Les portails Internet des projets de documentation sont également une autre porte d'entrée pour le TAL. Nous pouvons par exemple citer les corpus utilisés par Anastasopoulos (2019) lors de la conception d'outils pour la documentation de langues en danger : l'ainu (ain) du *Glossed Audio Corpus of Ainu Folklore* [1] (Nakagawa et al., 2016-2021), l'arapho (arp) de l'*Arapaho Language Project* [2] ainsi que le griko (pas de code ISO) (Lekakou et al., 2013) [3]. Au-delà des enregistrements, ces corpus contiennent également des annotations linguistiques comme des gloses ou des étiquettes PoS.

Plus généralement, la documentation des langues aboutit à des corpus annotés, qui sont de plus en plus accessibles en ligne sur des plateformes connus en TAL, comme Zenodo ou Hugging Face ; c'est notamment le cas de deux langues sino-tibétaine, le na (nru) et le japhug (jya) (Galliot et al., 2021). L'objectif est de faciliter la prise en main de ces données par la communauté TAL.

3.1.2 Descriptions grammaticales

Une deuxième approche consiste en l'utilisation des grammaires décrivant les langues étudiées. En effet, si l'annotation en gloses de textes est tout particulièrement plus rare du fait de son coût (voir section 2.3.1), les descriptions grammaticales tendent à présenter des phrases d'exemple dans le format IGT présenté à la figure 2.6 ; par la même occasion, nous pouvons alors avoir accès à une segmentation en mots et en morphèmes. Nous pouvons également remarquer que ces exemples sont souvent numérotées (« (149) ») et que des commentaires peuvent être présents, principalement dans la ligne de traduction (comme « (31-deluge) » ici).

ODIN En se reposant sur ces propriétés, il est possible de constituer un corpus de données IGT en extrayant les exemples des grammaires de différentes langues, disponibles en ligne : c'est ce qui a été fait pour ODIN [4] (Lewis, 2006; Lewis et Xia, 2010). Après avoir collecté des documents linguistiques en effectuant des requêtes Internet adaptées, puis effectué leur conversion du format PDF vers un texte à l'aide d'outils de reconnaissance optique de caractères (ou en anglais, *Optical Character Recognition*, OCR), les exemples linguistiques ont été extraits avec un modèle d'étiquetage de séquence. Un peu moins de 200 000 phrases ont été alors obtenues à partir de 2 868 documents. Ce processus a été complété par une vérification manuelle en corrigeant les frontières mal identifiées ainsi que la langue assignée automatiquement. Après cette étape, le corpus comporte 157 114 phrases glosées dont les trois quarts sont au format IGT tel que nous l'avons décrit en section 2.3.1, le reste étant incomplet (mais conservé).

Ensuite, après avoir nettoyé la base de données des erreurs dues à l'OCR dans les PDF notamment, ODIN est enrichi dans (Xia et al., 2014) en utilisant les alignements entre les gloses et les traductions en suivant le procédé de Xia et Lewis (2007) (voir section 2.3.1). La version la plus récente (v2.1) couvre alors 1 496 langues pour 158 007 phrases.

Au-delà de ces annotations automatiques, dans le cadre de (Xia et Lewis, 2007), une sous-partie de ces données pour sept langues (le corpus XL-IGT) a été manuellement corrigée pour l'alignement et l'analyse en dépendance.

1. https://ainu.ninjal.ac.jp/folklore/en/.
2. https://verbs.colorado.edu/ArapahoLanguageProject/.
3. http://griko.project.uoi.gr/index.php.
4. *Online Database of INterlinear text*, accessible à https://depts.washington.edu/uwcl/odin/.

Notons enfin les initiatives des dernières années visant à améliorer les méthodes de reconnaissance optique de caractères pour les langues très peu dotées. En effet, les outils standards sont peu performants voire impossibles à entraîner à cause du manque de données. Rijhwani et al. (2020) parvient à pallier ce problème en utilisant une post-correction intégrant également la traduction, souvent disponible dans les documents traités. Cette approche est étendue à travers une méthode semi-supervisée, en se référant à un lexique constitué à partir des unités transcrites (Rijhwani et al., 2021). Grâce à ce type de modèles, la méthodologie de ODIN pourrait aboutir à une base de données de gloses moins bruitée.

Cette approche rencontre néanmoins deux problèmes : les livres publiés sont fréquemment soumis à des droits d'auteurs et la conversion à partir d'un document PDF ajoute nécessairement, pour le moment, une part de bruit.

***Language Science Press* et IMTVault** L'exemple 2.6 du chapitre précédent est obtenu, en réalité, à partir du code source LaTeX présenté en figure 3.1.

```
\begin{exe}
\ex \label{ex:XsWm.pjAtunW}
\gll krndʑi-xtʏɣ χsum pjɤ-tu-nɯ \\
\textsc{coll}-brother three \textsc{ifr}.\textsc{ipfv}-exist-\textsc{pl} \\
\glt `There were three brothers.' (31-deluge)     \japhdoi{0004077\#S7}
\end{exe}
```

FIGURE 3.1 – Code source LaTeX correspondant à l'exemple extrait de la grammaire en figure 2.6.

En effet, l'éditeur de la grammaire, *Language Science Press* (Nordhoff, 2018) publie les fichiers sources LaTeX sur GitHub[5] avec une licence CC-BY, ce qui nous laisse la possibilité de parcourir les fichiers et de retrouver les exemples dans les grammaires. De fait, nous observons une structure spécifique pour les exemples avec un environnement `exe` et des balises pour chaque strate d'annotation IGT : la phrase source segmentée en morphèmes (balise `\gll`), sa version glosée (balise `\glo`, parfois absente mais la glose succède nécessairement à la phrase source) ainsi que sa traduction (balise `\glt`). Ainsi, en parcourant ces fichiers, nous avons accès à des phrases segmentées, glosées et traduites avec un bruit minimal. Il faut tout de même pré-traiter les commandes propres à LaTeX (« `\textsc` ») et les éléments externes, existants dans tout exemple de livres grammaticaux (« `(31-deluge)` »). Appliquer ce procédé sur les livres grammaticaux de *Language Science Press* permet alors de contourner les deux problèmes rencontrés avec ODIN : les livres sont accessibles librement et aucun procédé d'OCR n'est requis.

En utilisant une approche massive de cette méthode, il est aussi possible de constituer un corpus multilingue de phrases IGT (tel ODIN) : il s'agit d'IMTVault (Nordhoff et Krämer, 2022). Il est obtenu en explorant les codes sources LaTeX disponibles de tous les livres de linguistique publiés par l'éditeur *Language Science Press*. En effet, comme les exemples sont marqués par des balises spécifiques, communs à tous les ouvrages, cela permet une approche systématique pour extraire les phrases au format IGT. Après quelques pré-traitements pour enlever les commandes LaTeX et pour assurer la cohérence entre les unités sources et glosées, le corpus contient alors environ 40 000 phrases pour 280 langues. En accédant directement aux phrases des livres de grammaire, la quantité de bruit dans les données est largement diminuée par rapport à ODIN,

5. https://github.com/langsci.

bien que la quantité de phrases soit moindre. Notons que la base de données peut également être consultée de manière interactive en ligne (voir la note de bas de page 26).

Cette approche peut être étendue à d'autres sources où les phrases glosées sont disponibles directement et dans un format unifié, comme le journal de linguistique Glossa [6].

3.1.3 Autres sources

Bien que nous n'y avons pas recours dans le cadre de cette thèse, d'autres sources peuvent être employées pour le traitement des langues très peu dotées et en cours de documentation. Une première, difficile à catégoriser ici, repose sur les connaissances de la langue, comme sa structure morphologique, déjà utilisée pour la segmentation notamment (Godard et al., 2018b; Eskander et al., 2021).

Les archives de langues Les corpus audio et textuels obtenus lors des projets de documentation peuvent aussi être déposés sur certaines plateformes d'archivage ; si ces dernières sont connues dans le domaine de la linguistique, ce n'est pas encore le cas en TAL, comme le soulignent Zariquiey et al. (2022), et elles restent sous-utilisées. Un premier exemple est la collection Pangloss [7] (Michaud et al., 2016), une archive de langues à tradition orale en ligne, développée par le LACITO. Elle comporte, en (avril) 2023, 5 420 enregistrements pour 237 langues, dont un peu moins de la moitié sont transcrits et incluent des annotations telles que des segmentations en morphèmes ou des gloses. L'ensemble est libre d'accès, avec la possibilité de télécharger les fichiers audio (ou vidéos) et les éventuelles données auxiliaires. Six langues disposent également d'un dictionnaire.

L'*Atlas of Pidgin and Creole Language Structures* [8] (APiCS ; Michaelis et al. (2013)) se concentre quant à lui sur 76 pidgins et langues créoles dans le monde et présente entre autres 18 526 phrases glosées et traduites, possiblement assorties d'enregistrements.

En outre, certaines archives se focalisent sur des aires géographiques comme l'*Archive of the Indigenous Languages of Latin America* [9] (AILLA) pour les langues autochtones d'Amérique latine ou la *Pacific and Regional Archive for Digital Sources in Endangered Cultures* [10] (PARADISEC) pour celles de la région Pacifique. Elles contiennent des enregistrements avec éventuellement des transcriptions ; pour la seconde par exemple, plus de 1 300 langues sont représentées avec plus de 15 000 heures d'enregistrement au total.

Enfin, il existe des archives spécialisées dans les langues en danger comme l'*Endangered Languages Archive* [11] (ELAR) ou *The Language Archive* [12] (TLA) du Max Planck Institute for Psycholinguistics, réunissant les corpus audio et textuels obtenus par la documentation.

Ces quelques corpus sont notamment regroupés dans le *Digital Endangered Languages and Musics Archives Network* [13] (DELAMAN), qui recense davantage d'archives de documentation. Il s'agit ainsi d'un point d'entrée pour s'intéresser aux données de langues en danger, comme l'entreprend Nordhoff (2020), afin de les rendre plus accessibles pour la communauté TAL. De

6. https://www.glossa-journal.org/.
7. https://pangloss.cnrs.fr/.
8. https://apics-online.info/.
9. https://ailla.utexas.org/.
10. https://www.paradisec.org.au/.
11. https://www.elararchive.org/.
12. https://archive.mpi.nl/tla/.
13. https://www.delaman.org/.

même, le projet DoReCo, présenté en section 2.1.2, repose également sur ces ressources (Paschen et al., 2020).

Dictionnaires et lexiques Les dictionnaires sont une autre ressource émanant de la documentation des langues. Nous pouvons ici mentionner le projet *Automated Similarity Judgment Program* [14] ou ASJP (Wichmann et al., 2022) qui recense la même liste de 40 mots dans plus de 5 500 langues. Il est vrai que dans le cadre de cette thèse, nous utilisons des listes de mots, afin de simuler cette ressource. Il faut toutefois souligner que dans un cadre réel, les dictionnaires contiennent bien plus d'informations avec davantage d'unités, des traductions dans une ou plusieurs langues, des définitions et également des phrases d'exemple.

Les défis partagés Une autre catégorie de ressources qui peut être utilisée pour les langues très peu dotées repose sur les défis partagés spécifiques. Bien que ceux-ci proviennent souvent de corpus recueillis par les linguistes, ils ont été cependant pré-traités en vue de tâches de TAL. La visibilité ainsi que l'aspect pratique qui en résulte en font des ressources privilégiées. En effet, certaines étapes de la documentation peuvent faire directement l'objet d'un défi comme la segmentation en morphèmes, dans le cadre du groupe de travail SIGMORPHON [15] (*Special Interest Group on Computational Morphology and Phonology*) ou, plus spécifiquement, comme la génération automatique des gloses cette année (voir section 2.3.2).

De plus, nous pouvons mentionner les tâches connexes comme la flexion morphologique, où l'objectif est de prédire la flexion souhaitée à partir du lemme (par exemple, la forme conjuguée à partir d'un verbe à l'infinitif en français), vis-à-vis de la segmentation en morphèmes. Par exemple, lors de l'édition 2022 (Kodner et al., 2022), 33 langues très variées de 11 familles linguistiques ont été étudiées, dont des langues en danger comme le ket (`ket`). Les corpus comportent alors différentes formes par lemme, ce qui pourrait être employé pour entraîner un modèle de segmentation. Cette tâche est en relation étroite avec le corpus Unimorph [16] (la version 4.0 Batsuren et al. (2022b) est la plus récente), inventoriant des flexions morphologiques pour 169 langues à l'heure actuelle.

Par ailleurs, certains défis sont spécifiques à certaines langues : celui d'AmericasNLP [17] de 2023, par exemple, s'intéresse certes à la traduction automatique, mais fournit un corpus bilingue pour une dizaine de langues amérindiennes peu dotées (Ebrahimi et al., 2023). Ce type de corpus a déjà été utilisé pour pré-entraîner des modèles multilingues ; par rapport à la version (2021) utilisée par (Downey et al., 2022b) notamment, une nouvelle langue a été ajoutée.

Enfin, nous pouvons mentionner deux autres sources complémentaires. Tout d'abord, *Universal Dependencies* [18] est un ensemble de treebanks pour plus de 100 langues de différentes familles linguistiques. Les phrases y sont analysées en dépendance mais contiennent également, de ce fait, des phrases segmentées et parfois même glosées.

Ensuite, l'approche utilisée pour ODIN, à savoir la conversion de fichiers PDF accessibles en ligne, peut être étendue à d'autres types de documents. Bustamante et al. (2020) utilisent notamment les ressources pédagogiques de quatre langues autochtones du Pérou pour créer un corpus, chacune ayant environ une dizaine de milliers de phrases.

14. `https://asjp.clld.org/`.
15. `https://sigmorphon.github.io/`.
16. `https://unimorph.github.io/`.
17. `https://turing.iimas.unam.mx/americasnlp/`.
18. `https://universaldependencies.org/`.

3.2 Les langues et corpus étudiés dans cette thèse

Dans cette section, nous décrivons les langues étudiées dans cette thèse, en les catégorisant en fonction de leurs corpus et de leurs sources. Nous accompagnons chacune d'elles par leur code selon la norme ISO 639-3 [19], comme nous l'avons fait jusque là, et par des informations de la dernière édition à ce jour de l'*Ethnologue* [20] (Eberhard et al., 2023). La figure 3.2 présente une carte du monde afin de permettre au lecteur de situer ces langues.

FIGURE 3.2 – Carte des langues étudiées dans cette thèse, créée avec Khartis en utilisant les données d'*Ethnologue*.

Il s'agit donc non seulement de langues très variées géographiquement mais aussi linguistiquement ; nous comptons sept familles de langues : afro-asiatique (1), austronésienne (1), bantoue (1), maya (1), nakho-daghestanienne (2), sino-tibétaine (1) et tsimshianique (1). L'*Ethnologue* classifie trois de ces langues comme étant en danger (tsez, gitksan et mboshi) et cinq comme stables (japhug, lezghien, natugu, zaar et uspanteko). Concernant leur présence numérique, cinq sont qualifiées de « dormantes » (« *still* » en anglais ; tsez, mboshi, natugu, zaar et uspanteko), à savoir l'échelon le plus bas, et trois sont émergentes (« *emerging* » en anglais ; gitksan, japhug et lezghien), ce qui indique que quelques ressources et outils existent.

3.2.1 Le mboshi du projet BULB

Dans le cadre du projet BULB (Adda et al., 2016), présenté en section 2.1.2, trois langues bantoues ont été étudiées dont le mboshi (`mdw`). Il s'agit d'une langue parlée dans la région de la Cuvette, où dix variétés ont été recensées, dans la République du Congo ainsi que dans la capitale du pays et par la diaspora. Sa grammaire a déjà été étudiée (Amboulou, 1998 ; Embanga Aborobongui, 2013) et elle dispose également de dictionnaires en français (Beapami et al., 2000) et en anglais (Ndongo Ibara, 2014).

19. Gérée par SIL International ; `https://iso639-3.sil.org/code_tables/639/data`.
20. `https://www.ethnologue.com/`.

Le corpus (oral) recueilli [21] (Godard et al., 2018a) en entier est constitué de cinq sources : des phrases lues, des débats, des conjugaisons de 50 verbes, des extraits lus de la Bible et des commentaires sur 1 500 images. Les détails de chaque composante sont décrits dans (Rialland et al., 2018). Dans le contexte de cette thèse, nous nous intéressons uniquement à la première catégorie, qui représente environ 5 heures d'enregistrement, correspondant à un peu plus de 5 000 phrases transcrites. Celle-ci est elle-même divisée en 3 706 phrases provenant du dictionnaire de la langue (Beapami et al., 2000) et 1 472 des phrases traduites de (Bouquiaux et Thomas, 1976), un corpus utilisé pour la documentation de langues à tradition orale.

Les enregistrements sont alignés avec leur traduction en français et transcrits par les linguistes. À ce sujet, bien que le mboshi ne dispose pas d'un système d'écriture canonique, des linguistes ont défini une convention de transcription proche de sa phonétique. Elle comporte 7 voyelles et 25 consonnes, dont quelques unes sont pré-nasalisées, ce qui est une propriété courante pour les langues bantoues (Embanga Aborobongui, 2013; Kouarata, 2014).

De plus, c'est une langue possédant deux tons, signalés au moyen de diacritiques dans la transcription. Cependant, nous ne considérerons pas cette information et utilisons un pré-traitement les supprimant. De fait, selon (Godard et al., 2018d), les tons n'ont pas permis d'améliorer la qualité des segmentations du modèle de (Löser et Allauzen, 2016) au vu de la quantité de données présentes ; en mboshi, ils jouent un rôle aussi bien grammatical que lexical.

Comme nous étudions cette langue principalement pour la tâche de la segmentation, il faut mentionner que ses caractéristiques phonologiques rendent difficile l'identification des frontières, à travers notamment des phénomènes d'élision. En effet, la voyelle peut disparaître devant une autre à la jonction de deux mots, un phénomène commun aux langues bantoues (Rialland et al., 2015).

Notons que le mboshi a fait l'objet d'une étude détaillée précédemment pour la tâche de segmentation en mots à partir des données textuelles (Godard et al., 2016, 2018b, 2019), mais aussi à partir des enregistrements, en utilisant par exemple un système non supervisé de transcription de mots (Godard et al., 2018a,c; Anastasopoulos et Chiang, 2018; Anastasopoulos, 2019; Boito et al., 2022).

3.2.2 Le japhug, à partir de sa grammaire

Le japhug (inclus dans le code `jya` pour les langues gyalrong) est une langue sino-tibétaine de la branche qianguique, parlée dans la province du Sichuan au sud-ouest de la république populaire de Chine. Il possède 8 voyelles et 50 consonnes ; ces dernières peuvent être combinées pour générer plus de 400 groupes consonantiques (en anglais, *consonant cluster*). Cette particularité fait du japhug une des langues avec le plus riche inventaire de groupes consonantiques dans la famille des langues sino-tibétaines.

De plus, sa morphologie est complexe, pour les noms et surtout les verbes, qui peuvent présenter jusqu'à six ou sept préfixes successifs pour exprimer le temps, l'aspect ou le mode. Les suffixes servent quant à eux principalement pour la flexion et leur nombre est limité à quatre au plus. Ces procédés, malgré leur régularité, entraînent une augmentation de la quantité des formes pour un même lemme.

La transcription a principalement été effectuée en se basant sur l'alphabet phonétique international (ou en anglais, *International Phonetic Alphabet*, IPA), mais Jacques (2021) souligne l'utili-

21. https://www.islrn.org/resources/747-055-093-447-8/ et https://github.com/besacier/mboshi-french-parallel-corpus.

sation par les locuteurs natifs d'une orthographe fondée sur l'alphabet tibétain. La correspondance entre les deux se fait de manière relativement aisée.

Le corpus a été créé en utilisant les fichiers LaTeX de la grammaire détaillée du japhug [22] (Jacques, 2021), éditée par *Language Science Press*, en suivant la méthodologie décrite en section 3.1.2. Puisque nous étudions cette langue uniquement pour la tâche de segmentation (en mots, mais aussi en morphèmes), nous nous sommes seulement concentrés sur les phrases sources des exemples des fichiers, extraites grâce à la balise « `\gll` ».

D'un point de vue des ressources, il faut noter qu'un corpus issu de la documentation du japhug, constitué des enregistrements accompagnés de ses transcriptions et ses traductions, est disponible en ligne sur Pangloss [23], mais nous ne l'avons pas utilisé. Suite aux efforts de (Galliot et al., 2021, 2022), la langue est également présente sur Zenodo [24] et Hugging Face [25], pour favoriser la prise en main par la communauté TAL. De plus, un dictionnaire japhug-chinois-français (Jacques, 2015) est accessible en ligne. Notons également que le japhug constitue la langue la plus représentée dans IMTVault (Nordhoff et Krämer, 2022), avec plus de 3 000 exemples à l'heure actuelle.

Tout comme le mboshi, le japhug a également été étudié pour la tâche de transcription de la parole et de segmentation d'unités (Adams et al., 2021; Macaire, 2021; Guillaume et al., 2022).

3.2.3 Le tsez et le zaar à travers la documentation

Nous nous intéressons également à deux langues, à travers les corpus directement issus de leur documentation.

Le tsez (ddo) est une langue nakho-daghestanienne parlée dans la république du Daghestan en Russie. S'il est principalement oral, il est transcrit et translittéré en utilisant le système d'écriture cyrillique avar (Comrie et Polinsky, à paraître), du fait de leur proximité phonologique. La langue est complexe morphologiquement avec l'utilisation d'un jeu de suffixes conséquent.

Le corpus que nous utilisons principalement provient du *Tsez Annotated Corpus Project* et constitue un recueil de folklores (Abdulaev et Abdulaev, 2010). La langue dispose d'une grammaire en cours de publication (Comrie et Polinsky, à paraître) et d'un dictionnaire d'environ 7 500 mots, au format PDF (Xalilov, 1999).

Le tsez a d'ailleurs déjà étudié dans le cadre de la génération automatique des gloses dans (Zhao et al., 2020) et fait partie des sept langues du défi partagé SIGMORPHON (voir ci-dessous).

Le zaar (say) est une langue afro-asiatique de la branche tchadique, parlée dans le sud de l'État de Bauchi au Nigéria. La jeune génération est bilingue zaar-haoussa, une autre langue tchadique, parlée par plus d'une dizaine de million de locuteurs (Caron, 2015). La langue présente notamment la spécificité d'exprimer le temps, l'aspect et le mode des verbes à travers un clitique placé à sa gauche. Sa morphologie reflète par ailleurs la situation particulière du zaar, à la jonction entre les familles des langues afro-asiatiques et nigéro-congolaises.

Le corpus a été recueilli dans le cadre du projet CorpAfroAs (ANR-06CORP) [26]. Nous utilisons sa version publiée [27] pour Universal Dependencies et annotée dans le cadre du projet Auto-

22. https://github.com/langsci/295.
23. https://pangloss.cnrs.fr/corpus/Japhug.
24. https://zenodo.org/record/5521112.
25. https://huggingface.co/datasets/Lacito/pangloss.
26. https://corpafroas.huma-num.fr/Archives/corpus.php.
27. https://github.com/surfacesyntacticud/SUD_Zaar-Autogramm.

gramm, présenté en section 2.1.2. Chaque morphème y est annoté en gloses, mais aussi en parties du discours, en suivant le format CoNLL-U [28].

3.2.4 Les langues du défi partagé SIGMORPHON 2023 de génération automatique de gloses

La dernière catégorie des langues que nous étudions provient du défi partagé sur la génération automatique de gloses (Ginn et al., 2023), présenté en section 2.3.2. Nous nous concentrons sur les cinq des sept langues que nous avons étudiées dans cette thèse pour cette tâche spécifique ; les deux autres étant l'arapaho (arp), une langue algonquienne parlée dans l'État du Wyoming aux États-Unis, et le nyangbo (nyb), une langue kwa parlée au Ghana. La première a fait notamment l'objet d'une étude par (Zhao et al., 2020) ; pour la seconde, comme les traductions des énoncés glosés n'étaient pas fournies, nous ne l'avons pas considérée dans nos expériences du chapitre 6.

Le **tsez** (ddo), certes déjà étudié dans nos travaux, est représenté ici par la version plus récente du corpus (Abdulaev et al., 2022). Celui-ci comporte davantage de phrases que la version de la section 3.2.3. Nous utiliserons cette version uniquement dans les sections 6.5 et 6.6, consacrées au défi partagé.

Le **gitksan** (git) est une langue tsimshianique parlée en Colombie-Britannique au Canada. Il utilise une variété de clitiques qui rendent la morphologie complexe, plaçant la langue entre les langues analytiques et synthétiques (Rigsby, 1986, 1989). Le giktsan a aussi fait l'objet d'une documentation et dispose de ressources en ligne [29], comme des textes ou un dictionnaire et même d'un analyseur morphologique (Forbes et al., 2021). Le corpus est constitué de trois récits enregistrés par trois locuteurs (Forbes et al., 2017).

Le **lezghien** (lez) appartient à la même famille de langue que le tsez, à savoir les langues nakho-daghestaniennes. C'est également une langue agglutinante, morphologiquement complexe à travers ses suffixes (Haspelmath, 1993). Le corpus est constitué de textes dans le dialecte de Qusar parlé en Azerbaïdjan (Donet, 2014). Nous soulignons ici que la langue a déjà été étudiée par (Moeller et Hulden, 2018) et (Zhao et al., 2020) pour la même tâche de génération automatique de gloses.

Le **natugu** (ntu) est une langue austronésienne de la branche Reefs–Santa Cruz, parlée aux Îles Salomon par environ 5 000 personnes. Une de ses caractéristiques par rapport aux autres langues océaniques est son absence de redoublement (Boerger, 2022). De plus, elle possède une morphologie agglutinante avec une structure verbale complexe (Næss et Boerger, 2008). Le corpus ainsi qu'une grammaire sont accessibles en ligne [30].

L'**uspanteko** (usp) est une langue maya du sous-groupe quiché-mam. Elle est parlée par environ 6 000 locuteurs au Guatémala et par la diaspora (Bennett et al., 2016). Le corpus provient du projet de documentation des langues mayas, OKMA (Pixabaj et al., 2007), pré-traité selon la méthodologie de (Palmer et al., 2010). Ce corpus (et donc la langue) a déjà été étudié pour la génération de gloses par Palmer et al. (2009). Notons que c'est la seule langue parmi les cinq pour laquelle la traduction est en espagnol.

28. https://universaldependencies.org/format.html.
29. http://www.gitxsansimalgyax.com/.
30. https://www.langlxmelanesia.com/tilp.

3.2.5 Statistiques sur les corpus

Le tableau 3.1 présente différentes statistiques relatives aux corpus des langues présentées en sections 3.2.1, 3.2.2 et 3.2.3. Au-delà du nombre de phrases N_{utt}, nous reportons le nombre total d'occurrences et de types (N_{token} et N_{type}, respectivement) ainsi que la longueur moyenne des occurrences et des types (WL et TL, respectivement). De plus, comme nos travaux s'intéressent à deux niveaux de segmentation, nous présentons ces valeurs pour les mots et les morphèmes, lorsque ces informations sont disponibles.

En outre, puisque les ressources utilisées pour la supervision faible provient des 200 premières phrases de chaque corpus pour le mboshi, japhug et tsez, nous présentons séparément les longueurs moyennes et nombres d'unités. Nous pouvons constater que si la longueur moyenne des occurrences dans les données de supervision est proche de la moyenne calculée pour tout le corpus, la situation pour les types (aux deux niveaux) est un peu différente, avec des unités plus courtes que dans le corpus intégral. Par ailleurs, nous indiquons aussi le nombre de morphèmes par mot ($N_{M/W}$), si l'on dispose de la segmentation en morphèmes, ainsi que le nombre de parties du discours (N_{PoS}) pour le zaar.

	langue	mboshi	japhug		tsez		zaar	
	niveau	mot	mot	morph.	mot	morph.	mot	morph.
complet	N_{utt}	5 130	3 628	3 628	2 000	2 000	1584	1584
	WL	4,19	4,73	2,90	5,61	2,81	4,02	3,58
	TL	6,39	7,30	5,41	6,93	5,21	6,47	5,33
	N_{type}	5 312	6 739	2 731	5 733	1 604	2 123	1 452
	N_{token}	30 556	28 579	46 632	20 161	40 331	15 374	17 282
supervision	WL	4,40	4,60	2,85	5,50	2,83	-	
	TL	5,74	6,02	4,29	6,15	4,50	-	
	N_{type}	517	664	493	867	455	-	
	N_{token}	1 132	1 399	2 259	1 696	3 295	-	
	$N_{M/W}$	-	1,63		2,00		1,12	
	N_{PoS}	-	-		-		29	

TABLE 3.1 – Statistiques pour quatre corpus, segmentés en mots et éventuellement en morphèmes (morph.). Le matériel de supervision est créé à partir des 200 premières phrases.

Comme la tâche de génération de gloses est intrinsèquement supervisée (voir section 2.3), nous présentons les corpus du défi partagé SIGMORPHON dans le tableau 3.2, séparés selon la répartition officielle en entraînement, développement et test. Nous reportons également des statistiques sur les données d'entraînement, en reprenant des informations de (Ginn et al., 2023) comme le nombre de morphèmes par mot ($N_{M/W}$) ou de gloses par morphèmes ($N_{G/M}$).

Nous pouvons observer de grandes différences dans les tailles de données d'entraînement dans les corpus, allant de très peu de ressources pour le gitksan à environ 700 phrases pour le lezgi et le natugu, voire plus de ressources pour le tsez ou l'uspanteko. Soulignons ici que le nombre de morphèmes est l'indicateur principal de la quantité de ressources : si le corpus tsez est constitué de presque trois fois moins de phrases que le corpus uspanteko, il comporte davantage d'occurrences de morphèmes (mais moins de diversité). De plus, notons que la valeur de $N_{M/W}$ renseigne d'une part sur la complexité morphologique de la langue et que celle de $N_{G/M}$ indique d'autre part l'ambiguïté de l'étiquetage en gloses. Par exemple, le tsez est morphologiquement

langue	ddo	git	lez	ntu	usp
entraînement	3 558	31	701	791	9 774
développement	445	42	88	99	232
test	445	37	87	99	633
N_{type} (entraînement)	1 883	138	1 297	1 245	3 533
N_{token} (entraînement)	74 334	429	10 497	16 341	60 458
$N_{M/W}$ (entraînement)	2,0	1,6	1,5	1,6	1,4
$N_{G/M}$ (entraînement)	1,0	1,3	1,0	1,0	1,2
langue de documentation	EN	EN	EN	EN	ES

TABLE 3.2 – Statistiques des corpus du défi partagé SIGMORPHON sur la génération automatique des gloses. La première partie indique le nombre de phrases par langue. La deuxième présente les nombres d'occurrences et de types de *morphèmes*, de morphèmes par mots ($N_{M/W}$) et de gloses par morphèmes ($N_{G/M}$) sur le corpus d'*entraînement*. La troisième correspond à la langue de traduction.

plus complexe que les autres langues avec peu d'ambiguïté pour les gloses, tandis que le gitksan ou l'uspanteko présentent une moindre complexité morphologique pour plus d'étiquettes possibles par morphème. En outre, nous avons majoritairement l'anglais comme langue de documentation, mais aussi l'espagnol pour l'uspanteko. Cela nous permettra d'évaluer les possibilités d'adaptation de nos modèles pour d'autres langues de documentation que l'anglais.

3.3 Conclusion

Ce chapitre esquisse les différentes ressources disponibles dans le cadre de la documentation des langues. Nous nous sommes néanmoins principalement concentrés sur les données utilisées dans le cadre de cette thèse. Nous avons étudié des ressources constituées par les linguistes à différentes étapes de la documentation. Les enregistrements avec les traductions sur le terrain constituent le fondement. Les annotations subséquentes comme la transcription mais également les gloses sont également utilisées en TAL. Notons que ces données sont disponibles sur des plateformes linguistiques, qui sont encore, pour le moment, peu explorées par le domaine du TAL pour des raisons d'ergonomie entre autres (Michaud et al., 2018; Nordhoff, 2020). Le travail des linguistes aboutissent également à des grammaires et des lexiques. Il s'agit d'un complément d'un point de vue du TAL, dans la mesure où les corpus sont souvent constitués de récits, là où les exemples des grammaires présentent des informations linguistiques comme les gloses, sans avoir le contexte de la phrase dans un texte.

Par ailleurs, nous avons constaté les évolutions récentes dans l'accessibilité aux ressources dans le domaine du TAL, par exemple pour ces deux dernières années, à travers le recueil IMT-Vault (Nordhoff et Krämer, 2022) ou le défi SIGMORPHON de 2023, ce qui permet de faciliter l'accès aux tâches de documentation automatique des langues.

Enfin, le tableau 3.3 est un récapitulatif des données concernant notamment les annotations à disposition. Nous reportons aussi la langue (ou les langues) de documentation pour chaque corpus. Concernant la phrase source, les langues que nous avons étudiées ont été transcrites avec un alphabet latin adapté (pour la plupart des langues), cyrillique (pour le lezghien) ou l'alphabet phonétique international (comme le japhug).

langue ou corpus	segmentation		gloses	PoS	langue(s) cible(s)	chapitre(s)
	mots	morphèmes				
mboshi	✓	✗	✗	✗	EN, FR	4
japhug	✓	✓	✗	✗	EN	4, 5
tsez	✓	✓	✓	✗	EN, RU	5, 6
zaar	✓	✓	✓	✓	EN	6
SIGMORPHON	✓	✓	✓	✗	EN, ES	6

TABLE 3.3 – Récapitulatif des différentes langues et corpus associés.

Chapitre 4

Des approches faiblement supervisées pour la segmentation en mots

4.1 Introduction

Ce chapitre est consacré à la tâche de segmentation en mots dans le cadre de la documentation automatique des langues, comme définie à la section 2.2.1. La figure 4.1 présente notre configuration, où nous supposons que l'enregistrement audio a été correctement transcrit (**S1**). L'objectif est alors d'identifier dans la phrase les frontières des mots, représentées par des espaces (« ») en **S2**. Nous soulignons ici que sont considérés comme mots, les unités définies comme telles par le linguiste, car notre objectif principal est d'assister cette étape d'annotation.

Entrée **S1**	Phrase non segmentée	kʏndʑixtʏɣχsɯmpjɤtunɯ
Sortie **S2**	Segmentation en mots	kʏndʑixtʏɣ χsɯm pjɤtunɯ

FIGURE 4.1 – La tâche de segmentation en mots pour l'exemple 2.1.

Comme évoqué en section 2.2, pour les langues très peu dotées, les méthodes neuronales ne conviennent pas du fait de la petite quantité de données. C'est pourquoi, le choix jusque là s'est porté sur des méthodes statistiques et non supervisées, comme les modèles bayésiens non paramétriques (voir section 2.2 et Godard (2019)), afin de pouvoir être utilisables tôt dans le processus de documentation. S'il permet bien de segmenter un corpus, indépendamment de sa taille, les performances de ces modèles laissent encore une marge de progression en termes de qualité, et donc d'utilité.

De fait, en se limitant à des approches n'utilisant aucune donnée de supervision, le modèle est certes adapté à toutes les langues mais surtout à aucune d'entre elles ; des ressources existantes et accessibles sont en réalité mises de côté. Bien qu'elles ne soient pas d'envergures comparables aux langues fréquemment étudiées en TAL, les langues en cours de documentation disposent de données, notamment des textes ou des lexiques, comme le rappelle Bird (2020). Nous nous intéressons donc aux différentes manières d'intégrer ces ressources dans un modèle de segmentation en mots, en nous plaçant dans des cas réalistes de documentation.

Les contributions exposées dans cette partie concernent :

— la ré-implémentation du modèle bayésien non paramétrique `dpseg` (Goldwater et al., 2009) en Python et de variantes reposant sur le processus de Pitman-Yor ;

— l'utilisation de ressources auxiliaires telles que des phrases déjà segmentées ou des lexiques de la langue étudiée, intégrées au modèle de segmentation à travers diverses stratégies de supervision faible ;

— l'exploration d'une situation d'apprentissage incrémental, simulant une correction progressive des segmentations automatiques par un expert de la langue ;
— la comparaison des unités obtenues par le modèle, avec et sans supervision faible, par rapport à un texte de référence segmenté en mots et en morphèmes.

Tout d'abord, nous décrivons en section 4.2 la méthodologie générale concernant la segmentation en mots avec le modèle dpseg et sa réimplémentation en Python. La section 4.3 décrit les différentes stratégies pour y intégrer les ressources linguistiques existantes pour la langue étudiée. Trois expériences ont été menées pour la segmentation faiblement supervisée en mots, présentées dans la section 4.4. Les éléments de ce chapitre correspondent à (Okabe et al., 2021) et à (Okabe et Yvon, 2022a), ainsi qu'au document de travail rédigé pour le projet CLD2025 (Okabe, 2021).

4.2 La segmentation avec dpseg

Comme nous avons vu en section 2.2, la tâche de segmentation en mots s'effectue principalement de manière non supervisée, dans le cadre de la documentation automatique des langues. En effet, les données disponibles pour ces langues ne suffisent souvent pas pour entraîner des modèles supervisés.

Parmi les approches présentées en section 2.2, nous choisissons d'utiliser dpseg malgré l'existence de modèles plus sophistiqués, également bayésiens non paramétriques, comme ceux de (Mochihashi et al., 2009), de (Löser et Allauzen, 2016) ou l'*Adaptor Grammar* (Johnson et al., 2007). En effet, sa robustesse et ses performances sur les corpus de documentation, associées à sa complexité moindre, en font un point de départ permettant d'intégrer plus simplement les modifications présentées dans ce chapitre, qui pourront ensuite être transposées vers des modèles plus sophistiqués. De plus, par rapport à la meilleure méthode identifiée par Godard (2019), pypshmm de (Löser et Allauzen, 2016), dpseg obtient des segmentations plus stables.

Par ailleurs, nous nous concentrons sur la version unigramme du modèle dpseg, bien qu'elle présente des limites connues par rapport à sa version bigramme ; l'hypothèse d'indépendance des unités, inhérente à la vision unigramme, empêche notamment la segmentation de mots apparaissant souvent ensemble (par exemple, la locution « parce que » en français, qui serait segmentée comme « parceque ») (Goldwater et al., 2009). En effet, malgré cela, Godard (2019) observe des résultats similaires voire meilleurs sur les données que nous étudions, en utilisant la version unigramme de dpseg par rapport à son équivalent bigramme, initialement choisi selon les conclusions de (Goldwater et al., 2009).

4.2.1 Métriques d'évaluation

Nous évaluons la segmentation selon les métriques standards de précision (P), rappel (R) et score F1 (F), définies par les équations (4.1), (4.2) et (4.3), en notant VP les vrais positifs, FP les faux positifs et FN les faux négatifs. Pour une meilleure lisibilité, nous les multiplions par 100.

$$\text{Précision} = \frac{\text{VP}}{\text{VP} + \text{FP}} \tag{4.1}$$

$$\text{Rappel} = \frac{\text{VP}}{\text{VP} + \text{FN}} \tag{4.2}$$

$$\text{F-score} = 2 * \frac{\text{Précision} * \text{Rappel}}{\text{Précision} + \text{Rappel}} \tag{4.3}$$

Notons que la gravité des erreurs de frontières est variable au sein d'une phrase ou d'un corpus ; un mot fréquent systématiquement mal segmenté pose davantage de soucis que si c'est un hapax (mot n'apparaissant qu'une seule fois dans le corpus). C'est pourquoi les métriques PRF sont calculées aux trois niveaux ci-dessous, en suivant la méthodologie de Goldwater (2006) :

— au niveau des caractères (B pour *boundary* en anglais), nous étudions s'il y a une frontière ou non après le caractère considéré (BP, BR, BF). La figure 4.2 représente une phrase de référence et une version segmentée au niveau des frontières. Nous ne tenons pas compte de la dernière position de la phrase, car la valeur est nécessairement 1 (fin de phrase, donc fin de mot).
— au niveau des mots (W pour *word* en anglais), nous comparons si les segments correspondent (WP, WR, WF) pour chaque phrase, en tenant compte de l'ordre.
— au niveau du lexique (L pour *lexicon* en anglais), à savoir sur l'ensemble du corpus, nous analysons la concordance des types de mots (LP, LR, LF).

référence	kɤndʑixtɤɣ	χsɯm	pjɤtunɯ
prédiction	kɤndʑi xtɤɣ	χsɯm	pjɤtunɯ

position	i_{00}	i_{01}	i_{02}	i_{03}	i_{04}	i_{05}	i_{06}	i_{07}	i_{8}	i_{09}	i_{10}	i_{11}	i_{12}	i_{13}	i_{14}	i_{15}	i_{16}	i_{17}	i_{18}	i_{19}	i_{20}
	k	ɤ	n	d	ʑ	i	x	t	ɤ	ɣ	χ	s	ɯ	m	p	j	ɤ	t	u	n	ɯ
référence	0	0	0	0	0	**0**	0	0	0	1	0	0	0	1	0	0	0	0	0	0	1
prédiction	0	0	0	0	0	**1**	0	0	0	1	0	0	0	1	0	0	0	0	0	0	1

FIGURE 4.2 – Une phrase de référence en japhug et une segmentation à évaluer, où les valeurs des variables de frontières sont explicites.

Nous pouvons calculer les valeurs des différentes métriques en utilisant l'exemple de la figure 4.2. Premièrement, au niveau des frontières, il y a 2 positions à considérer où $b_i = 1$ pour la référence (le dernier 1 correspond à la fin de la phrase, pour rappel). Dans la segmentation, deux des trois positions identifiées sont correctes (VP $= 2$), la troisième est fausse (FP $= 1$) et aucune frontière correcte n'a été oubliée (FN $= 0$). Par conséquent, BP $= 66{,}7$, BR $= 100$ et BF $= 80{,}0$. Ensuite, au niveau de la phrase, nous observons que deux des trois mots de référence sont correctement identifiés (VP $= 2$) et le troisième n'est pas présent dans la prédiction (FN $= 1$). L'erreur de segmentation aboutit à deux unités non présentes dans la phrase de référence (FP $= 2$), ce qui donne : WP $= 50{,}0$, WR $= 66{,}7$ et WF $= 57{,}1$. Au niveau des types, le calcul sera identique au précédent, comme nous n'avons pas représenté de lexique et que cette phrase ne contient aucune répétition.

Intuitivement, il semble aisé d'obtenir des scores B élevés car il y a très souvent plus de non frontières que de frontières dans une phrase et, dans un cas extrême, prédire systématiquement des frontières donnerait un rappel valant 100. En outre, il est plus difficile d'avoir de hauts scores L comparativement aux scores W, car, pour ces derniers, l'identification de mots fréquents assure des valeurs plus élevées, là où les scores L accordent autant d'importance à la segmentation en unités rares ou fréquentes.

À titre indicatif, nous présentons également les statistiques génériques, utilisées en section 3.2.5, calculées sur les segmentations des modèles : le nombre d'occurrences de mots N_{token} et de types N_{type} ainsi que la longueur moyenne des occurrences WL et des types TL.

4.2.2 Paramétrage du modèle

Nous utilisons les paramètres par défaut de dpseg, tels qu'ils sont définis par Goldwater et al. (2009). Nous fixons donc le paramètre de concentration $\alpha = 20$ et la probabilité a priori de finir un mot $p_{\#} = 0{,}5$. Goldwater et al. (2009) observe à leur sujet que d'autres valeurs sont relativement équivalentes en terme de segmentation : la variation de ces méta-paramètres permet plutôt d'ajuster le compromis entre les performances aux niveaux des occurrences et des types de mots.

En ce qui concerne l'initialisation des variables de frontières, dpseg propose différentes configurations : il est non seulement possible d'en donner aucunes ou toutes, mais aussi de commencer aléatoirement ou avec les vraies frontières. Goldwater et al. (2009) explique et vérifie empiriquement qu'en pratique, l'initialisation n'a pas d'impact majeur sur les performances. En effet, la théorie de l'échantillonnage de Gibbs garantit la convergence vers la distribution a posteriori, et ce, quelle que soit la méthode d'initialisation. Nous choisissons donc d'attribuer de façon aléatoire les valeurs de départ, comme (Goldwater et al., 2009).

Nous conservons la même valeur de graine (*seed*) pour obtenir des résultats comparables tout au long de ce chapitre. Celui-ci intervient non seulement dans l'initialisation des variables de frontières b_i mais également lors des successions de tirages aléatoires.

Comme (Goldwater et al., 2009) et (Godard, 2019), nous effectuons 20 000 itérations d'échantillonnage de Gibbs avec 10 degrés de température dans le recuit simulé. Nous utilisons la segmentation à la fin de la dernière itération comme sortie de notre modèle. En effet, Goldwater (2006) a étudié d'autres méthodes, comme la moyenne de 10 échantillons espacés de 100 itérations du même lancer ou de 10 lancers initialisés différemment, et n'observe que très peu de différences dans les métriques.

4.2.3 Réimplémentation en Python

Afin de pouvoir intégrer nos propres modifications par la suite, nous avons dans un premier temps réimplémenté en Python, le modèle dpseg, originellement écrit en C++ [1]. Afin de vérifier leurs cohérences, le tableau 4.1 présente les résultats obtenus avec ces deux versions du modèle, dans les mêmes conditions expérimentales, décrites dans la section précédente, hormis le nombre d'itérations, qui est réduit à 1 000. En guise de complément, le corpus mboshi (5K) a également été tronqué en 500 (0,5K), 1 000 (1K) et 2 000 phrases (2K).

Dans le tableau 4.1, nous remarquons que les scores aux trois niveaux d'évaluation sont semblables, voire meilleurs avec l'implémentation en Python, pour les quatre tailles de texte. La différence principale provient de la génération des nombres aléatoires. De plus, les statistiques génériques sur le texte suggèrent également la correspondance entre les deux segmentations. Nous utiliserons donc pour la suite des expériences notre implémentation de dpseg.

Les tendances propres à dpseg Nous étudions plus en détail la qualité de segmentation du modèle dpseg non supervisé sur le corpus en entier. Tout d'abord, en l'évaluant à travers quatre tailles de données, nous constatons naturellement que plus il y a de phrases, plus les scores BF et WF augmentent. Le F-score des types semble, en revanche, stagner et évolue moins vite, confirmant les difficultés énoncées à ce niveau en section 4.2.1.

En outre, les statistiques sur le texte nous indiquent deux tendances. D'une part, les unités générées sont trop courtes aux niveaux des occurrences (un WL de 2,97 avec notre implémentation

1. https://homepages.inf.ed.ac.uk/sgwater/software/dpseg-1.2.1.tar.gz.

taille	0,5K		1K		2K		5K	
dpseg	C++	Python	C++	Python	C++	Python	C++	Python
BP	40,9	42,8	45,3	46,5	46,9	48,8	50,1	52,7
BR	83,6	85,1	82,1	83,4	81,6	82,3	78,2	78,6
BF	55,0	57,0	58,3	59,7	59,6	61,2	61,1	63,1
WP	16,8	19,1	19,8	22,2	21,5	24,2	24,4	27,8
WR	31,3	34,7	33,2	37,0	34,9	38,2	35,8	39,1
WF	21,9	24,7	24,8	27,8	26,6	29,6	29,0	32,5
LP	29,7	29,7	36,4	35,6	35,4	37,4	40,1	46,2
LR	6,55	7,40	6,71	7,37	5,30	6,41	5,74	7,76
LF	10,7	11,9	11,3	12,2	9,21	11,0	10,1	13,3
WL	2,30	2,36	2,48	2,50	2,60	2,67	2,85	2,97
TL	2,92	2,92	3,16	3,24	3,30	3,42	3,59	3,79
N_{type}	232	263	335	376	469	537	760	891
N_{token}	5410	5272	10376	10280	20404	19850	44816	43024

TABLE 4.1 – Comparaison des deux implémentations de dpseg en C++, l'originale, et en Python pour différentes tailles du corpus mboshi (1 000 itérations).

contre 4,19 pour le corpus 5K de référence), mais surtout des types, avec plus de deux caractères de différence (un TL de 3,79 caractères contre 6,39). Ceci traduit le phénomène de *sur-segmentation* déjà observé par (Goldwater et al., 2009) et (Godard, 2019) pour dpseg. Nous remarquons, de plus, un écart important entre la précision et le rappel des frontières, traduisant la propension du modèle à mettre plus souvent des frontières que nécessaire. D'autre part, le nombre de types est trop bas (un N_{type} de 891 contre 5 312 dans la référence). La différence notable entre WF et LF nous suggère, par ailleurs, que le modèle identifie correctement des unités fréquentes au détriment de celles qui sont rares.

Ces deux problèmes sont liés, dans la mesure où la sur-segmentation aboutit à des unités segmentées courtes et artificiellement plus fréquentes, ce qui laisse de côté celles qui sont plus rares et longues. Le tableau 4.2 présente la répartition des unités selon leur fréquence pour le texte 5K de référence et sa version segmentée par dpseg (en Python). Nous considérons les hapax comme étant rares, les mots apparaissant plus souvent que la moyenne pour le texte comme fréquents et le reste comme étant « dans la moyenne ».

	Rare	Moyenne	Fréquent	Fréquence moyenne
Référence	2 985	1 687	640	5,75
Proportion	56,2 %	31,8 %	12,0 %	-
Segmentation	6	709	176	48,3
Proportion	0,67 %	79,6 %	19,8 %	-

TABLE 4.2 – Fréquence (discrétisée) des unités dans le texte de référence mboshi (5K) et sa segmentation par dpseg. La proportion est calculée par rapport au nombre total de types.

Plus de la moitié du corpus mboshi est constituée de mots n'apparaissant qu'une seule fois, alors que la segmentation avec dpseg ne parvient à générer que 6 segments de la sorte. Notons également la différence dans la fréquence moyenne : seuls 12,0 % des mots apparaissent plus de

5 fois dans la référence. En utilisant le seuil de dpseg pour filtrer, il n'y aurait plus que 66 types de mots qui seraient considérés comme fréquents, soit 1,20 % des types. Nous constatons ainsi que la référence semble bien suivre une distribution en loi de puissance, la loi de Zipf, avec peu de mots très fréquents et une large proportion de mots rares. En revanche, la segmentation obtenue avec dpseg présente une distribution des types assez différente, avec manifestement trop peu d'unités rares. À titre indicatif, les mots apparaissant deux fois (dis legomonon) représentent 17,7 % et 1,57 % du vocabulaire pour la référence et la segmentation, respectivement.

4.2.4 Extensions de dpseg

Par ailleurs, cette réimplémentation nous a permis de mettre en en place quelques améliorations dans le modèle que nous détaillons ci-dessous.

Variante basée sur les processus de Pitman-Yor, pypseg Nous avons conçu un modèle suivant les mêmes principes que dpseg, mais en utilisant les processus de Pitman-Yor (Pitman et Yor (1997) ; section 2.2.4) à la place du processus de Dirichlet ; nous l'avons donc nommé pypseg. En effet, comme il permet, en théorie, de générer davantage de types de mots, face au constat que nous venons de faire en section 4.2.3 sur le manque de mots, en particulier rares, nous pouvons nous attendre à de meilleurs résultats de segmentation. Rappelons que (Teh, 2006b) et (Mochihashi et al., 2009) ont eu recours à ce processus pour justement mieux reproduire la distribution en loi de puissance des mots.

Au-delà de l'ajout d'un méta-paramètre additionnel d, le paramètre d'escompte, ce changement signifie que les tables du restaurant ont un impact direct sur les probabilités ; le nombre total de tables $K(\boldsymbol{z}_{-i})$ influe fortement la propension à générer un nouveau mot et nécessite d'être mis à jour en continu. De plus, dans l'équation de dpseg (2.5), le nombre de tables étiquetées par le mot w n'intervenait pas ; désormais, la modélisation doit prendre en compte explicitement les assignations des clients aux tables.

Échantillonnage des méta-paramètres À la lumière de Teh (2006a), nous effectuons également, après chaque itération d'échantillonnage de Gibbs sur le texte, un ré-échantillonnage des paramètres de concentration α et d'escompte d.

En effet, ces deux paramètres peuvent être appris à travers les données. En suivant l'implémentation du ré-échantillonnage présentée en annexe C de (Teh, 2006a), nous considérons que α suit une loi de probabilité a priori Gamma et d une loi Beta. On obtient alors des postérieurs Gamma et Beta correspondants, dont les paramètres peuvent être calculés au moyen de variables auxiliaires dépendant du nombre total de clients et de tables dans le restaurant.

En pratique, nous utilisons les distributions a priori suivantes : $\alpha \sim \text{Gamma}(1, 1)$ et $d \sim \text{Beta}(1, 1)$.

Le code correspondant à cette section est disponible à : https://github.com/shuokabe/pyseg.

4.3 Améliorer la segmentation par une supervision faible

Comme nous avons observé en section 2.2, l'approche utilisée pour la segmentation en mots des langues très peu dotées se fait principalement sans supervision. Or, comme le note Bird (2020),

dans le cadre de la documentation des langues, il arrive souvent que des travaux antérieurs aient produit des données ; des phrases ont déjà été annotées ou bien des lexiques (incomplets) ont été établis. Si l'envergure de ces ressources est typiquement modeste pour des modèles nécessitant des données en large quantité, comme pour les réseaux de neurones, d'autres types de modèles peuvent en bénéficier grandement. Ce chapitre s'inscrit de fait dans cette perspective : l'objectif est de s'appuyer sur ces ressources, qui sont rarement exploitées par les modèles de documentation automatique des langues, pour mieux initialiser, et par là, accélérer les premières séries d'annotations.

Notons cependant, comme évoqué en section 2.2.5, que la segmentation en mots dans des contextes similaires a déjà été abordée en utilisant des connaissances linguistiques spécifiques à la langue, comme des règles morphologiques, dans le modèle de l'*Adadptor Grammar* (Godard et al., 2018b) ou en mettant à profit les traductions des textes (Godard et al., 2019). Notre travail explore d'autres ressources disponibles dans un cadre réaliste de documentation de langues. Nous avons essentiellement identifié deux manières de les intégrer : d'une part, en effectuant une semi-supervision de l'échantillonnage de Gibbs, d'autre part, en augmentant la probabilité des unités connues pour la supervision lexicale. Ces méthodes seront respectivement marquées par le préfixe `g.` et `d.` par la suite.

4.3.1 Les informations sur les frontières

Dans cette catégorie, nous regroupons les ressources donnant accès à des informations sur la présence ($b_i = 1$) ou l'absence ($b_i = 0$) de frontières à des positions précises dans une phrase. En effet, notre stratégie pour ces ressources sera de rendre l'échantillonnage de Gibbs semi-supervisé, à la manière de (Sirts et Goldwater, 2013) ; lorsque la supervision concerne le statut de la frontière (à savoir la valeur de b_i), il n'est plus nécessaire d'arbitrer entre les deux possibilités.

`g.sparse`, signaler uniquement la présence de frontières Un premier cas où les informations sur les frontières peuvent être accessibles est lorsqu'un enregistrement est retranscrit, comme ci-dessous pour une phrase en mboshi (pré-traitée).

SIL subhu la tsosa SIL idi la mwanyaa SIL

Nous pouvons voir, à travers les marqueurs de silence « SIL », les positions des pauses effectuées par le locuteur. Naturellement, les deux unités au début et à la fin indiquent les limites de la phrase ; le deuxième, dans cet exemple, permet de faire l'hypothèse d'une frontière de mots.

Par conséquent, dans cette phrase, à chaque fois que l'échantillonneur considère la position entre le « a » et le « i », la valeur de la frontière b_i sera systématiquement 1. Cette méthode fixe uniquement les positions des frontières de mots et laisse le modèle choisir les valeurs de b_i pour les autres positions. La supervision donne alors des informations incomplètes, éparses, d'où son appellation `g.sparse`.

Notons que cette méthode peut s'appliquer à d'autres signaux dans l'enregistrement ; des informations concernant la prosodie pourraient également être utilisées.

`g.dense`, indiquer la présence et absence de frontières Lorsque des phrases entièrement segmentées par un annotateur sont disponibles, nous pouvons utiliser ces informations (denses, d'où son nom de `g.dense`) sur les frontières. Nous disposons alors non seulement de toutes les

frontières des mots pour ces phrases ($b_i = 1$), mais également de l'assurance qu'il *n'* y a *pas* de frontières ($b_i = 0$) à toutes les autres positions.

Cette méthode permet alors d'accéder à des informations concentrées sur quelques phrases, qui seront, par là, correctement représentées par le modèle. Leurs mots seront en effet nécessairement présents dans le restaurant associé au modèle et n'en disparaîtront pas au fil des itérations d'échantillonnage.

`g.random`, situation théorique La dernière stratégie que nous avons implémentée l'est uniquement à des fins de contraste. L'idée est d'observer si la concentration des informations sur les frontières comme dans `g.dense` est plus bénéfique qu'une situation où la même quantité d'informations de présence et d'absence de frontières serait donnée mais de manière diffuse sur tout le texte. Cela revient donc à fournir les valeurs de b_i pour des positions choisies aléatoirement.

En ce sens, nous pouvons considérer les méthodes `g.sparse` et `g.dense` comme des cas particuliers, où, toutes les informations de supervision indiquent uniquement la présence de frontières, dans la première, ou le hasard a concentré les positions données sur quelques phrases, dans la seconde.

supervision	k	ɤ	n	d	ʑ	i	x	t	ɤ	ɣ	χ	s	ɯ	m	p	j	ɤ	t	u	n	ɯ
sans	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	1
`g.sparse`	-1	-1	-1	-1	-1	-1	-1	-1	-1	1	-1	-1	-1	1	-1	-1	-1	-1	-1	-1	1
`g.dense`	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	1
`g.random`	-1	0	0	-1	-1	0	-1	-1	0	1	-1	0	0	-1	0	-1	0	-1	-1	0	1

FIGURE 4.3 – Exemple d'informations de supervision pour la phrase japhug de la figure 2.1 « kɤndʑixtɤɣ χsɯm pjɤtunɯ ».

La figure 4.3 illustre les différentes stratégies décrites pour une phrase d'exemple. La valeur $b_i = 1$ correspond à la présence d'une frontière et $b_i = 0$, à l'absence de frontière, dans les situations faiblement supervisées. Les positions sans aucune information sont marquées par la valeur $b_i = -1$, attribuée par défaut à tout le texte.

Disponibilité de l'information Si nous comparons les deux types de ressources à l'origine de `g.sparse` et `g.dense`, la première est accessible pour tout le corpus, s'il est bien retranscrit depuis un enregistrement, bien qu'elle souffre d'une présence sporadique, tandis que la seconde se concentre sur quelques phrases et divise en quelque sorte le corpus en fonction de la supervision.

Équilibrer la supervision à travers la quantité d'information À des fins de comparaison entre ces trois méthodes, nous utilisons la quantité d'information (en bits) donnée par les informations de frontières ou de non frontières. De fait, elle est définie par l'équation (4.4) pour un événement de probabilité p.

$$I(p) = -\log_2(p) \tag{4.4}$$

Dans nos expériences, nous considérons une probabilité de $p = \frac{1}{5}$ lorsqu'une frontière est donnée ($b_i = 1$) et de $p = \frac{4}{5}$ lorsqu'il s'agit d'une absence de frontière ($b_i = 0$), plus probable car il s'agit d'un événement plus commun. Les valeurs ont été sélectionnées par rapport à un corpus restreint du mboshi, où l'on observe en moyenne une frontière tous les cinq caractères.

Pour l'approche `g.dense`, nous faisons le choix de supposer connues les frontières et non frontières des 200 premières phrases du corpus, une valeur certes arbitraire mais que nous avons estimée raisonnablement accessible lors de la documentation. En se basant sur cette quantité d'information de supervision, nous calibrons les deux autres stratégies : cela revient notamment à 7 % des frontières de l'ensemble du texte, données pour `g.sparse` en mboshi. Pour `g.random`, nous donnons exactement le même nombre de frontières et non frontières que `g.dense` mais réparties sur l'ensemble du texte.

4.3.2 Utilisation d'un dictionnaire

Lors de la documentation des langues, un autre type de ressources est également disponible : un dictionnaire de mots. Il peut être déjà existant, par les travaux antérieurs sur la langue, ou constitué tout au long du projet. Il s'agit alors souvent d'un recueil de *types* de mots associés aux définitions ou traductions et éventuellement des formes fléchies (à travers la conjugaison ou la déclinaison notamment). En comparaison avec la stratégie précédente, la fréquence des unités ne joue plus de rôle et toutes seront, par conséquent, sur un pied d'égalité.

Dans nos expériences, nous reproduisons cette situation en créant un dictionnaire à partir des 200 premières phrases du corpus, toujours à des fin de comparaison entre les méthodes de supervision faible. Il faut noter ici qu'il correspond davantage à une liste de mots, dans la mesure où ce ne sont justement pas des lemmes : les formes fléchies d'un même lemme peuvent donc être présentes (par exemple, en français, « suis », « étaient », « sera »), tandis qu'un véritable dictionnaire ne contiendra plutôt qu'une seule entrée (« être »). Ce choix implique toutefois que les lemmes les plus fréquents seront vraisemblablement plus représentés malgré tout, à travers diverses flexions. De plus, en théorie, les mots concernés par cette supervision seront plus adaptés pour notre tâche, étant donné que ce sont les formes réellement observées, contrairement aux lemmes, moins nombreux (comparer la fréquence des infinitifs et des formes conjuguées en français notamment).

Nous sommes donc dans une situation expérimentale relativement avantageuse car la ressource utilisée pour la supervision correspond exactement aux entités présentes dans le texte et ne contient rien d'externe ; dans un cas réel, le dictionnaire pourrait avoir été créé à partir d'autres corpus (ce qui impliquerait la présence de mots absents du texte à segmenter) et ne pas contenir certaines flexions.

Les éléments des listes de mots peuvent être intégrés de diverses manières dans notre implémentation. En effet, dans l'équation (4.5) caractérisant le modèle `dpseg`, déjà présenté en section 2.2.3, le numérateur comporte deux termes que nous pouvons ajuster : la fréquence de l'unité w dans le restaurant ($n_w^{(h^-)}$) et la probabilité de générer le type w (la distribution de base $P_0(w|h^-)$).

$$P(w|h^-, \alpha) = \frac{n_w^{(h^-)} + \alpha P_0(w|h^-)}{n^- + \alpha} \tag{4.5}$$

`d.count`, ajouter un taux fixe aux mots du dictionnaire Une première méthode naïve est d'initialiser le lexique de `dpseg` par les mots du dictionnaire, en les ajoutant d'emblée à des tables du restaurant avec un taux fixe λ ($\lambda \in [0, 1]$). Cette modification signifie que les mots connus sont assurés d'être présents tout au long des itérations, grâce à une pseudo-fréquence de ($n_w^{(h^-)} + \lambda$) dans l'équation (4.5).

`d.mix`, favoriser les mots connus via P_0 L'incorporation des mots dans le dictionnaire peut également se faire en modifiant la distribution de base, P_0, définie en section 2.2.3 (équation (2.3)), par la fonction suivante :

$$P'_0(w) = \frac{\lambda}{|D|} * \mathbb{1}_{\{w \in D\}}(w) + (1 - \lambda) * P_0(w), \tag{4.6}$$

où $\lambda \in [0, 1]$ est un méta-paramètre, $|D|$ le nombre de types dans le dictionnaire de supervision D et $\mathbb{1}_{\{w \in D\}}$ la fonction indicatrice, active pour les mots présents dans le dictionnaire D. Cette nouvelle fonction permet de favoriser la probabilité de générer de nouveau un mot connu, grâce au facteur $\lambda/|D|$ du premier terme. À l'inverse, les mots inconnus seront pénalisés de $(1 - \lambda)$ par rapport au modèle standard, à savoir P_0.

Cette méthode a déjà été employée par (Goldwater et al., 2009) pour représenter un meilleur modèle de langue ; l'objectif était cependant de mettre en évidence les problèmes intrinsèques à l'hypothèse unigramme de `dpseg`.

Par ailleurs, nous remarquerons que dans l'équation (4.6), la distribution de base originelle, dans le second terme, peut être elle aussi modifiée ; d'autres modèles de caractères peuvent être employés.

Modifier les modèles de longueur et de caractères dans P_0 Jusque là, le modèle de base pour P_0 est composé, d'une part, d'un modèle de longueur et, d'autre part, d'un modèle de caractères, tous deux génériques et indépendants de la langue (ou même du texte à segmenter). Nous explorons donc plusieurs configurations en faisant varier ces deux modèles pour une représentation plus proche de la réalité grâce aux informations de supervision.

Comme nous le rappelons explicitement avec l'équation (4.7), la distribution de base P_0 est constituée de deux facteurs, pour un mot w de longueur L : un modèle de longueur simple P_l et un modèle de caractères, calculant le produit de la probabilité P_c de chacun des caractères. Par défaut dans `dpseg` (voir l'équation (2.3)), P_l suit une loi géométrique de paramètre $p_\#$, défini à l'avance par l'utilisateur, et P_c est uniforme, ce qui signifie que tous les caractères présents dans le texte ont la même probabilité. Notons également que P_l décourage de manière exponentielle les mots longs, par construction.

$$P_0(w) = P_0(c_1 \cdots c_L) = P_l(l = L) * \prod_{i=0}^{L} P_c(c = c_i) \tag{4.7}$$

Nous avons tout d'abord essayé d'améliorer la distribution de base P_0, en conservant l'approche unigramme de caractères. Trois variantes ont été implémentées, afin de mieux modéliser les *types* dans la langue, en se basant sur le dictionnaire de supervision D :

— `d.poisson+uniform` remplace uniquement le modèle de longueur P_l dans l'équation (4.7) par une loi de Poisson. Ceci fait écho aux travaux antérieurs de segmentation où une telle correction a été appliquée, comme (Xu et al., 2008; Mochihashi et al., 2009). Le paramètre λ associé à la distribution correspond à la moyenne des longueurs des mots dans le dictionnaire de supervision D. Le modèle de caractères est ici uniforme, inchangé.

— `d.poisson+1gram` repose sur la même distribution de Poisson que la méthode précédente, mais utilise un modèle de caractères où le calcul de $P_c(c_i)$ est basé sur leur distribution empirique dans D. Cela permet alors de mieux représenter les probabilités des caractères dans la langue, en tenant compte des fréquences.

— `d.empirical+1gram` utilise la distribution empirique non seulement pour les caractères (P_c) comme la méthode précédente, mais aussi pour les longueurs des mots (P_l).

Comme ces stratégies n'ont malheureusement pas apporté d'améliorations concluantes et sont strictement surpassées par la méthode suivante, nous ne les reporterons pas dans cette thèse.

`d.2gram`, une distribution de base bigramme Nous avons aussi étendu l'approche précédente vers un modèle bigramme pour P_0. En utilisant un caractère spécifique pour les débuts de mots (« < ») ainsi qu'un autre pour les fins (« > »), nous pouvons calculer, grâce aux mots dans le dictionnaire D, les probabilités empiriques de succession ($P_b(c_{l+1}|c_l)$) de deux caractères de l'alphabet (c_l et c_{l+1}), en nous assurant que $\sum_{c_{l+1}} P_b(c_{l+1}|c_l) = 1$ (Jurafsky et Martin, 2009). Nous obtenons alors la probabilité suivante, pour le mot $M = c_1...c_L$ de longueur L :

$$P_0(M) = P_0(< c_1...c_L >) = P(<) * \prod_{l=0}^{L-1} P_b(c_{l+1}|c_l) * P_b(> |c_L) \tag{4.8}$$

Par ailleurs, nous effectuons un lissage additif (*add-k smoothing*, en anglais) afin d'obtenir des probabilités non nulles pour tous les bigrammes possibles avec l'alphabet. De fait, sans lissage, la probabilité de générer un mot avec un bigramme inconnu (c'est-à-dire, jamais observé dans les mots du dictionnaire D) serait de 0, du fait du produit dans l'équation (4.8). Nous utilisons un lissage de valeur $0,01$ dans ce manuscrit.

Notons par la même occasion que cette stratégie est un premier pas vers le modèle hiérarchique de Mochihashi et al. (2009), où P_0 utilise un modèle n-gramme de caractères (ici, $n = 2$).

`d.mix+2`, la combinaison de `d.mix` et `d.2gram` La dernière stratégie est une combinaison des deux méthodes précédemment définies, `d.mix` et `d.2gram`. Comme le deuxième terme de l'équation (4.6) n'impose pas de restriction particulière sur P_0, nous pouvons notamment utiliser un modèle bigramme. Nous obtenons alors non seulement un modèle reposant sur une meilleure représentation des types de mots, mais profitons également du système de cache existant, grâce à l'indicatrice de `d.mix`.

4.3.3 Apprentissage incrémental

Nous avons effectué une expérience supplémentaire, en vue d'une intégration dans des outils informatiques d'annotations utilisés par les linguistes de terrain. Une fois que le modèle de segmentation a convergé, nous simulons la situation où chaque phrase est présentée à un expert de la langue, afin qu'elle soit manuellement corrigée. L'objectif est d'avoir une supervision progressive, pour améliorer graduellement le modèle, à travers ces annotations.

Contrairement à (Palmer et al., 2009) qui a évalué un cadre d'apprentissage actif pour une tâche de documentation avec deux linguistes, nous nous sommes ici limités à une simulation : nous utilisons simplement le texte de référence comme oracle pour chaque phrase.

Nous considérons trois configurations pour cette expérience, où l'état initial est la sortie du modèle `dpseg` sans supervision, après 20 000 itérations avec les paramètres par défaut :

— `baseline` représente une itération supplémentaire d'échantillonnage de Gibbs, sans aucune correction supplémentaire, afin d'avoir un point de comparaison ;

— `i.regular` effectue une mise à jour du dictionnaire interne (le restaurant) de `dpseg` après chaque phrase, pour simuler la correction manuelle par un expert ;

— `i.2level` effectue également une mise à jour périodique (10 fois sur l'ensemble du corpus, dans nos expériences) du modèle de caractères en utilisant le modèle bigramme présenté à l'équation (4.8).

Par ailleurs, afin d'accélérer la propagation des corrections, le modèle effectue régulièrement des itérations supplémentaires d'échantillonnage de Gibbs sur les phrases restantes. Deux paramètres sont alors introduits : batch contrôle le nombre de phrases entre les échantillonnages supplémentaires et iter indique le nombre d'itérations d'échantillonnage de Gibbs appliquées sur le reste du corpus.

4.4 Résultats expérimentaux

Cette partie présente les résultats obtenus par nos modèles, afin d'observer l'influence des différentes stratégies de supervision faible décrites dans la section précédente. Nous complétons ces expériences par un apprentissage incrémental, où nous avons simulé les corrections progressives de chaque phrase par un expert de la langue, afin de s'approcher d'un cas réel d'utilisation de nos modèles. De plus, grâce à des données segmentées aux niveaux des mots et des morphèmes, nous avons pu comparer les segments identifiés par nos modèles avec ces deux types d'unités.

4.4.1 Comparaison avec d'autres modèles de segmentation

Nous comparons nos approches bayésiennes non paramétriques avec deux modèles de segmentation en sous-mots, couramment employés dans des tâches connexes de TAL :

— Parmi les modèles de tokenisation fréquemment utilisés, nous choisissons SentencePiece [2] (Kudo, 2018), basé sur un modèle de langue unigramme, qui a obtenu de meilleures performances que BPE (Sennrich et al., 2016) dans nos expériences préliminaires.

— En nous inspirant de la segmentation en morphèmes, présentée en section 2.2.8, nous pouvons transposer le lien mots-morphèmes vers celui entre les phrases et les mots. Nous utilisons alors Morfessor 2.0 [3] (Creutz et Lagus, 2002, 2007; Smit et al., 2014), afin de segmenter les phrases, considérées comme de longs mots.

Paramétrage des modèles Ces deux modèles nécessitent chacun un paramètre à spécifier ; nous faisons le choix de les calibrer en donnant leur vraie valeur, en nous appuyant sur le corpus de référence. Ce faisant, nous obtenons des performances raisonnables sans avoir à explorer l'ensemble des valeurs. Pour SentencePiece, la taille finale du vocabulaire (vocab_size), paramètre à spécifier obligatoirement, correspond au vrai nombre de types dans le texte. Pour Morfessor, nous estimons empiriquement sur le corpus la longueur moyenne des occurrences de mots pour l'assigner au paramètre morph-length.

Positionnement vis-à-vis de `dpseg` Comparons tout d'abord ces deux modèles avec `dpseg`, sans supervision (mais avec ré-échantillonage du méta-paramètre α). Les résultats sur le corpus mboshi sont dans les trois premières colonnes du tableau 4.3. Nous y observons que `dpseg` obtient systématiquement les meilleurs F-scores aux trois niveaux, par rapport aux deux autres modèles ;

2. https://github.com/google/sentencepiece.
3. https://github.com/aalto-speech/morfessor.

cette approche bayésienne non paramétrique semble ainsi mieux convenir pour la segmentation en mots de ce corpus.

En revanche, d'un point de vue des statistiques sur les textes, comme nous spécifions les vraies valeurs, la proximité des autres méthodes avec la référence est manifeste. Pour SentencePiece, nous retrouvons bien plus de 5 000 types, comme la taille du vocabulaire a été spécifiée avec la valeur de référence (5 312 types). Il en résulte alors des statistiques globales relativement proches de la réalité, plus de 32K occurrences de mots (contre 30,6K), de 6,93 caractères en moyenne (par rapport à 6,39). Malgré ces similitudes, les métriques nous indiquent la mauvaise qualité de la segmentation avec moins de 20 points de F-score, aussi bien pour les occurrences que pour les types. Notons également que WF est inférieur à LF : ce modèle ne parvient pas même à prédire correctement les unités fréquentes.

Pour Morfessor, les statistiques divergent davantage de la réalité : si la longueur des occurrences (WL) est de 4,31, très proche de 4,19, valeur de référence donnée au modèle, et que leur nombre est proche de la réalité, les types sont trop longs (de plus de deux caractères). Cependant, les performances restent globalement meilleures que SentencePiece, avec plus de 10 points de différence pour BF et WF.

Enfin, bien que les résultats de `dpseg` selon les F-scores surpassent ceux des deux modèles présentés, le nombre de types identifiés est le plus bas parmi les trois. En particulier, nous pouvons remarquer que par rapport à la référence, la segmentation génère moins de la moitié du nombre de types, en dépit d'un score LF supérieur.

Nous avons ainsi comparé trois approches de segmentation : SentencePiece obtient les pires scores BF et WF, bien qu'il identifie suffisamment de mots. Morfessor génère quant à lui des types trop longs, pénalisant ses scores à ce niveau, bien qu'il ait accès à la longueur moyenne des occurrences de référence. Quoiqu'il tende à sur-segmenter et à ne pas générer assez de types de mots (voir la section 4.2.3), et ce, malgré le ré-échantillonnage du paramètre de concentration, `dpseg` semble finalement être la meilleure approche des trois pour segmenter ce corpus.

4.4.2 Étude des différentes versions de `dpseg`

Nous évaluons désormais les modifications apportées au modèle `dpseg` de base, à travers la supervision faible ou l'utilisation du processus de Pitman-Yor. Le tableau 4.3 présente l'impact des différentes stratégies de supervision faible sur le corpus mboshi, en utilisant, pour rappel, les 200 premières phrases pour constituer les ressources auxiliaires. De plus, nous fixons la valeur du méta-paramètre λ à $0,25$ pour les méthodes recourant au dictionnaire.

Supervision faible Dans le cas d'une supervision par annotations de frontières, nous obtenons des résultats médiocres avec `g.sparse`, qui voit ses trois F-scores dégradés par rapport au modèle `dpseg` sans supervision. En étudiant les longueurs des unités, nous constatons que dans cette situation, le modèle segmente davantage, alors même que certaines frontières sont données. Il ne parvient donc pas à corriger ses performances grâce ces informations supplémentaires ; au contraire, celles-ci semblent le conforter dans sa tendance à segmenter outre mesure.

Néanmoins, avec la supervision `g.dense`, nous observons bien une amélioration aux trois niveaux, dont une hausse de 7 points pour le LF. Ce modèle réussit aussi à identifier des segments légèrement plus longs en général, menant à un gain notable en nombre de types générés. Nous remarquons ainsi l'importance des informations de non-frontières, afin d'empêcher les phénomènes de sur-segmentation.

modèle	SP	Morf.	dpseg						
super.	/	/	base.	g.sparse	g.dense	d.count	d.mix	d.2gram	d.mix+2
BP	42,7	55,8	61,8	58,8	64,8	62,1	64,5	73,6	**75,6**
BR	46,6	53,8	70,7	72,7	**73,1**	71,1	72,6	57,7	59,2
BF	44,6	54,8	65,9	65,0	**68,7**	66,3	68,3	64,7	66,4
WP	17,1	29,4	35,6	33,5	40,4	36,0	39,7	40,5	**43,8**
WR	18,4	28,6	39,9	40,0	**44,7**	40,4	43,8	33,2	35,9
WF	17,7	29,0	37,6	36,4	**42,4**	38,1	41,7	36,5	39,4
LP	20,0	21,2	43,8	41,2	**53,1**	43,7	52,8	38,7	43,0
LR	18,9	10,6	16,3	15,0	22,3	16,5	21,7	33,8	**37,4**
LF	19,5	14,2	23,8	22,0	31,4	23,9	30,7	36,1	**40,0**
WL	3,89	4,31	3,74	3,50	3,78	3,73	3,79	5,10	5,11
TL	6,93	8,91	4,61	4,45	4,87	4,60	4,87	6,57	6,60
N_{type}	5031	2663	1980	1938	2237	1999	2181	4636	4620
N_{token}	32901	29685	34204	36562	33810	34264	33755	25063	25015

TABLE 4.3 – Résultats des différents modèles de segmentation, dont dpseg et ses versions faiblement supervisées sur le corpus mboshi (200 phrases de supervision). SP correspond au modèle SentencePiece et Morf., à Morfessor. Les meilleurs scores par métrique sont en **gras**.

Nous ne présentons pas les résultats de la stratégie g.random, qui a obtenu de moins bons scores que son équivalent dense, comme attendu. En effet, même si le nombre de frontières et de non frontières est strictement égal, le fait qu'ils soient répartis sur l'ensemble du texte ne permet pas d'en tirer profit aussi avantageusement : leur concentration sur quelques phrases permet spécifiquement de garantir l'existence des mots issus des phrases de supervision dans le dictionnaire du modèle.

Quant à la supervision par dictionnaire, ces méthodes permettent toutes d'améliorer au moins une des métriques d'évaluation. Dans le cas d.count, cette approche intuitive et naïve obtient des résultats proches mais supérieurs à la version de base ; les effets sont donc positifs mais relativement minimes. En comparaison, la stratégie d.mix parvient à améliorer significativement les performances, avec notamment les meilleurs scores aux niveaux des frontières et des occurrences de cette catégorie. Agir sur la distribution de base semble alors un moyen d'intégration plus effectif des informations de supervision. Nous avons, par ailleurs, observé que la supervision par un dictionnaire permet d'augmenter considérablement la probabilité d'identification de ces mots : 96 % des mots du lexique sont correctement retrouvés dans le texte segmenté par dpseg avec la stratégie d.mix+2, à comparer avec un taux de 44 % dans le cas sans supervision.

L'utilisation d'un modèle de caractères adapté à la langue, avec d.2gram, impacte principalement le score des types, où l'on observe une différence de plus de 12 points. Nous notons également un allongement conséquent dans la longueur des types de mots, de presque deux caractères. Les unités, en types, sont alors plus proches (et, en réalité, légèrement plus longs) des longueurs de référence, mais les occurrences se trouvent trop impactées et ont un caractère de trop en moyenne (5,10 pour d.2gram contre 4,19 dans le corpus). Enfin, la combinaison des deux méthodes, d.mix et d.2gram, obtient le plus haut score LF, grâce à la complémentarité des deux approches : d.mix permet une meilleure prédiction des frontières et des occurrences, là où d.2gram améliore principalement la modélisation des types.

modèle	dpseg				pypseg			
super.	base.	g.dense	d.count	d.mix+2	base.	g.dense	d.count	d.mix+2
BP	61,8	64,8	62,1	**75,6**	62,3	65,2	61,9	75,5
BR	70,7	**73,1**	71,1	59,2	70,5	72,8	69,5	58,4
BF	65,9	68,7	66,3	66,4	66,2	**68,8**	65,5	65,8
WP	35,6	40,4	36,0	**43,8**	36,1	40,6	35,8	43,2
WR	39,9	**44,7**	40,4	35,9	40,0	44,5	39,5	35,0
WF	37,6	42,4	38,1	39,4	37,9	**42,5**	37,6	38,7
LP	43,8	**53,1**	43,7	43,0	43,7	52,1	41,4	42,3
LR	16,3	22,3	16,5	37,4	17,0	22,7	16,9	**37,7**
LF	23,8	31,4	23,9	**40,0**	24,5	31,6	24,0	39,9
WL	3,74	3,78	3,73	5,11	3,77	3,82	3,80	5,16
TL	4,61	4,87	4,60	6,60	4,65	4,89	4,79	6,62
N_{type}	1980	2237	1999	4620	2063	2310	2163	4741
N_{token}	34204	33810	34264	25015	33905	33514	33691	24782

TABLE 4.4 – Comparaison des résultats des modèles dpseg et pypseg avec différents types de supervision faible sur le corpus **mboshi** (200 phrases de supervision). Les meilleurs scores par métrique sont en **gras**.

Processus de Dirichlet ou de Pitman-Yor L'utilisation des processus de Pitman-Yor, dont les processus de Dirichlet sont un cas particulier, semble légèrement bénéfique dans le cas sans supervision, comme l'indiquent les scores sensiblement plus élevés de pypseg aux trois niveaux d'évaluation. De plus, nous observons que davantage de types de mots sont identifiés, conformément à ce que la théorie postule. Nous pouvons mettre en parallèle la génération de ces mots avec l'amélioration du score LF de 0,7 point, plus notable qu'aux deux autres niveaux.

En revanche, dès lors que la supervision faible intervient, les bénéfices des processus de Pitman-Yor deviennent encore plus nuancés. L'écart se réduit grandement pour la stratégie g.dense pour devenir négligeable (une différence d'un dixième de point aux trois niveaux), là où les scores des modèles s'appuyant sur une liste de mots sont plus bas que pour leurs équivalents dpseg. Il semblerait donc que l'effet positif des processus sur la qualité de segmentation soit contrebalancé par l'utilisation de la supervision faible. Nous remarquons, en revanche, que l'accroissement du nombre de types est maintenu, avec une centaine de mots uniques supplémentaires découverts en moyenne. L'avantage du processus de Pitman-Yor sur ce plan semble donc se traduire ici par une identification de mots erronés.

Cette expérience nous montre donc, d'une part, que l'utilisation des processus de Pitman-Yor avec supervision ne semble pas strictement supérieurs aux processus de Dirichlet dans le cadre de la segmentation en mots de langues très peu dotées. D'autre part, l'allongement des segments ou le nombre de types générés ne sont pas des indicateurs déterminants dans l'amélioration de la qualité de la segmentation, car les mots rares restent toujours difficiles à identifier correctement.

Résultats en japhug Le tableau 4.5 présente les résultats cette fois sur le corpus japhug, avec la même méthodologie expérimentale. Seules les configurations les plus efficaces sont reportées pour la supervision faible : la stratégie g.dense pour l'échantillonnage de Gibbs semi-supervisé ainsi que d.mix+2 en présence d'un lexique de mots.

modèle	SP	Morf.	dpseg			pypseg	
super.	/	/	base.	g.dense	d.mix+2	base.	d.mix+2
BP	59,7	53,4	61,1	63,7	76,6	61,4	**76,7**
BR	59,8	74,9	90,2	**91,2**	81,1	90,1	80,2
BF	59,7	62,4	72,9	75,0	**78,8**	73,0	78,4
WP	30,3	31,1	39,0	43,5	**54,4**	39,4	54,0
WR	30,4	42,1	55,2	**59,9**	57,2	55,5	56,2
WF	30,3	35,8	45,7	50,4	**55,8**	46,1	55,0
LP	20,5	20,1	39,9	**50,8**	49,9	40,9	49,3
LR	19,5	8,0	13,4	19,7	37,3	13,9	**37,5**
LF	20,0	11,4	20,1	28,3	**42,7**	20,8	42,6
WL	4,72	3,50	3,34	3,44	4,50	3,36	4,55
TL	6,71	9,72	4,21	4,67	6,19	4,25	6,20
N_{type}	6413	2688	2258	2610	5041	2295	5124
N_{token}	28,6k	38,6k	40,5k	39,3k	30,0k	40,2k	29,7k

TABLE 4.5 – Résultats des différents modèles de segmentation en mots sur le corpus **japhug** (200 phrases de supervision). SP correspond au modèle SentencePiece et Morf., à Morfessor. Les meilleurs scores par métrique sont en **gras**.

Nous y observons des tendances similaires au mboshi : le modèle de base dpseg reste plus approprié que SentencePiece ou Morfessor et l'utilisation de la supervision faible permet d'améliorer les performances de manière notable, en particulier avec un dictionnaire. Sur ce corpus, notons également que l'utilisation de cette supervision est systématiquement meilleure que g.dense, aux trois niveaux d'évaluation. De plus, en comparaison avec sa variante pypseg, le bénéfice de cette version plus générale du processus s'estompe avec l'usage d'une meilleure distribution de base dans d.mix+2.

4.4.3 Analyse de la segmentation

Nous illustrons l'effet du modèle de segmentation dpseg ainsi que de la supervision faible, par deux phrases d'exemple en figure 4.4. La version supervisée correspond à la méthode d.mix+2.

	exemple 1					exemple 2				
référence	**obengi**	amipasa	koo	sa	kω	**bana**	ba	adi	otɛɛ	imbva
sans supervision	**obengi**a	mipasa	koo	sakω		**bana**ba		adio	tɛɛ	imbva
avec supervision	**obengi**	amipasa	koo	sakω		**bana**ba		adi	otɛɛimbva	

FIGURE 4.4 – Deux exemples de segmentations obtenues par dpseg avec et sans supervision en mboshi. La première phrase signifie « ces enfants ont la même taille », la seconde « le chasseur s'est frayé un chemin dans la forêt ». Les mots en **gras** font référence aux segments commentés ci-dessous.

Tout d'abord, nous pouvons constater dans les deux prédictions de dpseg, des erreurs de différents degrés de gravité. Une frontière de mots non identifiée comme entre les deux derniers (ou premiers) mots de l'exemple 1 (ou 2). Ceux-ci semblent relativement moins pénalisants pour un

annotateur que ceux s'immisçant en milieu de mots. Dans le cas du premier segment du premier exemple, « obengia », nous observons en réalité la conséquence de deux erreurs de frontières : celle entre le « i » et le « a » n'est pas identifiée, tandis que la position entre le « a » et le « m » est incorrectement considérée comme frontière. Ce type d'erreur (survenant aussi dans l'autre exemple) est plus ennuyeux car il impacte deux unités, devenues inexactes.

L'utilisation de la supervision faible permet de remédier en partie à ce problème. En effet, nous observons dans l'exemple 1, que l'erreur sur le premier segment est corrigée, rectifiant par la même occasion le second mot. Le mot « obengi » est présent dans le dictionnaire de supervision, ce qui aide son identification par le modèle ; la présence de la frontière entre le « i » et le « a » semble avoir suffisamment baissé la probabilité d'avoir une frontière à la position suivante.

Cependant, toutes les erreurs ne sont pas résolues par la supervision faible, comme dans le second exemple, où le modèle ne parvient pas à segmenter davantage « banaba », bien que « bana », le premier mot de référence, soit présent dans le dictionnaire. Nous attribuons cela au phénomène de cooccurence : ces deux mots « bana » et « ba » apparaissent de fait fréquemment ensemble, ce qui induit le modèle à considérer cette chaîne de caractères comme une seule unité. Ce phénomène est dû à l'hypothèse d'indépendance entre les segments : il s'agit là de la limite principale du modèle unigramme choisi pour `dpseg`, comme souligné par Goldwater et al. (2009).

texte	référence		`dpseg`		`g.dense`		`d.mix+2`	
rang	mot	fréq.	mot	fréq.	mot	fréq.	mot	fréq.
1	**wa**	1592	a	1090	a	944	**wa**	451
2	**la**	1139	o	774	**la**	752	a	423
3	**nga**	1102	**la**	758	**nga**	736	**la**	294
4	ya	835	**nga**	639	o	698	mo	265
5	m	581	**wa**	546	**wa**	591	**nga**	253
6	adi	579	i	541	i	491	o	247
7	l	545	baa	429	e	400	**kaa**	244
8	mo	410	e	419	baa	378	baa	233
9	**kaa**	385	**kaa**	366	**kaa**	363	i	192
10	nω	373	ma	364	ma	350	e	187

TABLE 4.6 – Les dix mots les plus fréquents dans le corpus mboshi (5K) et ses versions segmentées par `dpseg` sans supervision et avec les stratégies `g.dense` et `d.mix+2`. Les unités en commun aux quatre textes sont en **gras**.

Le tableau 4.6 présente les dix mots les plus fréquents en mboshi dans le texte de référence et les segmentations obtenues avec `dpseg` et ses variantes supervisées. Remarquons tout d'abord les disparités dans les distributions des mots dans le corpus mboshi et les segmentations. En particulier, dans le texte de référence, l'unité la plus fréquente cumule près de 1 600 occurrences ; pour les méthodes automatiques, ce nombre se situe entre 1 100 et 450, soit entre 1,5 et 3,5 fois moins. De plus, l'utilisation de la supervision faible ne modifie pas en profondeur les unités fréquentes identifiées : `g.dense` génère les mêmes segments que la version de base et la seule différence de `d.mix+2` est l'unité « mo », un mot également présent dans la référence. Cette seconde supervision est également caractérisée par l'aplatissement de la courbe de fréquence, avec une différence de seulement 264 entre la première et la dixième occurrence contre 1 219 dans la référence. En outre, les trois segmentations automatiques identifient des segments en moyenne trop courts : 4 parmi

les 10 plus fréquents ne contiennent qu'une seule lettre, contre un seul pour la référence, ce qui souligne la tendance de `dpseg` à sur-segmenter.

4.4.4 Apprentissage incrémental

Pour cette expérience, nous évaluons les trois configurations, présentées en section 4.3.3, par leur taux d'erreur moyen, calculé sur des plages de 100 phrases, comme illustré par l'équation (4.9). Une erreur correspond simplement à une incohérence de la valeur des frontières b_i et la longueur est mesurée en nombre de caractères.

$$\text{taux d'erreur} = \frac{\text{nombre d'erreurs sur 100 phrases}}{\text{longueur des 100 phrases}} \tag{4.9}$$

Ici, le texte est mélangé aléatoirement afin de lisser la difficulté sur tout le corpus, une approche qui s'est également révélée meilleure que de donner les phrases dans l'ordre, selon (Palmer et al., 2009). Par ailleurs, nous effectuons 50 itérations d'échantillonnage de Gibbs supplémentaires (iter), toutes les 100 phrases (batch), afin de propager plus rapidement les corrections sur le reste du texte.

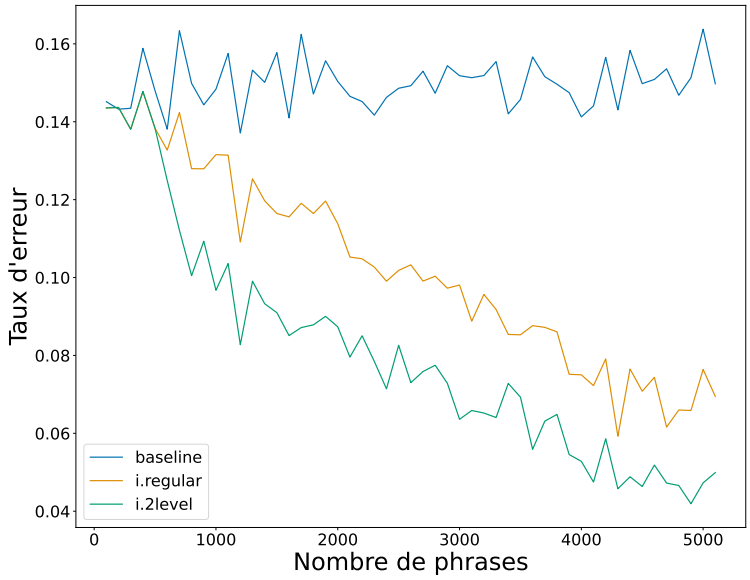


FIGURE 4.5 – Taux d'erreur en moyenne pour 100 phrases sur le corpus mboshi (5K) avec un apprentissage incrémental (batch $= 100$, iter $= 50$).

Nous constatons tout d'abord, sans surprise, que le modèle de base `dpseg` obtient en moyenne un taux d'erreur constant, oscillant autour de 0,15 pour le mboshi ; la segmentation reste stable, puisqu'elle est calculée par un modèle convergé, sans correction. Ensuite, les mises à jour régulières du modèle permettent d'obtenir une baisse nette du taux d'erreur pour atteindre 0,07. Dans cette situation simulée, le nombre de corrections que devrait faire un annotateur est donc divisé par deux entre le début la fin de l'annotation.

Enfin, l'utilisation d'un meilleur modèle de caractères, dans `i.2level`, permet de réduire davantage ce taux, vers 0,05 pour les dernières phrases du corpus. La chute initiale du score, aux alentours de 500 phrases, correspond aux premières modifications du P_0 par le modèle bigramme, effectuées 10 fois au total. L'écart par rapport à la configuration `i.regular` se maintient tout au

long du texte, indiquant l'importance d'une distribution de base plus pertinente dans la qualité des segmentations. Nous pouvons également noter que les améliorations ultérieures du modèle de caractères sont plus modestes ; son utilité se situe donc principalement en début d'annotation pour réduire significativement les erreurs. Autrement dit, ces 500 phrases suffisent à la création d'un modèle bigramme stable et robuste pour P_0.

4.4.5 Des mots aux morphèmes

La dernière expérience que nous avons considérée vise à mieux caractériser les unités qui sont identifiées par `dpseg` et ses variantes. En effet, comme nous l'avons noté en section 4.2.3, le modèle tend à effectuer une sur-segmentation des phrases. Nous aboutissons alors à des segments qui sont trop courts pour être des mots.

Si la présence de frontières au sein d'un mot n'est pas souhaitable pour notre tâche, la gravité de l'erreur diffère selon que la séparation se situe entre des morphèmes (comme « mot s ») ou au milieu d'un morphème (comme « m ots »). Nous analysons donc ici la nature des unités segmentées par `dpseg`.

Comme nous disposons pour le japhug d'un corpus segmenté en mots mais aussi en morphèmes, nous comparons les unités obtenues par nos modèles en fonction de ces deux niveaux de segmentation. Nous avons également recours à la supervision faible par un dictionnaire de mots ou de morphèmes, issu des 200 premières phrases, afin d'étudier leur influence sur les segments générés par `dpseg`. Le tableau 4.7 présente les résultats pour le japhug.

référence	mot			morphème		
supervision	/	mot	morph.	/	mot	morph.
BP	61,1	76,6	70,3	87,6	93,3	93,2
BR	90,2	81,1	83,0	75,0	57,3	63,8
BF	72,9	78,8	76,1	80,8	71,0	75,8
WP	39,0	54,4	47,5	58,9	50,1	54,3
WR	55,2	57,2	55,0	51,1	32,25	38,5
WF	45,7	55,8	51,0	54,7	39,2	45,1
LP	39,9	49,9	43,5	45,5	25,85	36,55
LR	13,4	37,3	26,3	37,6	47,7	54,6
LF	20,1	42,7	32,8	41,2	33,5	43,8
WL	3,34	4,50	4,09	3,34	4,50	4,09
TL	4,21	6,19	5,43	4,21	6,19	5,43
N_{type}	2258	5041	4077	2258	5041	4077
N_{token}	40,5k	30,0k	33,1k	40,5k	30,0k	33,1k

TABLE 4.7 – Comparaison des résultats sur le texte japhug pour un texte segmenté en mots ou en morphème (référence), avec ou sans supervision par un dictionnaire (supervision) de mots ou de morphèmes.

Nous remarquons que sans supervision, le texte segmenté par `dpseg` obtient de meilleurs scores d'évaluation, à tous les niveaux, lorsqu'il est comparé à la référence segmentée au niveau des morphèmes. Le phénomène de sur-segmentation observé semble donc mener à la génération

d'unités à mi-chemin entre mots et morphèmes, plus proches de ces derniers, comme en témoigne la différence de plus de 20 points de LF.

L'utilisation du dictionnaire de mots permet toutefois d'inverser cette tendance naturelle. Les performances sont améliorées au niveau des mots (par exemple, +10 points de WF), au détriment des scores pour les morphèmes (-15 points de WF) : les unités obtenues par le modèle se rapprochent ainsi des mots, tels qu'ils sont définis dans la référence.

La supervision faible à l'aide d'une liste de morphèmes, en revanche, ne semble pas apporter d'amélioration nette : elle dégrade les F-scores des frontières et des occurrences de manière non négligeable, le bénéfice pour le LF étant minime (+2 points). Quant au niveau des mots, nous observons une amélioration notable des trois F-scores ; en effet, certains mots sont des morphèmes, d'où cette influence positive au niveau des mots. Notons enfin que la longueur des unités se retrouve impactée tout de même : les unités sont plus longues avec la supervision par une liste de morphèmes que sans.

4.5 Conclusion

Dans les projets de documentation de langues, notamment à travers les travaux déjà entrepris par les linguistes de terrain, il est sous-optimal de considérer des méthodes de segmentation en mots exclusivement sans supervision ; il existe, en effet, des corpus déjà annotés ou des lexiques sur la langue étudiée (Bird, 2020). Ce chapitre étudie donc les différentes méthodes pour les incorporer, afin d'améliorer la qualité des segmentations par rapport aux configurations intégralement non supervisées, qui constituent les approches standards dans le domaine jusque là.

Pour ce faire, nous nous sommes tout d'abord intéressés au modèle bayésien non paramétrique non supervisé `dpseg`, reposant sur les processus de Dirichlet. Il semble notamment bien adapté pour la segmentation en mots de langues très peu dotées, selon les travaux antérieurs. Ce modèle, bien que robuste, présente principalement deux biais : il tend à sur-segmenter les unités (surtout dans sa version unigramme) et ne génère pas suffisamment de types, en particulier de types rares. Nous l'avons réimplémenté en Python et avons créé une variante basée sur les processus de Pitman-Yor, qui reproduisent mieux la distribution en loi de puissance des mots dans les langues naturelles.

Nous l'avons ensuite modifié afin de pouvoir intégrer ces ressources auxiliaires, disponibles dans des conditions réalistes, à travers une supervision faible. La première catégorie de méthodes intervient lorsque des indications sont disponibles sur les emplacements des frontières et des non frontières. La seconde s'efforce d'augmenter la probabilité d'identifier les mots de la langue en s'aidant d'une liste de mots. Rappelons toutefois que le dictionnaire que nous utilisons est issu du corpus à segmenter et est constitué de formes, ce qui est un scénario plus favorable qu'un lexique de lemmes provenant éventuellement d'un autre corpus, un scénario plus réaliste.

Nos expériences sur deux langues en cours de documentation montrent l'efficacité de certaines stratégies de supervision faible, aux effets variables, par rapport à la situation sans supervision. L'utilisation d'annotations denses, à savoir de quelques phrases entièrement segmentées, permet d'améliorer notablement la segmentation, par rapport à une méthode ne donnant que certaines frontières de mots sur l'ensemble du texte, soulignant l'importance de l'information d'absence de frontière. En utilisant un dictionnaire, l'approche alliant un meilleur modèle de langue avec une hausse de probabilité pour les mots connus atteint les meilleurs scores en général.

En ce qui concerne l'utilisation des processus de Pitman-Yor, généralisation des processus de Dirichlet, malgré des résultats sensiblement meilleurs dans le cas de base, nous n'avons pas

observé de différence notable, en particulier lorsqu'une supervision faible est employée. Il semblerait donc que cette dernière comble en grande partie les gains de segmentation dus à la meilleure reproduction de la distribution en loi de puissance.

En outre, comme nos modèles ont pour finalité, à terme, une utilisation dans des conditions réelles de documentation par des linguistes et annotateurs, nous avons simulé une approche d'apprentissage incrémental, où un expert corrigerait des phrases manuellement, qui seraient progressivement intégrées dans le modèle. Nous avons évalué deux configurations, l'une avec de simples actualisations tenant compte des corrections de chaque phrase, et l'autre effectuant également quelques mises à jour du modèle de caractères pour les types de mots. Nous avons constaté une baisse notable du taux d'erreurs tout au long du texte et l'utilisation d'un meilleur modèle de langue a permis d'améliorer davantage les performances, très tôt dans la simulation. Ces résultats sont donc prometteurs quant à l'intégration de notre modèle dans une plateforme d'annotation linguistique.

Enfin, nous avons comparé les segmentations obtenues par `dpseg` et ses variantes faiblement supervisées, avec un texte de référence segmenté en mots et morphèmes. Nous avons remarqué que les unités identifiées sans supervision semblent être à mi-chemin entre ces deux catégories d'unités. Si le recours à la supervision faible par un dictionnaire de mots permet de guider le modèle vers l'identification de ces unités, l'utilisation d'une liste de morphèmes obtient des performances plus mitigées. Cette nature hybride des unités identifiées par `dpseg`, due à la sur-segmentation inhérente au modèle, ouvre alors la voie au chapitre suivant, où nous effectuons une segmentation à deux niveaux dans le but d'atténuer ce phénomène.

Chapitre 5

La segmentation à deux niveaux

Ah ! lève-toi, soleil !
fais pâlir les étoiles,
qui, dans l'azur sans voiles,
brillent au firmament.

Acte II, *Roméo et Juliette*, Gounod

5.1 Introduction

Nous avons observé dans le chapitre précédent que les unités identifiées par dpseg se situent à mi-chemin entre les mots et les morphèmes, trop courts par rapport aux premiers, mais trop longs pour les seconds. Comme il a été constaté par Johnson (2008a) pour des modèles plus complexes, l'incorporation de la structure des unités apporte des améliorations notables. Nous étudions donc ici l'impact obtenu en introduisant un second niveau de segmentation, qui vise à capturer la notion de morphèmes, les plus petites unités significatives dans la langue. En effet, en effectuant une segmentation à deux niveaux [1], le but est d'observer une différenciation de ces deux unités, menant, idéalement, à une meilleure segmentation des mots. De fait, dans l'état actuel, les frontières de mots identifiées par dpseg contiennent entre autres des frontières de morphèmes, un type d'erreur bien plus justifiable qu'au milieu d'un morphème, selon le contexte.

Entrée **S1**	Phrase non segmentée	kʏndʑixtʏɣχsɯmpjʏtunɯ
Sortie **S3**	Segmentation en mots et morphèmes	kʏndʑi–xtʏɣ χsɯm pjʏ–tu–nɯ

FIGURE 5.1 – La tâche de segmentation jointe en mots et morphèmes pour l'exemple 2.1.

La figure 5.1 illustre notre tâche avec la phrase d'exemple (figure 2.1) : l'objectif est de segmenter simultanément en mots (séparés par des espace, « ») et en morphèmes (séparés par des tirets, « – », à l'intérieur des mots), en distinguant correctement la nature des frontières, telles qu'elles sont définies par les linguistes. Il s'agit en ce sens d'une tâche plus complexe que la segmentation en mots, étudiée précédemment. Cela s'apparente à une combinaison avec la tâche connexe de segmentation en morphèmes, généralement effectuée au niveau des mots et non des phrases. Notons, par ailleurs, que nous réalisons là une segmentation de surface et non canonique (Cotterell et al., 2016), comme nous devons avoir une correspondance exacte des caractères entre les versions non segmentées et annotées (voir section 2.2.8).

1. Cette « segmentation à deux niveaux » n'est pas liée à la « morphologie à deux niveaux » (Koskenniemi, 1983), qui décrit la relation entre les formes de surface et canonique.

Il faut toutefois souligner qu'il s'agit d'une tâche artificielle : dans le cadre de la documentation des langues, les linguistes ne procèdent pas de cette manière. Une phrase est tout d'abord segmentée en mots, puis, une fois la segmentation assurée, en morphèmes. Ainsi, nous n'essayons pas de concevoir ici un outil directement intégrable dans les outils d'annotation, afin de remplacer les travaux du chapitre précédent : bien que nous cherchons à effectuer une segmentation réussie aux deux niveaux, notre intention est également d'évaluer l'influence de ce niveau de segmentation supplémentaire dans la nature des unités identifiées par les modèles bayésiens non paramétriques. Ce chapitre s'intéresse, plus généralement, à déterminer s'il est possible de différencier les mots des morphèmes sur la seule base d'éléments statistiques.

Nous présentons les contributions suivantes :

— la conception de plusieurs modèles de segmentation à deux niveaux, à partir de `dpseg` ;
— le recours aux ressources auxiliaires disponibles, comme dans le chapitre précédent, afin d'améliorer ces modèles par une supervision faible ;
— l'analyse des hypothèses sous-jacentes aux méthodes bayésiennes non paramétriques au sein des corpus de documentation de langues.

La section 5.2 décrit les différents modèles conçus ou évalués pour la segmentation jointe en mots et morphèmes ainsi que les méthodes de supervision éventuellement utilisées. Les résultats obtenus sont présentés en section 5.3, en comparant les approches mises en place. Enfin, la section 5.4 étudie les distributions des unités dans les corpus, mises en perspective par rapport à nos méthodes statistiques. Ce chapitre reprend des éléments présentés dans (Okabe et Yvon, 2022a) et (Okabe et Yvon, 2023a).

5.2 Modèles de segmentation à deux niveaux

Pour concevoir des modèles de segmentation à deux niveaux, nous nous basons sur `dpseg`, plutôt que `pypseg` (ou d'autres modèles de segmentation), d'après les conclusions du chapitre précédent. Nous reprenons donc ici les mêmes notations, présentées en section 2.2.3. Cette partie présente les trois catégories d'extensions, que nous avons implémentées en Python dans `pyseg`, ainsi que deux configurations de l'état de l'art en segmentation. Enfin, nous considérons également l'utilisation de la supervision faible pour améliorer les performances.

5.2.1 Modèle de base, en cascade

Une première méthode naïve est d'aborder la tâche en cascade (modèle `pipeline`), pour passer (1) de la phrase aux mots, puis (2) des mots aux morphèmes. Cette approche consiste alors en la combinaison (1) de la tâche standard de segmentation des mots dans un premier temps, puis (2) de la tâche de segmentation en morphèmes des *types* de mots dans un second temps.

Le défi pour ce modèle en cascade réside dans les répercussions d'erreurs entre les deux étapes : toute segmentation erronée au niveau des mots affectera aussi les segmentations des morphèmes, augmentant les possibilités d'erreurs. Par ailleurs, la conception de ce modèle implique qu'un même mot est nécessairement segmenté en morphèmes d'une seule manière, la seconde étape segmentant les *types* et non les *occurrences* des mots identifiés en premier lieu. Dans nos expériences pour deux langues, cela ne pose pas de problème majeur ; seuls 51 mots en japhug et 14 en tsez possèdent plusieurs segmentations morphologiques.

5.2.2 Modèles couplés en parallèle

Comme nous disposons d'un modèle de segmentation en unités et que nous souhaitons segmenter en deux niveaux, une méthode simple est d'employer en parallèle un modèle `dpseg` pour les mots et un autre pour les morphèmes.

niveau	χ	p	ɯ	n	ɯ	p	ɯ
mot (b_i^w)	0	0	0	1	0	0	1
morphème (b_i^m)	0	0	0	1	1	0	1

FIGURE 5.2 – Exemple de phrase en japhug (χpɯn ɯ–pɯ, « petit moine », en français), représentée aux deux niveaux de segmentation.

Pour assurer la cohérence des segmentations lorsque les prédictions pour chaque niveau sont réunies, les modèles suivent une règle, traduisant la même réalité de deux manières : soit i) une frontière de mots est nécessairement une frontière de morphèmes, soit ii) une non frontière de morphèmes est également une non frontière de mots, comme nous pouvons l'observer dans la figure 5.2, respectivement pour les caractères « n » et « p ».

Dans le premier cas i), représenté par le modèle `parallel-w`, l'échantillonnage de Gibbs concerne tout d'abord les mots. Si une frontière de mots est identifiée, cela induit de manière déterministe la présence d'une frontière de morphèmes. De ce fait, lors de l'échantillonnage d'une frontière pour le caractère « n », si le modèle y identifie une frontière de mots ($b_i^w = 1$), alors la valeur de la frontière de morphème est fixée à 1 sans retirage. Tandis que le niveau des mots correspond au modèle `dpseg` standard, les unités segmentées en morphèmes sont mécaniquement plus fréquentes, car un échantillonage supplémentaire est effectué pour chaque position interne d'un mot. Ce second niveau aboutit donc à des segments plus courts, idéalement proches des morphèmes.

À l'inverse, dans le second cas ii), le modèle que nous nommerons `parallel-m`, échantillonne en premier les frontières de morphèmes. Une absence de frontière de morphèmes ($b_i^m = 0$), comme pour « p », signifie qu'il ne peut pas y avoir de frontière de mots non plus ; la valeur $b_i^w = 0$ est affectée directement pour les mots. Ici, le niveau des morphèmes correspond à la sortie du modèle `dpseg` de base ; les unités segmentées en mots sont alors plus rares, étant donné que l'échantillonnage à ce niveau ne se réalise que pour les frontières identifiées de morphèmes. Les segments seront donc plus longs que ce que `dpseg` génère habituellement à un niveau, se rapprochant plutôt des mots. Les algorithmes 5.1 et 5.2 récapitulent les choix d'échantillonnage pour les modèles `parallel-w` et `parallel-m`.

Algorithme 5.1 : `parallel-w`

pour $i = 1 \dots I$ **faire**
  Tirer la valeur de b_i^w (`dpseg`);
  si $b_i^w = 1$ **alors**
    $b_i^m = 1$;
  sinon `/*` $b_i^w = 0$ `*/`
    Tirer la valeur de b_i^m;
  fin
fin

Algorithme 5.2 : `parallel-m`

pour $i = 1 \dots I$ **faire**
  Tirer la valeur de b_i^m (`dpseg`);
  si $b_i^m = 0$ **alors**
    $b_i^w = 0$;
  sinon `/*` $b_i^m = 1$ `*/`
    Tirer la valeur de b_i^w;
  fin
fin

5.2.3 Modèles hiérarchiques

Sur le modèle de Mochihashi et al. (2009), présenté à la section 2.2.4, nous avons également conçu un modèle hiérarchique en intégrant, au sein du modèle, la structure même des mots, constitués d'un ou plusieurs morphèmes. Au niveau des mots, nous conservons le modèle de base `dpseg` en changeant uniquement la distribution de base P_0. La probabilité d'un mot $\widetilde{P}_0$ [2] se calcule alors en combinant le modèle de longueur et le produit de la probabilité de chacun de ses morphèmes constitutifs, P_m, comme explicité par l'équation (5.1).

$$\widetilde{P}_0(w = m_1...m_K|h^-) = p_\#(1-p_\#)^{(L-1)} * \prod_{k=1}^{K} P_m(m_k|h^-) \tag{5.1}$$

Nous utilisons le modèle `dpseg` standard pour obtenir la probabilité d'un morphème P_m, définie par l'équation (5.2), qui est une adaptation de l'équation caractéristique de `dpseg` (voir section 2.2.3, équation (2.5)), avec des notations propres à ce niveau de segmentation. Nous introduisons ainsi le paramètre de concentration α_m et la distribution de base Q_0, spécifiques au modèle de morphèmes. Q_0 consiste en un modèle de longueur et un modèle de caractères classiques, et donc identique à l'équation (2.3).

$$P_m(m_k|h^-;\alpha_m) = \frac{n_{m_k}^{(h^-)} + \alpha_m Q_0(m_k)}{n_{\bar{m}} + \alpha_m} \tag{5.2}$$

Ces modifications interviennent lorsqu'un nouveau mot est considéré (à travers P_0) : sa probabilité est alors obtenue via sa segmentation selon le modèle de morphèmes. Si ce mot est retenu comme hypothèse lors de l'échantillonnage, sa décomposition en morphème est également sauvegardée pour les prochaines segmentations. Notons ici que l'analyse morphologique concerne les *types* de mots ; les expériences préliminaires menées sur les *occurrences* n'ont pas été concluantes. Ce choix signifie également que le modèle ne recalcule pas la segmentation d'un mot connu.

Le modèle que nous venons de décrire sera dénommé `hier-type`. Deux variantes supplémentaires ont été implémentées afin d'améliorer les segmentations au vu des résultats préliminaires. En employant le procédé `pipeline`, il est possible d'obtenir une segmentation en morphèmes des types de mots identifiés par le modèle. L'idée est de favoriser davantage la cohérence des segmentations en morphèmes et, par là, se rapprocher de la distribution en loi de puissance pour ces unités. Nous avons donc `hier-final`, si l'échantillonnage avec `pipeline` se fait uniquement une fois toutes les itérations en mots terminées (iter_f itérations d'échantillonnage de Gibbs pour les morphèmes). La seconde variante, `hier-iter`, effectue régulièrement cette analyse supplémentaire au cours des itérations (iter_r itérations d'échantillonnage de Gibbs pour les morphèmes, toutes les batch_m itérations pour les mots).

5.2.4 *Adaptor Grammar*

Les *Adaptor Grammars* (Johnson et al., 2007), présentées en section 2.2.5, sont également des modèles non supervisés et hiérarchiques, capables de représenter des structures imbriquées complexes. Ils ont déjà été utilisés pour la segmentation en mots dans le cadre de la documentation des langues pour leur haute performance (Godard et al., 2018b).

Nous utilisons la grammaire `colloc` où les Phrases sont constituées de Locutions (*Collocation*), elles-mêmes composées de Mots, à savoir des séquences de Caractères (les entités sont

2. Le tilde indique la distribution de base des mots dans les modèles hiérarchiques.

notées en majuscule). En suivant le constat de Johnson (2008a), nous considérons la segmentation en Locutions comme identifiant des frontières de mots et la segmentation en Mots pour les frontières de morphèmes, lors de l'évaluation.

En pratique, nous utilisons l'implantation originelle de l'AG [3]. Pour son paramétrage, nous référons aux valeurs définies pour MorphAGram [4] (Eskander et al., 2020), qui obtient de meilleurs résultats pour la tâche de segmentation morphologique que Morfessor (Creutz et Lagus, 2002, 2007; Smit et al., 2014) notamment.

5.2.5 Supervision faible

Les méthodes que nous avons vues dans le chapitre précédent pour la segmentation à un niveau (section 4.3) peuvent être aisément adaptées pour la segmentation à deux niveaux. À la lumière des résultats obtenus par les différentes stratégies, nous choisissons de garder uniquement les meilleures pour les deux types de ressources auxiliaires disponibles : les phrases entièrement segmentées (situation `sentence`) et les dictionnaires (situation `dictionary`). Notons ici que nous supposons des ressources auxiliaires segmentées conjointement en mots et en morphèmes. Nous considérons ici aussi les 200 premières phrases du corpus de chaque langue comme données de supervision.

Situation « `sentence` » Lorsque des phrases entièrement segmentées sont disponibles, la stratégie de supervision faible `g.dense` est naturellement utilisée. Ceci revient donc à donner la valeur de toutes les variables de frontières pour les phrases connues, dans le format illustré par la figure 5.3. Nous remarquons que les informations données se font au niveau des *occurrences*.

niveau	k	ɤ	n	d	ʑ	i	x	t	ɤ	ɣ	χ	s	ɯ	m	p	j	ɤ	t	u	n	ɯ
mot	0	0	0	0	0	**0**	0	0	0	1	0	0	0	1	0	0	**0**	0	**0**	0	1
morphème	0	0	0	0	0	**1**	0	0	0	1	0	0	0	1	0	0	**1**	0	**1**	0	1

FIGURE 5.3 – Représentation des deux niveaux de frontières pour la phrase japhug « kɤndʑi–xtɤɣ χsɯm pjɤ–tu–nɯ » dans la situation « `sentence` ». Les frontières de morphèmes sont en **gras**.

Dans le cas des modèles couplés, l'intégration est directe : la présence ou l'absence de frontière est déterminée de manière déterministe pour toutes les positions données par supervision aux deux niveaux. Pour les modèles hiérarchiques, cette supervision implique que pour les phrases déjà segmentées, la probabilité d'un mot, correctement segmenté, est le produit des probabilités de ses morphèmes, eux aussi corrects.

Enfin, lors de la segmentation des *types* de mots, comme dans `pipeline`, `hier-final` et `hier-iter`, nous donnons également les analyses morphologiques de référence pour chacun d'eux, à travers les annotations denses. Dans l'exemple 5.3, lors de la segmentation des *morphèmes*, les trois mots « kɤndʑi–xtɤɣ », « χsɯm » et « pjɤ–tu–nɯ » seront donnés segmentés pour la supervision faible.

3. https://web.science.mq.edu.au/~mjohnson/code/py-cfg-2013-09-23.tgz.
4. https://github.com/rnd2110/MorphAGram.

Situation « `dictionary` » Pour les cas où un accès à un dictionnaire est possible, nous utilisons la stratégie `d.mix+2` pour l'intégrer. Cette méthode, pour rappel, modifie la distribution de base P_0 vers P_0'', définie par l'équation (5.3).

$$P_0''(w) = \frac{\lambda}{|D|} * \mathbb{1}_{\{w \in D\}} + (1-\lambda) * \underbrace{P_0(w)}_{\text{modèle bigramme}}, \tag{5.3}$$

Il faut noter que notre étude se focalise sur des dictionnaires de mots et de morphèmes cohérents : tous les mots connus sont formés par les morphèmes du dictionnaire ; en effet, ce dernier est constitué en utilisant les segmentations de référence des mots. Par ailleurs, cette méthode implique une supervision par les *types*, car les fréquences des unités n'impacte pas le calcul de la distribution de base.

Une particularité de cette supervision dans le modèle hiérarchique est qu'elle affecte différemment les deux niveaux. En effet, si la distribution de base au niveau des morphèmes (Q_0 dans l'équation (5.2)) est modifiée suivant l'équation (5.3), le niveau des mots, défini par $\widetilde{P}_0$ de l'équation (5.1), n'utilise pas le modèle bigramme, mais le produit de la probabilité de ses morphèmes. On a alors l'équation (5.4) suivante :

$$\widetilde{P''}_0(w) = \frac{\lambda}{|D|} * \mathbb{1}_{\{w \in D\}} + (1-\lambda) * \underbrace{p_\#(1-p_\#)^{(L-1)} * \prod_{k=1}^{K} P_m(m_k|h^-)}_{=\text{équation (5.1)}} \tag{5.4}$$

Les informations de supervision impactent donc principalement le premier terme pour le niveau des mots.

5.2.6 Apprentissage supervisé

Nous évaluons également une configuration de supervision classique, en supposant un accès à des phrases entièrement segmentées, ce qui implique une comparaison uniquement avec la stratégie de supervision faible `sentence`.

Pour ce faire, nous employons un champ aléatoire conditionnel (CRF) (Lafferty et al., 2001), présenté en section A.2, avec la méthodologie de Moeller et Hulden (2018) pour la tâche de segmentation en morphèmes. Nous effectuons donc une prédiction au niveau des caractères en utilisant des étiquettes similaires aux étiquettes BIO. La figure 5.4 illustre la configuration utilisée : l'étiquette « B-w » correspond aux frontières de mots après le caractère, là où « B-m » représente les frontières de morphèmes à l'intérieur des mots, enfin, « I » indique que le caractère se trouve au sein d'un morphème (et donc au sein d'un mot).

Phrase originale : χpɯn ɯ–pɯ

χ	p	ɯ	n	ɯ	p	ɯ
B-w	I	I	I	B-w	B-m	I

FIGURE 5.4 – Exemple de phrase en japhug étiquetée pour le CRF (« petit moine », en français).

Pour l'implantation du CRF, nous utilisons Wapiti [5] (Lavergne et al., 2010), avec sa configuration par défaut. Nous choisissons l'algorithme d'optimisation OWL-QN (*Orthant-Wise Limited-memory Quasi-Newton*), (Andrew et Gao, 2007), dû à la taille restreinte des données d'entraînement. Le modèle repose sur des fonctions caractéristiques classiques testant à chaque position,

5. https://wapiti.limsi.fr/.

une fenêtre de cinq caractères (les deux avant et après la position courante) avec l'étiquette correspondante (unigramme) ou combinée avec l'étiquette précédente (bigramme). Les 200 premières phrases sont réservées pour l'entraînement, comme pour la méthode `sentence`, le reste du corpus étant utilisé pour l'évaluation.

5.3 Résultats expérimentaux

Nous présentons dans cette section les résultats obtenus avec les modèles présentés précédemment sur les corpus japhug et tsez. Comme les tendances observées sont similaires, nous reportons principalement les expériences en japhug ; leurs équivalents en tsez sont en annexe A.3. Les statistiques des corpus sont en section 3.2.5.

5.3.1 Configuration expérimentale

Nous reprenons le paramétrage de `dpseg` défini dans le chapitre précédent (section 4.2.2). L'échantillonnage de Gibbs de 20 000 itérations est ici aussi accéléré par un processus de recuit simulé avec 10 incréments de température.

Les modèles `dpseg`, éventuellement modifiés, consacrés à la segmentation en morphèmes utilisent également les mêmes valeurs de méta-paramètres : $\alpha = \alpha_m = 20$. Initialement, les deux paramètres de concentration ont la même valeur, mais lors de la convergence de l'algorithme, elles diffèrent grâce au ré-échantillonnage après chaque itération sur le corpus (voir section 4.2.3).

De plus, les résultats de ce chapitre pour les modèles basés sur `dpseg` sont obtenus par une moyenne des scores de trois lancers, correspondant aux graines (*seeds*) 42, 142 et 1234. Nous constatons des résultats stables, avec un écart-type moyen de moins de 1 pour chaque métrique.

Enfin, pour les modèles hiérarchiques, la segmentation supplémentaire des types de mots en morphèmes (procédé `pipeline`) s'opère à travers 1 000 itérations (iter_f) d'échantillonnage de Gibbs après convergence pour le modèle `hier-final` et 5 itérations de segmentation morphologique (iter_r) toutes les 100 itérations de segmentation en mots (batch_m) pour `hier-iter` (voir section 5.2.3).

5.3.2 Métriques d'évaluation

Nous utilisons les mêmes métriques que dans le chapitre précédent (section 4.2.1) : la précision, le rappel et le score F1 aux niveaux des frontières (B), des occurrences (W) et des types (L). Comme nous avons deux niveaux de segmentation, nous comparons donc les prédictions avec un texte de référence segmenté à chacun de ces niveaux. La figure 5.5 illustre les comparaisons effectuées pour une phrase d'exemple en japhug.

Nous comparons donc indépendamment chacun de ces niveaux : ainsi, l'erreur de la valeur de la frontière après « p » impacte uniquement les scores des morphèmes, là où la frontière de mot prédite après « ɯ » affecte le score des mots. Par ailleurs, la frontière de fin de phrase (frontière de mots, par définition) n'est également pas considérée lors de l'évaluation.

Une autre possibilité, non traitée ici, aurait été d'utiliser une représentation unique des frontières : 0 en l'absence de frontière, 1 pour les morphèmes et 2 pour les mots.

	niveau	χ	p	ɯ	n	ɯ	p	ɯ
référence	mot	0	0	0	1	**0**	0	1
	morph.	0	**0**	0	1	1	0	1
prédiction	mot	0	0	0	1	**1**	0	1
	morph.	0	**1**	0	1	1	0	1

(a) Au niveau des frontières

référence	mot	χpɯn	**ɯpɯ**
	morph.	**χpɯn**	ɯ pɯ
prédiction	mot	χpɯn	**ɯ pɯ**
	morph.	**χp ɯn**	ɯ pɯ

(b) Au niveau des occurrences

FIGURE 5.5 – Exemple de phrase en japhug (χpɯn ɯ–pɯ, « petit moine », en français), représentée aux deux niveaux de segmentation pour l'évaluation. Les deux erreurs sont en **gras**.

5.3.3 Résultats sans supervision

Nos travaux s'intéressent aux langues japhug et tsez, pour lesquelles nous avons accès à une segmentation en mots et en morphèmes. Les tableaux 5.1 et 5.2 présentent les résultats obtenus par l'*Adaptor Grammar* (AG), par le modèle `dpseg` à un niveau (marqué par l'astérisque) ainsi que par les différents modèles présentés en section 5.2, sans supervision. La séparation en deux tableaux (les modèles en cascade et couplés dans le premier, les modèles hiérarchiques dans le second) est uniquement à des fins de lisibilité.

modèle	AG		`dpseg*`		`pipeline`		`parallel-w`		`parallel-m`	
niveau	mot	morph.	mot	morph.	mot	morph.	mot	morph.	mot	morph.
BP	**70,9**	77,3	61,3	**87,8**	61,3	69,3	61,5	84,9	64,5	87,6
BR	71,1	90,4	90,6	75,2	90,6	**96,3**	**90,8**	82,4	84,5	75,0
BF	71,0	83,4	73,1	81,0	73,1	80,6	**73,3**	**83,6**	73,2	80,8
WP	**45,8**	58,3	39,4	59,3	39,4	50,1	39,7	**62,1**	41,1	58,9
WR	45,9	67,3	55,9	51,5	55,9	**68,2**	**56,2**	60,4	52,3	51,1
WF	45,8	**62,5**	46,2	55,1	46,2	57,8	**46,5**	61,3	46,0	54,7
LP	34,3	49,9	40,6	45,7	40,6	43,3	**41,1**	**50,5**	39,4	45,4
LR	**28,4**	20,3	13,6	37,8	13,6	11,0	13,8	33,9	17,1	**37,9**
LF	**31,1**	28,9	20,4	**41,4**	20,4	17,5	20,6	40,5	23,8	41,4
WL	4,72	2,51	3,34		3,34	2,13	3,34	2,98	3,73	3,35
TL	6,60	3,27	4,22		4,22	2,64	4,23	3,99	4,77	4,21
N_{type}	5582	1113	2260		2260	694	2257	1834	2921	2281
N_{token}	28,6k	53,9k	40,5k		40,5k	63,4k	40,5k	45,4k	36,3k	40,4k

TABLE 5.1 – Première partie des résultats des modèles de segmentation à un (marqué par *) et deux niveaux sur le corpus japhug sans supervision (modèles en cascade et couplés ; moyenne de trois lancers). Les meilleurs scores par métrique et niveau de segmentation sont en **gras**.

Comparons tout d'abord nos modèles avec `dpseg`. La méthode `pipeline`, qui segmente en morphèmes les *types* de mots obtenus par une segmentation automatique, présente des scores décevants, en particulier très dégradés au niveau des types de morphèmes (17,5). Nous remarquons, par ailleurs, que les longueurs moyennes des unités segmentées (WL et TL) sont les plus courtes observées (à titre comparatif, les longueurs de référence sont respectivement de 2,90 et 5,41 caractères), suggérant la présence de sur-segmentation. En effet, nous constatons un écart significatif entre les scores WF et LF au niveau des morphèmes : quelques morphèmes très fré-

modèle	AG		dpseg*		hier-type		-final	hier-iter	
niveau	mot	morph.	mot	morph.	mot	morph.	morph	mot	morph.
BP	**70,9**	77,3	61,3	**87,8**	67,6	48,4	73,6	69,6	73,5
BR	71,1	90,4	**90,6**	75,2	83,4	88,2	**93,4**	77,8	91,2
BF	71,0	**83,4**	73,1	81,0	**74,7**	62,5	82,3	73,5	81,4
WP	**45,8**	58,3	39,4	**59,3**	44,6	19,2	54,9	44,9	53,7
WR	45,9	67,3	**55,9**	51,5	53,7	33,8	**68,5**	49,6	65,7
WF	45,8	**62,5**	46,2	55,1	**48,7**	24,5	60,9	47,2	59,1
LP	34,3	**49,9**	**40,6**	45,7	40,0	31,5	46,7	37,5	47,6
LR	**28,4**	20,3	13,6	**37,8**	22,6	11,7	15,6	27,4	16,7
LF	31,1	28,9	20,4	**41,4**	28,8	17,0	23,4	**31,7**	24,7
WL	4,72	2,51	3,34		3,93	1,65	2,32	4,29	2,37
TL	6,60	3,27	4,22		4,78	2,83	2,87	5,12	2,88
N_{type}	5582	1113	2260		3806	1013	911	4925	956
N_{token}	28,6k	53,9k	40,5k		34,4k	82,1k	58,2k	31,5k	57,0k

TABLE 5.2 – Deuxième partie des résultats des modèles de segmentation à un (marqué par *) et deux niveaux sur le corpus japhug sans supervision (modèles hiérarchiques ; moyenne de trois lancers). Les meilleurs scores par métrique et niveau de segmentation sont en **gras**.

quents contribuent à améliorer le score WF d'environ 2 points par rapport à dpseg, au détriment des morphèmes rares (et longs). Les très hautes valeurs de rappel aux niveaux des frontières et des occurrences soulignent aussi cet effet de sur-segmentation.

Pour les méthodes couplées, le niveau segmenté en premier (les mots pour parallel-w et les morphèmes pour parallel-m), par construction, ne diffère pas des résultats obtenus par dpseg ; les variations numériques s'expliquent par le caractère aléatoire de la procédure d'échantillonnage. Le modèle parallel-w améliore principalement les scores au niveau des frontières (+2 points) et des occurrences de morphèmes (+6 points) et présente en contrepartie une baisse des résultats des types (-1 point). Nous constatons donc ici aussi un phénomène de sur-segmentation, mais dans des proportions plus modérées qu'avec pipeline. Dans le cas du modèle parallel-m, l'amélioration est plus modeste, avec une hausse notable (+3 points) du F-score au niveau des types de mots seulement. Nous retrouvons sinon des valeurs proches des celles de dpseg.

Modèles hiérarchiques En ce qui concerne les modèles hiérarchiques, présentés dans le tableau 5.2, nous observons tout d'abord que hier-type apporte une véritable amélioration des scores au niveau des mots par rapport à dpseg : +1 point pour les frontières, +2 points pour les occurrences et +8 points pour les types. En revanche, au niveau des morphèmes, les résultats sont très altérés avec des baisses significatives des trois F-scores (de -18 à 30 points). Les statistiques indiquent que les morphèmes obtenus par segmentation sont particulièrement courts dans ce cas (une longueur moyenne d'occurrences de 1,65). Il s'avère donc que ce modèle hiérarchique de base favorise de manière excessive la segmentation en morphèmes.

Le recours à un procédé similaire à pipeline après la dernière itération de segmentation, à savoir un ré-échantillonnage des *types* de mots identifiés permet de remédier à ce problème, comme le montrent les résultats de hier-final. Une simple étape finale permet d'obtenir de meilleures valeurs de BF et WF que dpseg. Le niveau des types semble toutefois être le point

faible du procédé `pipeline`, qui aboutit à une identification correcte des morphèmes fréquents au détriment des plus rares, comme en témoigne l'écart plus marqué entre WF et LF.

Enfin, intégrer le ré-échantillonnage des types de mots au sein du processus de segmentation (`hier-iter`) apporte une différence sensible par rapport à `hier-final` : les scores BF et WF des mots et des morphèmes sont sensiblement baissés pour une légère hausse des scores de types. Le choix entre ces deux variantes dépend donc du contexte et des préférences ; il s'agit d'un compromis entre des performances améliorées pour les occurrences ou les types.

5.3.4 Comparaison des différents types de modèles

L'approche `pipeline` est la moins performante, avec des scores plus bas au niveau des morphèmes, par rapport aux deux autres types de modèles de segmentation jointe. Entre ces derniers, si les modèles couplés obtiennent de meilleurs scores au niveau des morphèmes, les approches hiérarchiques atteignent de plus hauts scores de mots. Au-delà de cette tendance, les modèles hiérarchiques permettent de générer des mots plus longs (de 4 caractères environ), et augmentent par là la distinction de ces unités vis-à-vis des morphèmes. Ils répondent donc davantage à notre question initiale sur la différenciation des deux niveaux de segmentations.

L'*Adaptor Grammar*, modèle hiérarchique par nature, présente des résultats, dans l'ensemble, assez semblables aux modèles présentés, avec de meilleurs scores de types et de morphèmes. De plus, il parvient à obtenir des mots plus longs (le plus long TL parmi les modèles, 6,60), se rapprochant grandement de la réalité, ce qui explique en partie ses bons scores LF, comparativement.

Néanmoins, de manière générale, nous observons que les unités identifiées restent trop courtes par rapport à la référence, même dans les meilleurs cas, en particulier pour les types. En effet, l'AG identifie certes des occurrences de mots de 4,72 caractères (contre 4,73 en réalité), mais ses types restent trop courts, de 0,7 caractères. Pour les morphèmes, la différence est plus visible, avec des unités de 5,41 caractères en référence. Ce constat est d'autant plus valable pour nos modèles basés sur `dpseg` ; si les longueurs des mots semblent relativement proches, il y a environ 2 caractères de différence pour les morphèmes. Sur ce niveau, le phénomène de sur-segmentation semble donc plus marqué.

Nous pouvons donc conclure que l'essentiel des performances de segmentation repose sur les unités (mots et morphèmes) les plus courtes et fréquentes, permettant d'atteindre de hauts scores WF ; celles qui sont plus longues et plus rares restent un défi majeur pour les modèles statistiques, comme l'indiquent les scores systématiquement bas pour le LF. Ce phénomène est plus accentué lorsque l'écart entre ces deux F-scores est prononcé. Par ailleurs, nous observons également que l'incorporation de la hiérarchie entre les mots et les morphèmes (modèles hiérarchiques et AG) permet d'obtenir une distinction plus visible des deux niveaux de segmentation, concordant avec de meilleures performances selon les métriques.

5.3.5 Supervision faible

En s'appuyant sur les méthodologies proposées dans le chapitre précédent, il est aussi possible de recourir à une supervision faible pour améliorer les performances des modèles. Nous nous intéressons donc aux meilleures stratégies pour les deux types de ressources : `sentence` lorsque des phrases déjà segmentées sont disponibles et `dictionary` avec un dictionnaire de mots et de morphèmes. Au vu des résultats obtenus, pour une meilleure lisibilité dans cette section, nous ne

reportons pas les résultats du modèle `hier-type`, dont les résultats sont strictement plus bas que sa variante `hier-final`.

modèle	CRF		dpseg		pipe.	parallel-w		parallel-m		hier-final		hier-iter	
niveau	W	M	W	M	M	W	M	W	M	W	M	W	M
BP	**73,5**	83,2	63,8	88,1	79,2	64,0	86,4	66,4	**88,9**	70,9	80,9	72,4	80,1
BR	80,8	85,2	91,4	79,6	**97,4**	**91,4**	83,7	86,3	77,7	84,0	96,0	80,2	94,5
BF	**77,0**	84,2	75,1	83,6	87,3	75,3	85,1	75,0	82,9	76,9	**87,8**	76,1	86,7
WP	**52,6**	66,4	43,7	64,3	67,2	43,9	65,6	44,8	63,6	49,6	**68,5**	49,8	66,9
WR	57,3	67,8	60,2	58,7	**81,5**	**60,3**	63,7	56,5	56,3	57,5	80,3	54,5	78,0
WF	**54,9**	67,1	50,6	61,4	73,6	50,8	64,7	50,0	59,7	53,3	**74,0**	52,1	72,0
LP	39,4	27,5	50,7	53,9	**61,6**	**51,2**	55,3	47,2	51,1	46,3	59,6	43,2	60,8
LR	**49,5**	**50,3**	19,6	40,2	23,5	19,9	39,4	22,3	42,7	30,0	25,9	33,4	26,4
LF	**43,9**	35,5	28,3	45,8	34,0	28,7	46,0	30,3	**46,5**	36,4	36,1	37,6	36,8
WL	4,35	2,84	3,44	3,19	2,39	3,45	2,99	3,75	3,28	4,08	2,48	4,32	2,49
TL	6,67	5,09	4,66	4,25	3,44	4,66	4,12	5,04	4,30	5,13	3,47	5,33	3,46
N_{type}	8453	4999	2610	2061	1040	2627	1946	3182	2283	4363	1186	5208	1185
N_{token}	31,1k	47,6k	39,4k	42,5k	56,5k	39,2k	45,3k	36,0k	41,2k	33,2k	54,6k	31,3k	54,4k

TABLE 5.3 – Résultats des modèles CRF, `dpseg` et ses extensions à deux niveaux (W pour les mots, M pour les morphèmes) sur le corpus japhug, supervisés par des annotations denses (**`sentence`**) de 200 phrases (moyenne de trois lancers). Les meilleurs scores par métrique et niveau de segmentation sont en **gras**.

Supervision par des phrases déjà segmentées Le tableau 5.3 présente les résultats des différents modèles à deux niveaux avec la supervision par des annotations denses (`sentence`) sur le corpus japhug. Les deux colonnes `dpseg` correspondent au modèle à un niveau supervisé faiblement par des phrases segmentées respectivement en mots ou en morphèmes. Il est donc difficile de combiner ces deux segmentations, indépendantes, afin d'obtenir des phrases segmentées aux deux niveaux ; la cohérence des frontières n'est pas garantie.

L'approche `pipeline` aboutit cette fois à de meilleurs scores par rapport au modèle `dpseg` de base, supervisé par des annotations de morphèmes ; les scores de BF et de WF étant parmi les meilleurs. Nous noterons toutefois la dégradation marquée au niveau du LF, explicable par le phénomène, à nouveau, de sur-segmentation : le score de rappel des frontières (BR) est le plus élevé, les unités sont plus courtes (moins de trois caractères par morphème, en moyenne) et trop peu de types sont générés (à peine 1 000). La supervision faible semble modérer cette tendance inhérente au procédé `pipeline`, sans pour autant la contrebalancer entièrement. Le grand écart entre le score au niveau des occurrences et des types confirme que l'essentiel du succès de ce modèle repose sur les morphèmes les plus fréquents.

Les modèles couplés conduisent aux mêmes constats que dans le cadre sans supervision. Le modèle segmentant d'abord les mots (`parallel-w`) obtient de meilleurs résultats au niveau des morphèmes, en particulier pour le BF et le WF, comparé à `dpseg`. Dans une moindre mesure, `parallel-m` parvient également à atteindre des valeurs similaires pour les F-scores des frontières et des occurrences voire meilleurs pour celui des types au niveau des mots. Nous avons donc ici des segmentations à deux niveaux, systématiquement équivalentes ou meilleures que les segmentations de `dpseg`, supervisées indépendamment.

En ce qui concerne les modèles hiérarchiques, les performances au niveau des mots sont les meilleurs parmi nos variantes de dpseg, en particulier pour les scores de types, qui atteignent environ 36. Pour les morphèmes, le modèle hier-final obtient les meilleurs scores de frontières et d'occurrences. De plus, hier-final et hier-iter, aux résultats similaires, permettent d'avoir un compromis entre d'une part les scores BF et WF, d'autre part le score LF.

Enfin, le CRF, qui bénéficie d'une supervision intégrale, obtient les meilleurs F-scores au niveau des mots, avec notamment un LF de 44, loin devant nos modèles. Ces valeurs peuvent s'expliquer par la longueur des unités, les plus longues (et parmi les plus proches de la réalité) observées et par la génération d'un nombre presque doublé de types de mots par rapport à hier-final. La proximité des valeurs de précisions et de rappels ou de WF et de LF suggèrent que ce modèle subit moins l'effet de concentration d'unités, comme observé avec nos modèles. Cependant, comme les valeurs des BF et des WF sont relativement proches de nos modèles (environ 1 point de différence avec hier-final), il s'agit donc d'un modèle complémentaire à notre approche, qui parvient à générer également des unités rares et correctes, mais reposant moins sur les mots fréquents.

modèle	dpseg		pipe.	parallel-w		parallel-m		hier-type		hier-iter	
niveau	W	M	M	W	M	W	M	W	M	W	M
BP	**76,6**	**93,2**	87,0	76,6	91,0	76,4	93,0	66,4	83,6	66,6	84,3
BR	81,0	64,3	83,1	81,2	71,3	74,9	64,2	89,6	90,2	**90,1**	**90,8**
BF	78,7	76,1	85,0	**78,8**	80,0	75,6	76,0	76,2	86,8	76,6	**87,4**
WP	54,4	54,8	65,9	**54,5**	60,1	51,6	54,4	45,6	67,2	46,0	**68,7**
WR	57,1	39,2	63,2	57,4	48,1	50,7	38,9	59,6	72,1	**60,1**	**73,6**
WF	55,7	45,7	64,5	**55,9**	53,5	51,1	45,4	51,7	69,6	52,1	**71,1**
LP	49,9	37,0	47,0	**50,5**	40,9	46,4	37,2	46,4	56,0	47,3	**57,9**
LR	37,3	54,8	43,5	**37,8**	52,3	36,8	**54,8**	21,6	34,3	21,9	34,9
LF	42,7	44,2	45,2	**43,2**	**45,9**	41,1	44,3	29,5	42,5	29,9	43,6
WL	4,51	4,06	3,03	4,49	3,62	4,81	4,06	3,63	2,71	3,62	2,71
TL	6,18	5,40	4,45	6,16	5,14	6,49	5,38	4,46	3,84	4,52	3,86
N_{type}	5041	4044	2524	5040	3492	5356	4027	3141	1671	3116	1646
N_{token}	30,0k	33,3k	44,7k	30,1k	37,3k	28,1k	33,3k	37,3k	50,0k	37,4k	49,9k

TABLE 5.4 – Résultats des modèles dpseg et ses extensions à deux niveaux (W pour les mots, M pour les morphèmes) sur le corpus japhug, supervisés par des dictionnaires de mots et de morphèmes (**dictionary**) obtenus sur 200 phrases (moyenne de trois lancers). Les meilleurs scores par métrique et niveau de segmentation sont en **gras**.

Supervision par une liste de mots et de morphèmes Le tableau 5.4 présente les résultats des modèles supervisés cette fois par un dictionnaire de mots et de morphèmes, dictionary. La colonne dpseg correspond là aussi à deux segmentations supervisées indépendamment. Le procédé pipeline parvient cette fois à surpasser les scores du modèle à un niveau, même pour le LF, aidé vraisemblablement par les unités données par la supervision. Toutefois, il souffre toujours du problème de sur-segmentation, comme en témoigne la longueur des unités : il y a une différence d'un caractère pour les occurrences et les types avec les prédictions de dpseg à un niveau. Si cela reflète davantage la réalité au niveau des occurrences (car le WL est de 2,90 dans la référence), les types sont trop courts et s'éloignent de la valeur théorique de 5,41.

Les modèles couplés se différencient davantage ici : si `parallel-w` maintient toujours un avantage notable au niveau des morphèmes, vis-à-vis de son équivalent à un niveau, le modèle `parallel-m` décroche et dégrade les trois F-scores de mots comparativement. Malgré des unités plus longues et un nombre de types plus important, les unités générées ne conviennent pas. De manière plus générale, il semble ainsi plus efficace de commencer par la segmentation des mots puis des morphèmes (`parallel-w`) que l'inverse (`parallel-m`) pour les modèles couplés, étant donné les meilleures performances obtenues dans les trois situations de supervision. De fait, bien qu'il n'améliore pas la qualité des mots identifiés, ce meilleur modèle apparaît plus adapté pour les morphèmes, à travers sa tendance à sur-segmenter.

Enfin, pour les modèles hiérarchiques, nous constatons que les résultats au niveau des mots sont moins bons que ceux de `dpseg`, en particulier pour les types, alors que les F-scores pour les frontières et les occurrences sont à nouveau les meilleurs parmi nos modèles. Ce phénomène semble tenir de la manière d'intégrer les informations du dictionnaire dans le modèle : les mots connus interviennent uniquement dans le premier terme de la distribution de base modifiée $\widetilde{P''}_0$ (équation (5.4)), tandis que le modèle des morphèmes bénéficie également d'un meilleur modèle de caractères. Cette différence dans l'intégration suggérerait donc l'importance du modèle bigramme dans les améliorations obtenues par `dpseg` de manière générale. Une piste d'amélioration serait donc d'incorporer dans $\widetilde{P''}_0$ la probabilité de succession des morphèmes ou des caractères, en se référant aux mots du dictionnaire, en introduisant une dépendance bigramme par exemple. Toutefois, malgré ces scores élevés au niveau des morphèmes, l'identification des types rares paraît plus difficile que chez les autres modèles, comme en témoignent les scores sensiblement plus bas.

Comparaison entre modèles et situations Nous avons jusque là comparé les différents modèles avec la version standard de `dpseg` à un niveau. La figure 5.6 récapitule leurs résultats obtenus avec et sans supervision. Nous ne conservons que le meilleur modèle par catégorie, choisi en fonction du score agrégé de tous les F-scores : `parallel-w` pour les modèles couplés et `hier-final` pour les modèles hiérarchiques. Le modèle indépendant représente le modèle `dpseg` à un niveau, supervisé par les ressources correspondantes.

Au niveau des mots, tout d'abord, les performances de tous les modèles sauf `hier-final` sont (presque) identiques, par construction. Le modèle hiérarchique se démarque dans les cas sans supervision ou avec des annotations denses (`sentence`). Nous constatons ainsi que l'intégration de la structure des mots permet d'améliorer la qualité de segmentation de `dpseg`. L'impact du dictionnaire est toutefois délétère, avec l'implémentation actuelle du moins.

Dans le cas des morphèmes, les résultats sont plus disparates. L'utilisation de l'approche `pipeline` permet d'obtenir de meilleurs scores aux niveaux des frontières et des occurrences (et une segmentation à deux niveaux cohérente) par rapport à la version indépendante dans les trois configurations de supervision. La chute de performance au niveau des types est cependant préjudiciable. Le modèle `parallel-w`, quant à lui, permet d'améliorer toutes les métriques (sauf le LF sans supervision) et ne subit pas une baisse de performance drastique au niveau des types. Enfin, le modèle hiérarchique obtient là aussi de meilleurs résultats que `dpseg`, mais les scores au niveau des types restent faibles. Dans ses version supervisées, il parvient à combler partiellement ses faiblesses. Nous noterons que l'utilisation de la supervision `sentence` permet d'améliorer systématiquement les performances des modèles couplés et hiérarchiques ; l'impact du dictionnaire, positif au niveau des mots, semble plus mitigé pour les morphèmes.

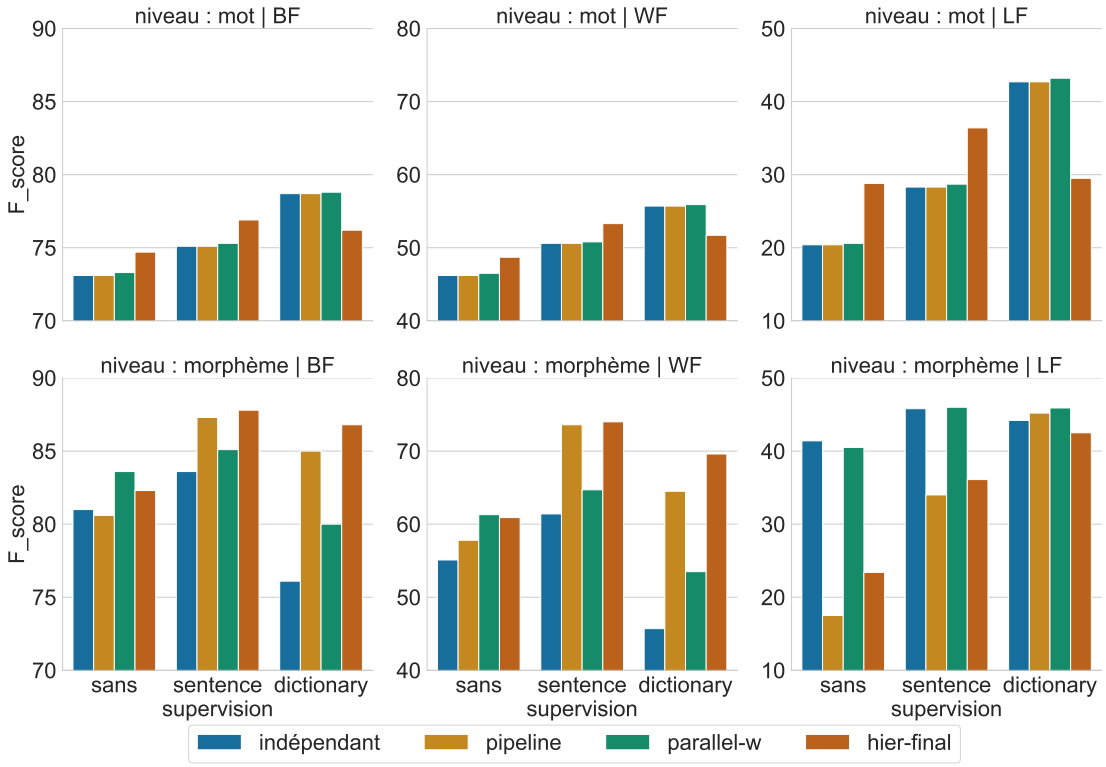


FIGURE 5.6 – Comparaison des résultats pour les quatre types de modèles avec et sans supervision faible. Chaque modèle est représenté par une couleur, les métriques sont calculées pour les frontières (gauche), les occurrences (milieu) et les types (droite) aux niveaux des mots (haut) et des morphèmes (bas). Enfin, les barres sont regroupées par configuration de supervision : sans supervision (gauche), `sentence` (milieu) et `dictionary` (droite).

Nous pouvons à présent classer les différents types de modèles : indépendant, `pipeline`, `parallel-w` et `hier-final`, par ordre croissant selon le score agrégé. Dans l'ensemble, la hausse des scores est néanmoins assez faible, par rapport au modèle indépendant. L'introduction d'un second niveau de segmentation ne permet donc pas de différencier les mots des morphèmes de manière satisfaisante, au point de voir une amélioration plus notable des métriques ; le modèle hiérarchique répond en partie à ces besoins, mais insuffisamment. De même, les autres modèles statistiques testés, hiérarchique et non supervisé comme l'*Adaptor Grammar* ou entièrement supervisé comme le CRF, font face à la même limite : l'identification des unités rares, aussi bien pour les mots que pour les morphèmes, qui se traduit par le faible score au niveau des types.

5.3.6 Analyse qualitative de la segmentation à deux niveaux

La figure 5.7 présente une phrase japhug segmentée par différents types de modèles de segmentation. Avec le modèle de base de segmentation à un niveau, `dpseg`, nous observons quelques erreurs communes comme une frontière de mots au lieu d'une frontière de morphème ou l'absence de frontière là où il devrait y en avoir une. Il faut noter que ces types d'erreurs, en particulier sans supervision, aboutissent à des unités segmentées dénuées de sens et impactent les trois niveaux d'évaluation. Introduire un deuxième niveau de segmentation (avec `parallel-w`, par exemple) permet de remédier partiellement à ces problèmes ; davantage de frontières sont identifiées, même si elles ne sont pas nécessairement du bon type, comme entre « a–mbro » et « ɯ–jme ».

modèle	supervision	phrase
`dpseg`	/	a mbroɯjme zɯ kɤzo
`parallel-w`	/	a mbro–ɯjme z ɯkɤ zo
`parallel-w`	`sentence`	a–mbro ɯjme zɯ kɤ–zo
`hier-final`	/	a–mbro ɯ–jme zɯ kɤ–zo
`hier-final`	`sentence`	a–mbro ɯ–jme zɯ kɤ–zo
référence		a–mbro ɯ–jme zɯ kɤ–zo

FIGURE 5.7 – Exemple de phrase en japhug segmentée par les différents modèles, avec et sans supervision : « Atterrissez sur la queue de mon cheval ».

Enfin, à travers une supervision faible (ici, `sentence`), nous observons une meilleure prédiction des frontières : le modèle détermine correctement les frontières de mots et de morphèmes, à l'exception d'une position dans le mot « ɯ–jme ». L'erreur provient principalement du problème de cooccurrences observées dans les données (Goldwater et al., 2009) : toutes les occurrences du morphème « jme » sont précédées du morphème « ɯ » dans le corpus. L'unité la plus probable d'un point de vue statistique est alors « ɯjme » ; il s'agit là d'un héritage de la limite principale du modèle `dpseg` unigramme.

En ce qui concerne les modèles hiérarchiques, ici représentés par le modèle `hier-final` (bien que `hier-iter` ait obtenu les mêmes prédictions dans ce cas précis), la phrase est correctement segmentée avec ou sans supervision faible. Nous pouvons donc voir un exemple des bénéfices de la modélisation de la structure des mots via la distribution de base P_0. Elle a notamment permis de contourner certains biais de ce modèle, comme l'effet des cooccurrences fréquentes.

5.4 Analyse des distributions des unités

Les résultats obtenus dans la section précédente sont quelque peu décevants ; les phénomènes de sur-segmentation restent présents et la distinction des frontières de mots et de morphèmes émerge difficilement, même avec supervision. Au fondement de nos modèles bayésiens non paramétriques se trouve l'hypothèse d'une distribution en loi de puissance des unités (Goldwater et al., 2005), vérifiée en pratique dans les corpus de langues variées. En effet, c'est pour cette raison que nous utilisons un modèle à deux étages composé d'un générateur (représenté par la distribution de base P_0) de types et d'un adapteur permettant de les répartir suivant cette loi (créant les occurrences). Nous nous intéressons ici de plus près à cette hypothèse sur nos corpus étudiés dans le cadre de la documentation des langues : dans quelle mesure vérifient-ils cette distribution naturellement observée ?

5.4.1 Courbe type-occurrence

Une première méthode pour évaluer cette hypothèse est d'examiner l'évolution du rapport type-occurrence (ou en anglais, *Type-Token Ratio*, TTR). De fait, il s'agit d'une façon de visualiser l'effet des « plus riches devenant plus riches », hypothèse de base de `dpseg`. Nous comparons nos deux corpus avec des textes de langues variées d'un point de vue morphologique : allemand, anglais, finnois, français et turc (ordonnées alphabétiquement). Les données correspondent à des textes journalistiques de 2020, provenant du *Leipzig Corpora Collection* (Goldhahn et al., 2012).

Afin d'avoir une évaluation plus comparable avec nos données, nous utilisons uniquement les 2 000 premières phrases de ces corpus.

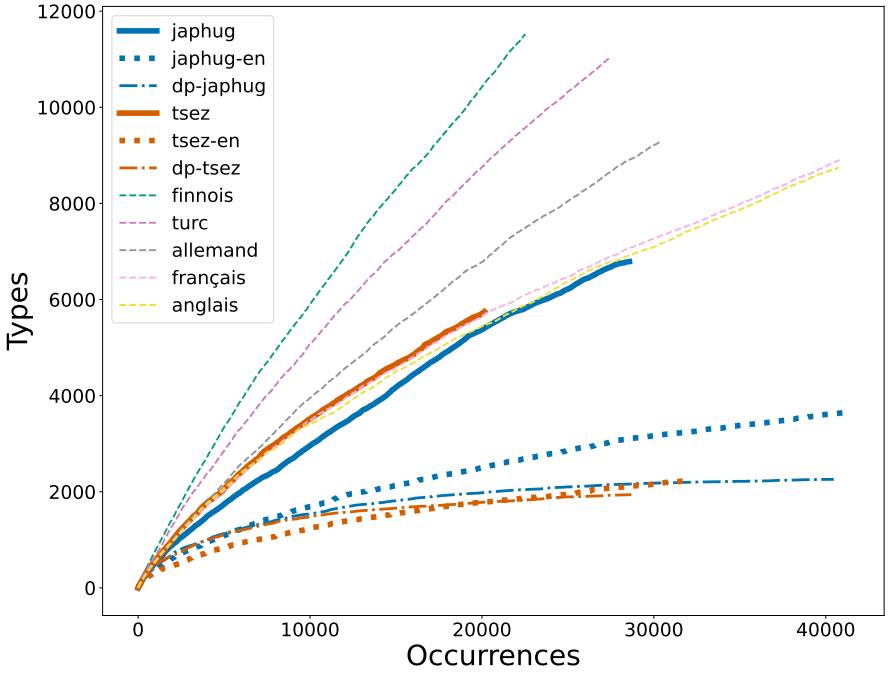


FIGURE 5.8 – Courbes des rapports type-occurrence pour plusieurs langues.

La figure 5.8 présente les différentes courbes type-occurrence. Nous remarquons tout d'abord que celles des langues morphologiquement riches comme le finnois ou le turc sont plus pentues par rapport au français ou l'anglais, l'allemand se situant au milieu. Les deux langues étudiées, selon leur courbe, s'apparentent alors davantage au français ou à l'anglais qu'aux langues comme le finnois, bien qu'elles aient une morphologie plus complexe que les premières.

Pour avoir un regard différent sur ce phénomène, nous avons aussi considéré les traductions en anglais des deux corpus, identifiées par le marqueur « -en » en légende et par des traits pointillés pour la courbe. Les pentes correspondantes sont alors toutes deux significativement plus basses par rapport à celle de l'anglais du *Leipzig Corpora Collection*. Les deux corpus sont donc très concentrés ; comme en témoignent les allures divergentes des trois courbes de l'anglais, les mêmes mots sont répétés plus souvent que dans un autre domaine (ici, journalistique). De fait, le corpus tsez est constitué de contes folkloriques, où les personnages sont volontiers répétés d'une phrase à une autre. Le corpus japhug quant à lui provient d'exemples grammaticaux utilisés pour illustrer des phénomènes linguistiques particuliers, eux-mêmes parfois issus de récits. Nous observons ainsi que les corpus traités dans le cadre de la documentation des langues présentent ici des distributions de mots particulières, plus répétitives.

Enfin, les prédictions obtenues par `dpseg` (« dp- ») sont aussi représentées par les traits en pointillés irréguliers. Ces courbes suivent alors une tendance très similaire à celles des corpus traduits en anglais : elles sont trop peu pentues par rapport aux attentes théoriques. Cette différence dans les pentes reflète alors le manque de diversité dans les unités segmentées, constaté notamment par le faible score LF tout au long de nos expériences.

Il faut toutefois nuancer que l'analyse effectuée présente uniquement une comparaison avec le domaine des textes journalistiques, qui présente sa propre distribution type-occurrence, vraisem-

blablement différente d'autres contextes (comme les romans, par exemple). Néanmoins, l'inclinaison des courbes représentant nos deux corpus traduits soulignent une différence significative dans la fréquence d'apparition des mots dans nos corpus par rapport à d'autres textes.

5.4.2 Visualisation de la loi de Zipf dans nos corpus

La loi de Zipf explicite, pour une unité, la relation entre sa fréquence normalisée f ($f = \frac{F}{N}$ où F est la fréquence de l'unité et N le nombre total d'unités dans le texte) et son rang (de Zipf) R dans un corpus, selon l'équation (5.5) :

$$f = \frac{c}{R^a} \tag{5.5}$$

où c est une constante de normalisation et a le paramètre de la distribution (Baayen, 2001). Par une transformation logarithmique, nous obtenons alors l'équation (5.6), qui décrit une relation linéaire entre $\log(f)$ et $\log(R)$.

$$\log(f) = -a\log(R) + \log(c) \tag{5.6}$$

Afin d'observer cette relation, nous effectuons une régression linéaire des moindres carrés sur des données de langues différentes en calculant f et R au niveau des mots, illustrée dans la figure 5.9. L'idée est d'étudier l'allure des courbes obtenues ainsi que les différences dans les pentes a. Nous remarquons que plus la valeur de a est grande, plus les fréquences des unités sont concentrées, alors que pour des plus petites valeurs de a, les unités sont plus réparties. Pour les autres langues, nous utilisons les mêmes données que dans la section précédente (5.4.1), mais cette fois avec 1 600 phrases uniquement, pour obtenir un nombre total d'occurrences similaire aux corpus japhug et tsez. En effet, la pente de la courbe dépend de la taille du corpus (Baayen, 2001).

Tout d'abord, dans ces courbes obtenues par régression, la relation linéaire s'observe généralement pour les unités de rang intermédiaire en particulier (ici entre 10^1 et 10^3 environ) : les mots les plus fréquents ont des fréquences plus volatiles et la fin de distribution (les types les plus rares) aboutit à des barres horizontales (Baayen, 2001).

Nous observons que les courbes obtenues pour nos deux corpus sont plus proches de l'anglais que de l'allemand ou du finnois, plus riches morphologiquement. Pour le japhug, nous remarquons une proximité manifeste avec l'anglais, en particulier pour les hauts rangs Zipf (c'est-à-dire, les mots rares), où les courbes semblent se rejoindre. De plus, pour les mots de rangs faibles (donc fréquents), les mots en japhug obtiennent des fréquences normalisées plus élevées que l'allemand ou le finnois. Nous constatons donc à nouveau une concentration des mots dans le corpus japhug, similaire à la distribution de l'anglais, alors même que la langue a une morphologie plus complexe ; il serait ainsi attendu de voir une pente a moins élevée, plus proche du finnois par exemple.

Par ailleurs, la distribution des mots tsez est particulière par rapport aux autres : ses unités de rang faibles sont significativement moins fréquentes et celles de rang supérieur à 100 environ sont plus fréquentes que pour les autres langues. L'aspect répétitif du texte, identifié en section précédente, ne s'explique pas ici par ses mots les plus fréquents, comme en japhug, mais concerne plutôt les mots de rang moyen ; la distribution des fréquences des mots est alors la plus abrupte parmi les cinq langues considérées, bien que les mots en tête de la distribution (rang faible) soient moins fréquents en valeur absolue que dans les autres langues.

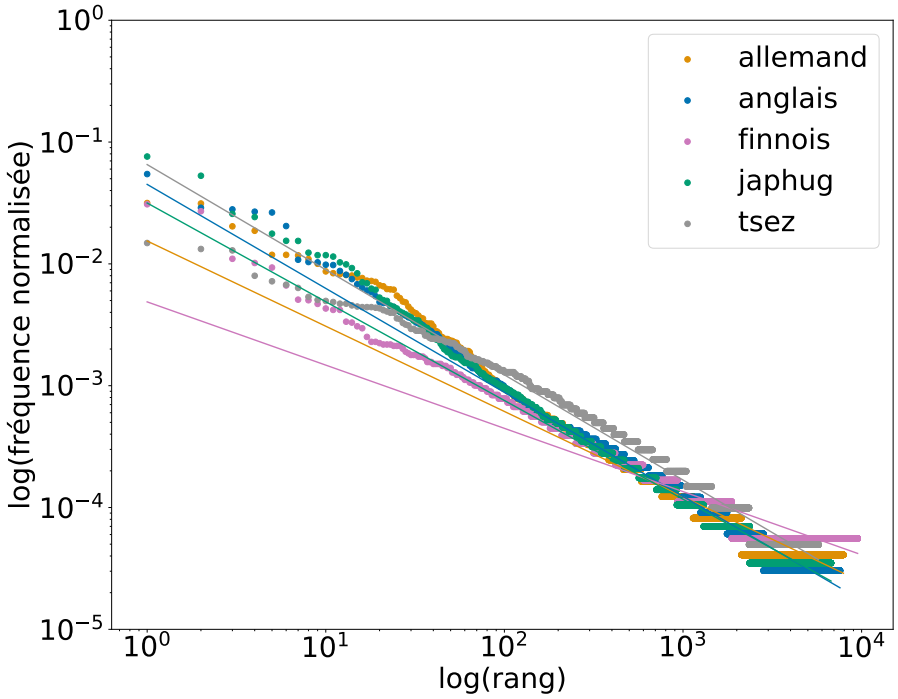


FIGURE 5.9 – Courbes logarithmiques de la fréquence normalisée par rapport au rang de Zipf dans plusieurs langues.

5.4.3 Modélisation de la distribution des morphèmes

Au-delà des différences d'implémentation, les deux catégories de modèles de segmentation à deux niveaux diffèrent par leur approche vis-à-vis de l'obtention des morphèmes : la méthode couplée effectue une segmentation des *occurrences* des mots, là où la vision hiérarchique considère les *types* de mots. De ce fait, en utilisant le modèle `dpseg`, se pose la question de la distribution en loi de puissance pour les morphèmes : concerne-t-elle davantage un corpus constitué d'occurrences ou de types de mots ?

Pour y répondre, nous avons calculé les valeurs des pentes a de régression linéaire, comme définies en section 5.4.2, en tsez aux deux niveaux de segmentation, illustrées à la figure 5.10. Nous comparons la référence avec les versions segmentées par `parallel-w` et `hier-final`, afin de représenter les deux catégories de modèles. Pour les morphèmes, nous ne considérons que les *types* de mots pour obtenir les fréquences des morphèmes.

Comme nous avons pu le constater dans la section précédente, le texte tsez de référence présente une pente particulière au niveau des mots. Bien que sa valeur dépende de la taille du corpus, elle se situe en général aux alentours de $[-1{,}2, -1]$ (Baayen, 2001), ce qui est le cas au niveau des morphèmes. Pour les six segmentations obtenues avec nos modèles et méthodes de supervision, nous remarquons que la valeur des pentes au niveau des mots suit la tendance attendue d'une loi de Zipf standard.

En revanche, au niveau des morphèmes, la différence est apparente : les modèles hiérarchiques permettent de mieux s'approcher de la distribution des morphèmes dans le texte, comme en témoigne la valeur du coefficient a, comparé aux modèles couplés aux courbes peu pentues. De plus, l'utilisation de la supervision faible ne permet pas de corriger la distribution des unités et son impact y est plutôt limité.

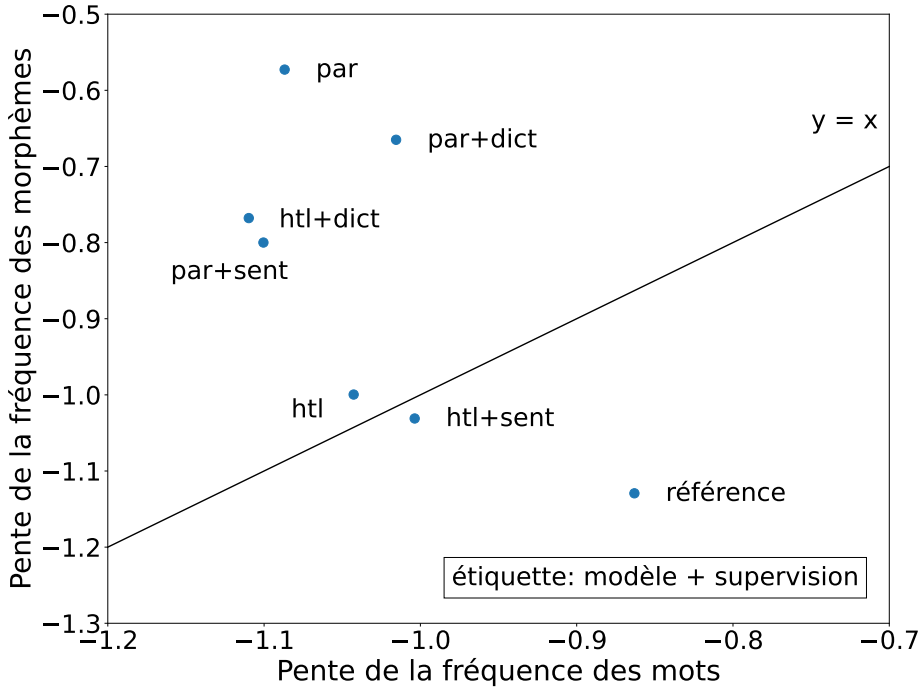


FIGURE 5.10 – Pentes de la fréquence des morphèmes et des mots dans le texte de référence et les segmentations automatiques en tsez. Les modèles couplés (`parallel-w`) sont indiqués par le préfixe « par » et les modèles hiérarchiques (`hier-final`) par « htl » ; la méthode de supervision est notée par le signe +.

Nous observons donc que les distributions réelles dans le texte et celles qui sont supposées par nos modèles ne correspondent pas nécessairement ; la concentration des occurrences dans nos corpus ne semble pas refléter un ajustement à une loi de puissance, en pratique. Cette divergence permet d'expliquer en partie les difficultés rencontrées par nos modèles, qui essaient d'identifier une telle distribution d'unités dans les données. Par ailleurs, l'échantillonnage des morphèmes semble être plus pertinent au niveau des types de mots (et non des occurrences), comme nous l'indiquent les coefficients des pentes. Nous retrouvons là une des observations de Virpioja et al. (2011), pour qui l'utilisation des types de mots a permis d'obtenir une meilleure segmentation en morphèmes par rapport aux occurrences.

5.4.4 Comparaison des unités les plus fréquentes

Sans supervision Le tableau 5.5 présente les dix mots (à gauche) et morphèmes (sur les occurrences de mots ; à droite) les plus fréquents dans le corpus tsez de référence ainsi que dans les versions segmentées par `parallel-w` et `hier-final` sans supervision. Au niveau des mots, le premier constat est la différence manifeste des segments obtenus avec nos modèles et la référence ; seuls deux mots (courts) sont présents dans ces trois listes. De plus, les fréquences associées à ces unités suivent une autre distribution : le premier mot de la référence est seulement trois fois plus fréquent que le dixième, là où la segmentation aboutit à un changement d'ordre de grandeur (plus de six fois). Nous remarquons ici la répartition typique d'une loi de puissance que produisent nos modèles et qui aboutit notamment à la segmentation en des unités très courtes.

	mot						morphèmes					
texte	référence		parallel-w		hier-final		référence		parallel-w		hier-final	
rang	forme	fréq.	forme	fréq.	forme	fréq.	forme	fréq.	forme	fréq.	forme	fréq.
1	eƛi–n	299	n	1299	n	1068	**n**	5059	**n**	2797	**n**	4348
2	**ža**	267	a	707	a	495	**r**	1827	**a**	1389	**a**	2277
3	zow–n	260	r	407	ƛin	262	**a**	1755	**r**	785	**r**	1592
4	**di**	161	ƛin	371	**ža**	252	b	1650	**ƛin**	475	**ƛin**	929
5	mi	145	tow	285	r	237	**ƛin**	839	**s**	368	**s**	816
6	sis	136	s	273	s	189	**x**	755	ža	333	**bi**	737
7	sida	128	**ža**	265	**di**	174	**s**	748	**x**	320	**x**	692
8	kid	108	**di**	217	gon	166	y	673	tow	300	y	640
9	neła–a	102	x	194	tow	163	oq	494	**bi**	289	ay	583
10	ik'i–n	100	gon	189	neła	162	**bi**	468	di	283	b	543

TABLE 5.5 – Les dix mots (gauche) et morphèmes (droite) les plus fréquents dans le corpus tsez et les textes segmentés par `parallel-w` et `hier-final` sans supervision. Les unités en commun (par niveau de segmentation) sont en **gras**.

L'utilisation d'une approche hiérarchique plutôt que couplée permet de réduire sensiblement cet effet : les mots ont globalement une fréquence plus réduite (celle du segment « n » baisse d'environ 18 %, par exemple). Néanmoins, les unités restent insuffisamment longues et s'apparentent encore à des morphèmes, tels « neła ».

En ce qui concerne les morphèmes, nous observons une meilleure identification des unités : sept morphèmes parmi les dix plus fréquents sont correctement identifiés comme fréquents par nos modèles. Les fréquences associées sont, en revanche, significativement différentes entre ces deux approches. `parallel-w` ne prédit pas assez les morphèmes fréquents, tels que « n », dont il manque près de la moitié des occurrences par rapport à la référence. Il s'agit d'une tendance vérifiée de manière plus générale avec ce modèle. À l'inverse, `hier-final` obtient des fréquences plus proches de la réalité comparativement. Cet exemple corrobore le choix d'une modélisation des morphèmes à travers les types des mots, évoqué précédemment.

Ainsi, la préférence pour les unités courtes, associée à l'hypothèse d'une distribution suivant une loi de puissance (voir section précédente), semble condamner les modèles à favoriser des unités fréquentes et aisément identifiables dans le corpus, comme des mots ou morphèmes d'un à deux caractères.

Avec supervision Le tableau 5.6 s'intéresse au texte tsez de référence et les segmentations de `hier-final` supervisées. Nous remarquons que les deux méthodes de supervision se différencient clairement, en particulier au niveau des mots. De fait, l'utilisation d'annotations denses (`sentence`) permet d'aplatir davantage la distribution des mots, comme en témoigne la chute des fréquences dans l'ensemble. À l'inverse, dans la situation `dictionary`, le modèle ne parvient pas à identifier de meilleures unités et sa distribution ne semble pas substantiellement modifiée par rapport au cas non supervisé. Notons par ailleurs le mot « eƛi–n » correctement décomposé, d'une part, avec les phrases entièrement annotées et non segmenté en morphèmes, d'autre part, avec le dictionnaire. Dans la première situation, la segmentation en morphèmes est garantie par la méthode de supervision, ce qui n'est pas le cas avec le dictionnaire, dans l'implémentation actuelle.

Au niveau des morphèmes, les unités fréquentes sont plus similaires. Les fréquences sont maintenant plus élevées avec `sentence` par rapport au cas sans supervision, se rapprochant plutôt

	mot						morphèmes					
texte	référence		`hier+sent`		`hier+dict`		référence		`hier+sent`		`hier+dict`	
rang	forme	fréq.	forme	fréq.	forme	fréq.	forme	fréq.	forme	fréq.	forme	fréq.
1	eχi–n	299	n	750	n	1144	**n**	5059	**n**	4409	**n**	4107
2	**ža**	267	a	374	a	499	**r**	1827	**a**	2193	**a**	2145
3	zow–n	260	**ža**	259	χin	318	**a**	1755	**r**	1643	**r**	1006
4	**di**	161	χin	211	r	276	b	1650	b	1286	**s**	760
5	mi	145	r	200	**ža**	261	**χin**	839	**χin**	952	**χin**	742
6	sis	136	**di**	162	tow	215	**x**	755	**s**	842	**x**	652
7	sida	128	eχi–n	161	eχin	209	**s**	748	y	782	neła	427
8	kid	108	gon	148	gon	194	y	673	**bi**	631	q	397
9	neła–a	102	s	135	**di**	192	oq	494	**x**	573	ru	391
10	ik'i–n	100	neła	121	s	173	**bi**	468	ay	495	**bi**	363

TABLE 5.6 – Les dix mots (gauche) et morphèmes (droite) les plus fréquents dans le corpus tsez et les textes segmentés par `hier-final` (`hier`) avec les supervisions faibles `sentence` (`+sent`) et `dictionary` (`+dict`). Les unités en commun (par niveau de segmentation) sont en **gras**.

des références, là où `dictionary` les rend plus basses. De plus, les segmentations denses permettent d'identifier deux morphèmes corrects supplémentaires : « b » et « y ». Pour cette catégorie de modèles, la supervision `sentence` apparaît plus appropriée, en garantissant la cohérence des décompositions morphologiques entre les deux niveaux.

Néanmoins, la supervision faible ne semble pas apporter de changements majeurs pour les unités identifiées, comme observé en section 5.4.3. Le modèle génère toujours des mots trop courts (et beaucoup trop fréquents), tandis que les morphèmes sont, au contraire, insuffisamment fréquents. Les tendances héritées de `dpseg` ne sont donc que partiellement compensées.

5.5 Conclusion

Ce chapitre s'est intéressé à la segmentation simultanée en mots et morphèmes, une tâche qui, dans l'absolu, est relativement artificielle, car l'annotation linguistique l'aborde toujours de manière séquentielle, plutôt que simultanément. Cependant, elle part du constat précédent que le modèle `dpseg` aboutit à des unités segmentés entre ces deux niveaux, dû au phénomène de sur-segmentation. Cette approche nous a alors permis d'observer les comportements des modèles statistiques, afin d'évaluer, notamment, s'il est possible d'améliorer la segmentation en mots, qui reste notre objectif ici, par une meilleure représentation des mots et des morphèmes.

Plusieurs modèles ont été conçus à cette fin : une approche simple en deux temps, `pipeline`, deux modèles couplés où `dpseg` est utilisé pour chaque niveau en maintenant la cohérence à travers une règle et trois variantes hiérarchiques qui prennent en compte la structure des mots, en les décomposant comme une suite de morphèmes. Nous avons également eu recours à une supervision faible grâce à des ressources disponibles dans le cadre de la documentation automatique. En nous appuyant sur les résultats du chapitre précédent, nous avons testé deux configurations, avec des phrases déjà annotées (`sentence`) et un dictionnaire d'unités (`dictionary`).

Nous avons observé dans l'ensemble une amélioration des performances avec ces modèles pour deux langues, le japhug et le tsez. L'approche `pipeline` s'est avérée peu intéressante du fait de sa propension à répercuter les erreurs des mots vers les morphèmes, conduisant à des

résultats médiocres notamment dans l'identification des listes de mots apparaissant dans le corpus (faible score LF). La combinaison de deux modèles mis en parallèle est une option intuitive pour obtenir une segmentation à deux niveaux décente. Enfin, les méthodes hiérarchiques permettent de mieux modéliser les deux types d'unités, en intégrant dans le modèle la structure des mots comme succession de morphèmes. Elle permet alors de pallier partiellement les faiblesses du modèle `dpseg` comme la sur-segmentation, en générant des unités plus longues, ce qui différencie davantage les deux types d'unités.

En outre, l'intégration de la supervision faible entraîne une amélioration des performances : la méthode `sentence` parvient à augmenter de manière systématique et stable les métriques, là où le recours à un dictionnaire impacte de manière limitée les modèles, en particulier hiérarchiques.

Nous avons néanmoins décelé, tout au long des expériences, une sorte de « plafond de verre » dans la segmentation et la distinction des mots et des morphèmes par les modèles purement statistiques, et ce, même en présence d'information de supervision. Les hausses des scores sont modestes dans l'absolu ; les faiblesses inhérentes au modèle `dpseg` unigramme, à la base de nos modèles de segmentation jointe, ne sont que partiellement traitées. En effet, le phénomène de sur-segmentation, qui est marqué au niveau des mots, persiste et les unités les plus rares sont difficilement identifiées. Les autres modèles évalués sont également concernés par cette limite ; il semble ainsi complexe de distinguer clairement les mots des morphèmes uniquement à l'aide de statistiques simples.

Par ailleurs, afin de mieux comprendre les raisons derrière ces difficultés rencontrées notamment par nos modèles bayésiens non paramétriques, nous avons analysé les distributions des unités dans les corpus étudiés. Nous avons alors constaté que les textes recueillis dans le cadre de la documentation des langues présentent une distribution particulière des mots (et morphèmes), avec une concentration différente dans la fréquence des mots les plus souvent usités. Ceci semble tenir de la nature même des corpus et de l'objectif ; les contes ou les histoires contiennent davantage de répétitions. Nous pouvons rapprocher ce constat de l'*IGT Bias* (Lewis et Xia, 2008), où les exemples sont souvent choisis par les linguistes pour illustrer des phénomènes grammaticaux particuliers (comme pour les phrases japhug issues du livre de grammaire), menant à leur sur-représentation dans le corpus. Du fait de ce biais dans les données, cette concentration se trouve renforcée par nos modèles `dpseg` qui se basent sur l'hypothèse d'une distribution des unités en loi de puissance, qui n'est pas observée dans les corpus étudiés.

Enfin, ces deux derniers chapitres soulèvent la question de la définition d'un mot ou d'un morphème. Est-ce une erreur majeure si le modèle considère « parceque » comme un seul mot ? Si nous nous contentons dans cette thèse d'essayer de reproduire les décisions observées dans les données d'apprentissage, qui reflètent les intuitions linguistiques employées dans chaque corpus, étant donné que nos outils sont destinés à alléger au mieux l'annotation, nous ne cherchons pas à répondre à cette question, dépassant largement le cadre de cette thèse et toujours d'actualité, comme en témoigne (Baldwin et al., 2021) (groupe de travail 1).

Théoriquement et idéalement, après l'étape de segmentation en mots ou en morphèmes, la phrase source est entièrement annotée. En supposant que nous avons également à disposition la phrase traduite dans une langue plus dotée, nous pouvons alors générer les gloses interlinéaires pour chaque morphème source, comme nous allons nous y attacher au chapitre suivant.

Chapitre 6

La génération automatique des gloses

6.1 Introduction

Ce chapitre aborde une tâche de génération automatique de gloses, où l'objectif, comme illustré par la figure 6.1, est de prédire, à partir de la phrase dans la langue source étudiée, segmentée en mots et morphèmes (**S3**), et sa traduction dans une langue plus dotée (**S5**), les gloses associées à chacun de ses morphèmes (**S4**). Nous rappelons donc ici la correspondance exacte entre les morphèmes source et les gloses, résultant de la nature même de ces annotations linguistiques.

Entrée **S3**	Phrase source segmentée	kɤndʑi–xtɤɣ	χsɯm	pjɤ–tu–nɯ
Entrée **S5**	Traduction (EN)	*There were three brothers*		
Sortie **S4**	Phrase glosée	COLL–brother	three	IFR.IPFV–exist–PL

FIGURE 6.1 – La tâche de génération automatique de gloses pour la phrase de l'exemple 2.1.

Cette figure correspond en réalité au format *Interlinear Glossed Text* (IGT), le plus répandu pour représenter les phrases glosées. Nous faisons le choix de garder les mêmes conventions typographiques que dans la section 2.1.1 pour faciliter la distinction entre les différentes unités. En particulier, les gloses lexicales seront indiquées par une police spécifique (comme three dans l'exemple 6.1), là où les mots issus de la phrase traduite seront en italique (comme *three*).

Comme détaillé en section 2.3.2, l'obtention des gloses est coûteuse car elle nécessite une expertise linguistique et se fait principalement manuellement. Comparativement aux paires de phrases source et cible (**S3** et **S5**), ces annotations sont de ce fait disponibles en quantité plus limitée. Par l'automatisation, l'idée est d'alléger la part répétitive qui existe lors de l'annotation d'un corpus, qui ne présente pas systématiquement des phénomènes linguistiques complexes du début à la fin. Ces derniers seront laissés aux annotateurs, qui vérifieraient également les gloses générées par le modèle.

Dans cette optique d'annotation semi-automatique des données afin d'alléger le processus, nous considérons cette tâche comme un étiquetage de séquences, dans la continuité des travaux antérieurs. Face au défi majeur posé par la variété des gloses lexicales, un problème traité de diverses manières jusque là, nous proposons une approche reposant sur leur alignement avec les mots de la traduction. Ceci nous permet non seulement de réduire les choix lexicaux proposés au modèle pour plus de pertinence, mais également de pouvoir prédire des gloses lexicales non observées à l'entraînement. Nous étudions donc l'apport d'un tel modèle notamment pour cinq langues, à différentes étapes de la documentation.

Ce chapitre aborde les points suivants :

— l'estimation de la difficulté de la génération automatique de gloses vis-à-vis de la tâche d'étiquetage en parties du discours (PoS), similaire et plus connue ;

— une nouvelle approche de génération de gloses reposant sur le réemploi d'un modèle statistique d'étiquetage, basé sur les CRF, supervisé par des alignements automatiques entre les gloses lexicales et les mots de la traduction lors de l'apprentissage ;
— l'application de notre modèle sur cinq langues peu dotées du défi partagé SIGMORPHON 2023 sur la génération de gloses, dans des conditions de supervision variées ;
— une étude des bénéfices d'un pré-entraînement multilingue dans le cadre de notre tâche, en utilisant des corpus glosés.

Nous commençons tout d'abord en section 6.2 par un avant-propos afin de situer la tâche de génération de gloses à travers quelques simplifications, pour ensuite exposer notre approche d'étiquetage. La section 6.3 détaille les différents choix effectués vis-à-vis de l'alignement entre les gloses lexicales et les mots de la traduction, premier pilier de notre méthode ; le second est décrit en section 6.4, à savoir la ré-utilisation d'un modèle de traduction statistique, Lost, fondé sur la théorie des CRF, dont nous exposerons également les configurations de ses fonctions caractéristiques. Puis, la section 6.5 présente les expériences effectuées, notamment sur les langues du défi partagé SIGMORPHON sur la génération automatique de gloses. Enfin, la section 6.6 considère différentes configurations de pré-entraînement de notre modèle en utilisant des ressources additionnelles. Les résultats de (Okabe et Yvon, 2022b), (Okabe et Yvon, 2023c) et (Okabe et Yvon, 2023b) servent de base à ce chapitre.

6.2 Génération de gloses comme étiquetage de séquences

Nous abordons pas à pas la tâche de génération automatique de gloses, vue comme un étiquetage de séquences. Nous commençons par un cas relativement aisé d'identification de la nature des morphèmes, pour ensuite complexifier en introduisant toutes les gloses grammaticales ; nous retrouvons alors les méthodes de travaux antérieurs. Enfin, nous présentons notre approche en détaillant les motifs derrière nos choix pour répondre au défi posé par la variété des gloses lexicales existantes.

6.2.1 Tâche préliminaire : classification binaire

Avant d'aborder plus en détail la génération de gloses, nous effectuons tout d'abord une expérience plus simple consistant à identifier le type de gloses, grammaticale (GRAM) ou lexicale (LEX), afin d'avoir un premier aperçu de la difficulté de la tâche. Dans ce cas, nous faisons donc face à une simple classification binaire : la modèle doit prédire pour chaque morphème source une des deux étiquettes, comme illustré par la figure 6.2. Grâce à la correspondance un pour un des étiquettes avec les morphèmes, un modèle d'étiquetage de séquences classique tel que le CRF (Lafferty et al., 2001) convient alors.

Entrée **S3**	Phrase source segmentée	kʏndʑi–xtʏɣ	χsɯm	pjʏ–tu–nɯ
Sortie **S4'**	Étiquettes binaires de gloses	GRAM–LEX	LEX	GRAM–LEX–GRAM

FIGURE 6.2 – Prédiction du type de gloses pour la phrase de l'exemple 6.1.

Nous rappelons que le CRF modélise la probabilité des étiquettes $\boldsymbol{y}$, pour une séquence source $\boldsymbol{x}$, par l'équation (6.1) (voir section A.2 pour les détails).

$$p_\theta(\boldsymbol{y}|\boldsymbol{x}) = \frac{\exp\left\{\sum_{k=1}^{K} \theta_k G_k(\boldsymbol{x}, \boldsymbol{y})\right\}}{\sum_{\boldsymbol{y'} \in \mathcal{Y}^T} \exp\left\{\sum_{k=1}^{K} \theta_k G_k(\boldsymbol{x}, \boldsymbol{y'})\right\}} = \frac{1}{Z_\theta(\boldsymbol{x})} \exp\left\{\sum_{k=1}^{K} \theta_k G_k(\boldsymbol{x}, \boldsymbol{y})\right\} \tag{6.1}$$

Dans un cadre de classification binaire, l'ensemble des étiquettes possibles, $\mathcal{Y}$, revient à $\mathcal{Y} = \{\text{LEX}, \text{GRAM}\}$. De plus, nous définissons *l'espace de recherche* comme l'ensemble des étiquetages possibles pour une phrase ; ici, pour une séquence de T morphèmes sources, il correspond simplement à $\mathcal{Y}^T$.

Expérience en tsez Nous entraînons le CRF sur une proportion variable du corpus tsez, présenté en section 3.2.3, allant de 100 à 1 400 phrases (soit l'intégralité du corpus disponible pour la supervision), afin de prédire ces deux étiquettes. Nous gardons 100 phrases pour le développement et évaluons sur les 500 phrases restantes. Nous choisissons d'utiliser comme métrique pour cette expérience l'exactitude, à savoir la proportion d'étiquettes correctement prédites sur l'ensemble des morphèmes considérés.

En pratique, nous utilisons Wapiti, une implémentation des CRF décrite en section 5.2.6, avec comme seule entrée, là aussi, le morphème source ; les caractéristiques unigrammes et bigrammes sont également appliquées sur une fenêtre de cinq morphèmes, c'est-à-dire deux positions avant et après le morphème courant. Les séparateurs de morphèmes (« – ») sont traités comme unités spécifiques à prédire, afin de conserver la distinction avec les frontières de mots. Leur présence permet notamment d'obtenir de meilleurs résultats, selon nos expériences préliminaires.

Phrases d'entraînement	100	200	500	1 000	1 400
Exactitude	96,8	97,2	98,1	98,3	98,5

TABLE 6.1 – Évolution de l'exactitude obtenue par la classification binaire des gloses du corpus tsez pour différentes tailles de données d'entraînement.

Comme présenté dans le tableau 6.1, nous obtenons dès 100 phrases, un taux d'exactitude systématiquement supérieur à 96 %. Nous constatons donc que cette tâche est relativement aisée, suggérant une différenciation claire des gloses grammaticales et lexicales, à travers la forme des morphèmes et, indirectement, leurs positions. En revanche, une autre façon de voir ces résultats est de remarquer que l'utilisation de 14 fois plus de phrases d'entraînement n'aboutit qu'à moins de 2 points d'amélioration de l'exactitude. Il existe donc une part non négligeable de morphèmes pour lesquels la nature est incertaine, malgré la supervision.

étiquette	LEX	GRAM
LEX	4 776	117
GRAM	221	5 476

TABLE 6.2 – Matrice de confusion obtenue avec 100 phrases d'entraînement (référence en ligne, prédiction en colonne).

Le tableau 6.2 présente la matrice de confusion associée en guise de complément. Nous observons que les erreurs sont davantage dues aux gloses grammaticales prédites comme étant lexicales que l'inverse. Cette tendance peut notamment s'expliquer par les formes de morphèmes pouvant avoir plusieurs étiquettes, grammaticales et lexicales, en fonction du contexte.

6.2.2 Approche en cascade reposant sur un ensemble fini d'étiquettes

Si l'identification du type de gloses semble accessible avec peu de données, il en est autrement pour la générations des gloses en elles-mêmes. Une première approche serait de reproduire la méthodologie de Moeller et Hulden (2018) où l'on prédit soit la glose grammaticale, soit une étiquette unique pour toutes les gloses lexicales, « stem », afin d'avoir un ensemble fini d'étiquettes, comme illustré en figure 6.3. La prédiction de la glose lexicale se ferait alors dans un second temps, soit manuellement, soit à travers un dictionnaire (Samardžić et al., 2015). Il s'agit en effet d'une fonctionnalité présente dans certains outils d'annotation utilisés par les linguistes tels que ELAN-CorpA (CNRS-LLACAN, 2023) ou *FieldWorks Language Explorer* (FLEx) (Rogers, 2010) ; les étiquettes précédemment observées avec un morphème donné sont proposées comme candidats de gloses.

Entrée **S3**	Phrase source segmentée	kʏndʑi–xtʏɣ	χsɯm	pjʏ–tu–nɯ
Sortie **S4'**	Étiquettes finies de gloses	COLL–stem	stem	IFR.IPFV–stem–PL

FIGURE 6.3 – Une première approche de prédiction de gloses pour la phrase de l'exemple 6.1.

Rappelons que le coût du calcul de Z_θ dans l'équation (6.1) augmente de manière quadratique par rapport au nombre d'étiquettes. Bien que cette approche accroisse beaucoup le nombre total d'étiquettes possibles, en pratique, l'exécution des CRF reste réalisable : à titre d'exemple, pour le corpus tsez étudié, l'ensemble des gloses grammaticales représente 158 étiquettes (le CRF aura donc 159 étiquettes à traiter, avec « stem »). De fait, ces modèles sont capable de gérer jusqu'à plusieurs centaines d'étiquettes (Mueller et al., 2013 ; Lavergne et Yvon, 2017).

Formellement, si nous notons $\mathcal{Y}_G$, l'ensemble (fini) des gloses grammaticales pour une langue donnée, l'ensemble des étiquettes pour cette expérience revient à $\mathcal{Y} = \mathcal{Y}_G \cup \{\text{STEM}\}$. L'espace de recherche, quant à lui, sera également $\mathcal{Y}^T$, pour T morphèmes source.

D'un point de vue de l'implémentation, nous utilisons à nouveau Wapiti, avec la configuration détaillée en section 6.2.1. Nous nommerons par la suite `CRF`, cette méthode d'étiquetage prédisant des gloses grammaticales ou une étiquette lexicale à compléter.

Nous comparons la méthode `CRF` avec deux approches simples :

— `stem` affecte systématiquement l'étiquette « stem », la plus fréquente, à tous les morphèmes ; son score n'est donc pas affecté par la taille des données d'apprentissage.

— `maj` repose sur un lexique appris sur les phrases d'entraînement et prédit l'étiquette majoritaire (« stem » inclus) observée pour tout morphème source ; il s'agit donc d'une approximation de la fonctionnalité déjà implémentée dans certains outils d'annotation et basée sur un dictionnaire.

Le tableau 6.3 présente les valeurs de l'exactitude en fonction de la taille du corpus d'apprentissage en tsez, de 200 à 1 600 phrases. Deux séries distinctes de 200 phrases sont utilisées pour le développement et le test. Nous reportons les résultats moyennés de deux lancers.

Nous observons tout d'abord qu'environ 47 % des morphèmes sont lexicaux dans la référence, comme en témoigne le score de l'approche `stem`. Nous remarquons, par ailleurs, qu'affec-

Taille entraînement	200	500	800	1000	1300	1600
`stem`	47,0 (0,6)	47,0 (0,6)	47,0 (0,6)	47,0 (0,6)	47,0 (0,6)	47,0 (0,6)
`maj`	83,6 (0,0)	84,0 (0,2)	84,0 (0,5)	84,1 (0,4)	84,1 (0,4)	84,2 (0,4)
`CRF`	**84,3** (1,7)	**89,7** (1,2)	**90,8** (0,3)	**91,6** (0,3)	**92,1** (0,4)	**92,8** (-)
Δ (`CRF - maj`)	+ 0,7	+ 5,7	+ 6,8	+ 7,5	+ 8,0	+ 8,6

TABLE 6.3 – Évolution de l'exactitude pour un ensemble fini d'étiquettes, en fonction du nombre de phrases dans les données d'entraînement du corpus tsez (moyenne de deux lancers (écart type)).

ter l'étiquette majoritaire s'avère être une bonne base pour cette tâche, avec 83 % d'exactitude dès 200 phrases de supervision. En revanche, notons que l'évolution du score en accroissant les données d'entraînement est assez décevante : un gain de moins d'un point pour 8 fois plus de phrases. L'approche `maj` semble donc avoir vu la plupart des étiquettes grammaticales assez tôt et les morphèmes lexicaux, plus variés, posent moins problème ici, étant donné qu'ils sont tous regroupés sous l'étiquette stem.

À l'inverse, si les performances du modèle `CRF` semblent relativement proches de `maj` lorsque peu de données sont disponibles, l'écart se creuse, lorsqu'une plus large proportion du corpus est utilisée (voir ligne Δ). À ce titre, le seuil de 500 phrases correspond à un gain non négligeable dans les performances et le `CRF` parvient à mieux mettre à profit les données supplémentaires, sans plafonner comme `maj`.

6.2.3 Complexité de la tâche par rapport à l'étiquetage en PoS

Nous pouvons positionner cette première méthode par rapport à une tâche similaire et plus connue de TAL : l'étiquetage en parties du discours (PoS), une analogie à la base de l'approche de Samardžić et al. (2015) notamment. Malgré leurs similitudes, la principale distinction vient de la quantité et la distribution des étiquettes : il y un nombre beaucoup plus important de gloses possibles, inégalement réparties entre elles [1]. Dans cette section, nous allons évaluer dans quelle mesure cette différence impacte la difficulté de notre tâche par rapport à l'étiquetage en PoS.

Pour effectuer cette comparaison, nous étudions le zaar : en effet, les phrases de ce corpus (Caron, 2015) sont non seulement glosées mais contiennent aussi les annotations en PoS. Pour rappel, il y a 190 gloses grammaticales (et une étiquette « stem » pour regrouper toutes les gloses lexicales) pour 29 étiquettes PoS (voir tableau 3.1).

En utilisant la même méthodologie que dans la section 6.2.2, mais cette fois sur le corpus zaar, nous obtenons les exactitudes moyennes de deux lancers pour ces deux tâches, présentées dans le tableau 6.4. La seule différence réside dans les jeux d'étiquettes.

Tout d'abord, le corpus zaar paraît relativement aisé à annoter, au vu des meilleures performances du modèle de base `maj`, et ce, pour les deux tâches. Si nous comparons maintenant ce modèle par rapport aux performances obtenues par le modèle `CRF`, il obtient de meilleures performances ; l'approche plus simple, `maj`, apparaît comme plus appropriée pour le corpus zaar. La taille du corpus semble être une première explication du résultat, avec presque autant de types de morphèmes (environ 1 500) que le tsez pour plus du double d'occurrences (17 000 pour le zaar contre 40 000 pour le tsez ; voir tableau 3.1). Il n'y aurait donc, comparativement, pas encore assez

1. Il y a par exemple 17 étiquettes en PoS universelles dans l'*Universal Dependencies* ; `https://universaldependencies.org/u/pos/all.html`.

Taille	207 PoS / Gloses	507 PoS / Gloses	807 PoS / Gloses	1007 PoS / Gloses	1307 PoS / Gloses
`stem`	- / 44,7 (1,8)	- / 44,7 (1,8)	- / 44,7 (1,8)	- / 44,7 (1,8)	- / 44,7 (1,8)
`maj`	**71,6** (12,7) / **83,7** (9,8)	**80,5** (5,7) / **87,9** (5,1)	**83,9** (4,0) / **88,8** (4,5)	**85,1** (3,6) / **89,0** (4,2)	86,2 (3,7) / **91,1** (1,2)
CRF	61,8 (8,6) / 67,9 (0,1)	77,6 (4,7) / 79,0 (2,6)	82,2 (2,4) / 82,3 (2,3)	83,1 (4,0) / 83,6 (3,4)	**86,2** (2,3) / 85,3 (-)

TABLE 6.4 – Comparaison de l'évolution de l'exactitude pour les tâches de génération de gloses et d'étiquetage en PoS en fonction du nombre de phrases dans les données d'entraînement du corpus zaar (moyenne de deux lancers (écart type)).

de données de supervision pour le modèle `CRF`. Néanmoins, notons que l'accroissement du nombre de phrases en entraînement permet d'améliorer plus rapidement notre modèle que `maj`, soulignant là encore l'efficacité de l'approche statistique par rapport à une méthode basée sur un lexique.

Pour revenir à notre question initiale, nous remarquons pour le modèle `CRF` que les valeurs des exactitudes pour ces deux tâches sont assez proches, voire légèrement plus hautes par moment pour la génération de gloses. Il s'avère ainsi que les tâches de génération de gloses et d'étiquetage en PoS sont de complexité comparable pour ce corpus, malgré les disparités dans les étiquettes de gloses.

6.2.4 La définition de notre approche

Contrairement à l'approche évoquée ci-dessus en section 6.2.2, nous souhaitons également générer automatiquement les gloses lexicales. Se pose alors le défi principal de la tâche : l'ensemble des gloses lexicales n'est pas défini a priori, contrairement aux gloses grammaticales. De fait, dans la plupart des cas, les étiquettes lexicales observées à l'entraînement ne suffisent pas pour prédire celles qui apparaîtront à l'inférence.

Par ailleurs, nous aimerions avoir une approche de bout en bout, au lieu d'une configuration en cascade comme (Samardžić et al., 2015; Moeller et Hulden, 2018). Cela permettrait notamment d'éviter les éventuelles propagations d'erreurs lors de l'assignation des gloses lexicales ; si un morphème est incorrectement annoté par une glose grammaticale, il ne peut pas être corrigé à la seconde étape, et, à l'inverse, s'il est étiqueté par « stem », un lemme dans la langue de documentation lui sera associé, bien qu'il soit grammatical.

Pour ce faire, une méthode serait alors d'abandonner entièrement la prédiction des gloses lexicales inconnues pour les laisser à une annotation manuelle et ne considérer que les étiquettes déjà observées, à la manière de l'étiquetage en PoS. Au-delà de l'impossibilité de traiter les morphèmes non observés à l'entraînement, cette méthode pose toutefois un problème majeur de calcul : pour les ordres de grandeur correspondant à la variété des gloses lexicales, à savoir au-delà de quelques milliers d'étiquettes, le calcul de la fonction de partition Z_θ dans l'équation (6.1) devient trop coûteux et ne devient plus réalisable. Par exemple, Zhao et al. (2020) rapportent que le modèle de McMillan-Major (2020), basé sur des CRF, ne parvenait pas à être entraîné en temps raisonnable (moins de la moitié du corpus était traité en une semaine) sur l'entièreté d'un corpus de 25 000 phrases.

Nous cherchons donc à surmonter ces deux points noirs : le traitement des étiquettes inconnues à l'entraînement ainsi qu'un coût de calcul raisonnable. Notre idée repose alors sur l'ajout d'étiquettes lexicales adaptées pour chaque phrase traitée : il s'agirait alors d'avoir un ensemble d'étiquettes $\mathcal{Y}$ dynamique, décomposable en une partie commune pour tout le corpus, à savoir

les gloses grammaticales en quantité finie $\mathcal{Y}_G$, ainsi qu'un ensemble local pour traiter les gloses lexicales.

Afin de proposer des étiquettes pertinentes, nous considérons l'hypothèse [H] suivante, stipulant que les gloses lexicales puissent être identifiées à travers les mots de la traduction. En réalité, il s'agit d'une supposition implicite déjà considérée par les modèles des travaux antérieurs tels que (McMillan-Major, 2020) ou (Zhao et al., 2020), où la phrase dans la langue cible était utilisée comme entrée supplémentaire.

En outre, dans un autre contexte, Georgi (2016) compare une approche heuristique et des méthodes d'alignement statistiques entre gloses (même grammaticales) et traductions. Les résultats obtenus par cette expérience étaient suffisamment positifs pour suggérer la validité de notre hypothèse, bien que nous la vérifions en section 6.3.3 pour nos données.

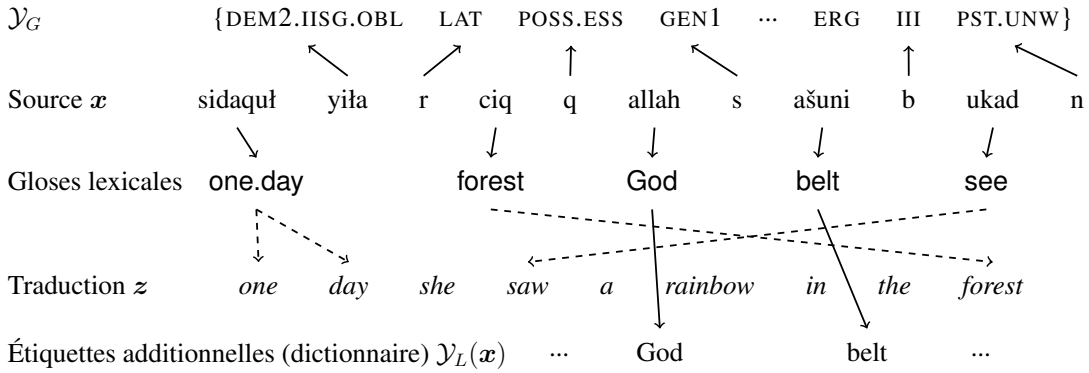


FIGURE 6.4 – Illustration de notre approche de génération de gloses pour une phrase d'exemple en tsez. Les flèches en trait plein pour les gloses grammaticales $\mathcal{Y}_G$ et les mots supplémentaires (issus d'un dictionnaire) $\mathcal{Y}_L(\boldsymbol{x})$ correspondent à un étiquetage classique, là où les pointillés indiquent un alignement avec la phrase de traduction $\boldsymbol{z}$. Notons que les étiquettes lexicales peuvent provenir de la traduction mais aussi du dictionnaire.

La figure 6.4 illustre l'intuition derrière notre approche pour une phrase d'exemple. Un morphème donné peut, d'une part, être étiqueté par une des gloses grammaticales de $\mathcal{Y}_G$, un ensemble fini et commun à tout le texte, à la manière de ce que nous avons vu dans les sections précédentes. D'autre part, pour les gloses lexicales, nous pouvons nous appuyer sur les mots de la traduction (ou plus exactement, les lemmes qui leur sont associés), d'après l'hypothèse [H], comme c'est le cas de « ukad » (dont la glose de référence est see) et le mot anglais aligné *saw*. Enfin, il peut nous rester des mots non retrouvables à travers la traduction, comme God ou belt (qui semblent ici correspondre à *rainbow* en anglais). Pour pouvoir traiter ces cas, nous avons recours à un dictionnaire qui recense, pour chaque morphème source déjà observé, sa glose lexicale la plus fréquemment associée. Ceci correspond en quelque sorte à l'approche `maj` de la section 6.2.2, mais restreinte aux gloses lexicales. Nous noterons cet ensemble de gloses lexicales supplémentaires, dépendant de la phrase source $\boldsymbol{x}$, $\mathcal{Y}_L(\boldsymbol{x})$.

Formellement, notre approche hybride prédit une étiquette y_t depuis l'ensemble $\mathcal{Y}(\boldsymbol{x}, \boldsymbol{z}) = \mathcal{Y}_G \cup \{1, \ldots, |\boldsymbol{z}|\} \cup \mathcal{Y}_L(\boldsymbol{x})$ pour une paire de phrases source-cible ($\boldsymbol{x}$, $\boldsymbol{z}$). L'équation (6.2) définit comment obtenir la glose réelle $\tilde{y_t}$ à travers la transformation déterministe ϕ, permettant de convertir les indices d'alignement vers une étiquette lexicale.

$$\tilde{y_t} = \phi(y_t) = \begin{cases} y_t & \text{, si } y_t \in \mathcal{Y}_G \cup \mathcal{Y}_L(\boldsymbol{x}) \\ z_i & \text{, si } y_t = i \in \{1, \ldots, |\boldsymbol{z}|\} \end{cases} \tag{6.2}$$

L'entraînement de ce modèle est plus complexe que pour un CRF standard car nous observons les gloses réelles $\tilde{y_t}$ et non y_t. Or, comme évoqué dans la figure 6.4, les gloses lexicales peuvent provenir de deux sources : les mots de la traduction ainsi que le lexique complémentaire ($\mathcal{Y}_L(\boldsymbol{x})$). Pour traiter cette possible ambiguïté, nous introduisons donc une variable latente, $\boldsymbol{o}$, qui indique l'origine de chaque étiquette prédite y_t : soit la glose lexicale provient du mot z_{o_t} de la traduction et $o_t > 0$, soit y_t est issu de $\mathcal{Y}_G \cup \mathcal{Y}_L(\boldsymbol{x})$ et $o_t = 0$. Le modèle que nous considérons est alors défini par l'équation (6.3).

$$p_{\boldsymbol{\theta}}(\boldsymbol{y}, \boldsymbol{o} | \boldsymbol{x}, \boldsymbol{z}) = \frac{1}{Z_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{z})} \exp\left\{\boldsymbol{\theta}^T \boldsymbol{G}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}, \boldsymbol{o})\right\} \tag{6.3}$$

En pratique, nous n'observons pas entièrement cette nouvelle variable ; c'est pourquoi, nous avons recours aux alignements automatiques $\boldsymbol{a}$ comme approximation de cette information. Pour un morphème donné x_t, cette variable n'est pas active ($a_t = 0$) si la glose est grammaticale ou lexicale et non alignée (c'est-à-dire, issu du dictionnaire). Sinon, l'étiquette provient de la phrase de traduction, donc la variable correspond à l'indice du mot cible aligné ($a_t > 0$). Le tableau 6.5 récapitule les différentes provenances d'étiquettes de gloses ainsi que les valeurs de $\boldsymbol{a}$ et $\boldsymbol{o}$ associées.

	y_t	a_t	o_t
glose grammaticale	$\in \mathcal{Y}_G$	$= 0$	0
glose lexicale du dictionnaire	$\in \mathcal{Y}_L(\boldsymbol{x})$	$= 0$	0
glose lexicale de la traduction	$\in \{1, \ldots, \lvert\boldsymbol{z}\rvert\}$	> 0	a_t
glose de référence à l'entraînement*	$\notin \mathcal{Y}_L(\boldsymbol{x})$	$= 0$	0

TABLE 6.5 – Provenance des gloses et correspondance avec les valeurs des variables $\boldsymbol{a}$ et $\boldsymbol{o}$ de notre modèle. *À l'entraînement, si la glose de référence n'est présente ni dans le dictionnaire, ni dans la traduction, nous l'ajoutons à l'ensemble des étiquettes possibles pour maintenir l'atteignabilité de la référence.

Notre approche repose donc sur l'équation (6.4) suivante :

$$p_{\boldsymbol{\theta}}(\boldsymbol{y}, \boldsymbol{a} | \boldsymbol{x}, \boldsymbol{z}) = \frac{1}{Z_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{z})} \exp\left\{\boldsymbol{\theta}^T \boldsymbol{G}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}, \boldsymbol{a})\right\} \tag{6.4}$$

6.3 Alignement entre gloses lexicales et traduction

Le premier pilier de notre approche consiste en l'utilisation d'alignements automatiques, pour combler le manque de liens de référence entre les gloses lexicales et les mots de la traduction lors de l'entraînement. Notons qu'il s'agit d'un alignement monolingue donc relativement simple. Nous détaillons ici le modèle utilisé et l'analysons afin d'estimer s'il convient à nos besoins : en particulier, nous vérifions la validité de l'hypothèse [H] énoncée en section 6.2.4.

6.3.1 SimAlign, un modèle d'alignement neuronal

Nous avons choisi le modèle d'alignement neuronal multilingue SimAlign (Jalili Sabet et al., 2020), pour ses meilleures performances par rapport aux modèles d'alignement statistiques plus répandus, basés sur les modèles IBM (Brown et al., 1993). Il présente l'avantage de pouvoir être directement utilisé, sans supervision par des données parallèles notamment. En effet, il repose sur le calcul de la similarité cosinus entre les plongements lexicaux des entités des phrase source et cible. À travers la matrice de similarité ainsi constituée, le modèle extrait des alignements en utilisant différentes heuristiques. SimAlign propose trois méthodes :

— Argmax, la plus intuitive, associe deux unités lorsqu'elles sont mutuellement les plus proches ; si la proximité n'est vérifiée que dans un sens, il n'y a alors pas d'alignement ;
— Itermax effectue, afin de pallier le caractère parcimonieux des alignements identifiés par la méthode précédente, plusieurs itérations d'Argmax, en réduisant à chaque fois la similarité des unités déjà alignées ;
— Match considère les alignements comme un problème de couplage (*matching* en anglais) en créant un graphe biparti avec les mots sources et cibles, où les arêtes sont pondérées par les similarités calculées. Ce type de problème peut être résolu en temps polynomial (comme l'algorithme de (Galil, 1986)).

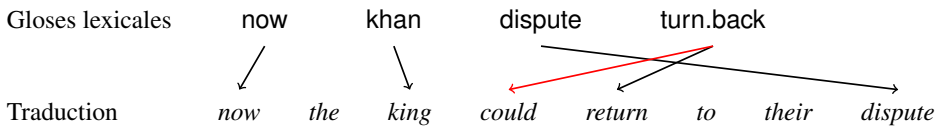
Notons ici que Argmax identifie des alignements locaux, qu'Itermax est par essence un algorithme glouton et que Match cherche l'optimum global. C'est pourquoi, le choix entre ces trois méthodes dépend des alignements souhaités, en d'autres termes, s'il faut favoriser la précision (Argmax) ou le rappel (Match), avec Itermax comme compromis intermédiaire.

Dans le cadre de notre approche, nous souhaitons idéalement aligner chaque glose lexicale avec un mot de la traduction, pour pouvoir les retrouver. Nous nous sommes donc intéressés plus particulièrement à deux de ces méthodes pour leurs propriétés respectives. La stratégie Argmax n'aligne pas nécessairement toutes les unités, mais est plus prudente et produit donc des liens plus fiables. À l'inverse, la méthode Match aboutit nécessairement à un appariement de toutes les unités sources [2].

Identifier les gloses lexicales pour l'alignement Nous effectuons quelques pré-traitements avant d'aligner les gloses avec les mots de la traduction. Les gloses lexicales composées telles que one.day dans la figure 6.4 sont décomposées pour aligner séparément « one » et « day » ; les mots alignés sont par la suite recombinés. Ce procédé permet d'obtenir de meilleurs liens d'alignement, d'un point de vue qualitatif, dans une expérience préliminaire sur un jeu restreint de phrases. Cela autorise également en pratique les associations d'une glose à plusieurs mots cibles (*one-to-many*). De même, dans le cas des gloses mixtes composées comme he.OBL, nous alignons uniquement la partie lexicale. Notons ici que les alignements de plusieurs gloses vers un seul [3] (*many-to-one*) ou plusieurs mots (*many-to-many*) sont ignorés dans notre approche.

Une fois ces traitements effectués, nous pouvons calculer les alignements entre, d'une part, les gloses lexicales et, d'autre part, les mots de la traduction avec SimAlign, comme illustré par la figure 6.5 pour une phrase tsez, avec les deux méthodes principalement considérées. La partie inférieure correspond aux gloses obtenues via alignement, afin de visualiser une version de la phrase où les gloses sont atteignables depuis la traduction.

2. Elle obtient exactement autant d'alignements que de gloses lexicales, uniquement si elles sont moins nombreuses que les mots dans la traduction.
3. Par exemple, les gloses widowed et woman alignées au mot *widow* de la traduction.



source	howži	xan	yizi–z	daʕba–ƛ'or	uti–ɬ–n
référence	now	**khan**	DEM2.IPL.OBL–GEN2	dispute–SUPER.LAT	**turn.back**–POT–PST.UNW
`Match`	now	**king**	DEM2.IPL.OBL–GEN2	dispute–SUPER.LAT	**could.return**–POT–PST.UNW
`Argmax`	now	**king**	DEM2.IPL.OBL–GEN2	dispute–SUPER.LAT	**return**–POT–PST.UNW

FIGURE 6.5 – Alignements obtenus avec SimAlign pour une phrase d'exemple. Les flèches noires indiquent les liens communs aux deux méthodes et la flèche rouge indique le lien supplémentaire identifié par `Match`. Les différences entre la référence et les étiquettes de gloses obtenues par alignement sont en **gras**.

Nous pouvons déjà remarquer que la plupart des liens obtenus sont triviaux : il s'agit d'une correspondance de lemmes comme now/*now*. Au-delà de ceux-ci, nous obtenons aussi des synonymes telle la paire khan/*king*. Dans ces situations, comme la glose n'est pas directement présente dans la traduction, si notre système parvient à prédire un synonyme via le lien d'alignement, il est fort probablement utile dans un cas d'utilisation réel.

La différence entre les deux stratégies, dans l'exemple (indiquée par la flèche rouge), réside dans le lien erroné entre turn et *could*, dû à la nature de la méthode `Match`, qui aligne nécessairement toutes les gloses, même lorsqu'il n'y a aucun candidat correct dans la traduction. Dans ce cas précis, l'alignement d'`Argmax` est plus fiable, là où `Match` introduit du bruit en étiquetant « uti » par could.return, par la combinaison des deux mots alignés. Il s'agit là d'un exemple d'une tendance plus générale, où la contrainte de maximalité d'alignement de `Match` mène à des erreurs ; les morphèmes seront alors associés avec des mots vides présents dans beaucoup de phrases comme ici, *could*. En effet, comme ces mots sont très fréquents de manière générale, ils ont plus de probabilité d'être en co-occurrence avec tous les morphèmes et d'avoir un score de similarité suffisamment élevé pour motiver un alignement.

Il faut toutefois noter que d'autres modèles d'alignement peuvent être utilisés comme des modèles d'alignements statistiques, comme les modèles IBM (Brown et al., 1993), ou d'autres modèles d'alignements neuronaux comme le plus récent AWESOME Align (Dou et Neubig, 2021). La seule contrainte de cette section reste l'accès limité du modèle aux données, ce qui encourage l'utilisation d'une approche ne nécessitant pas d'entraînement supplémentaire.

6.3.2 Paramétrage des alignements

Choix des plongements lexicaux Les plongements lexicaux employés par SimAlign peuvent provenir de différentes sources : Jalili Sabet et al. (2020) étudient principalement les représentations statiques calculées par fastText (Bojanowski et al., 2017) ainsi que deux modèles pré-entraînés produisant des représentations contextuelles, mBERT (Devlin et al., 2019) et XLM-RoBERTa base (Conneau et al., 2020). Pour ces deux derniers, les plongements contextualisés permettent d'obtenir des alignements qui peuvent se passer de modèles de distorsion.

D'après les conclusions de (Jalili Sabet et al., 2020), comme les gloses et les mots de la traduction sont dans la même langue, nous faisons le choix d'utiliser BERT (« `bert-base-uncased` »), lorsque la documentation se fait en anglais, et mBERT (« `bert-base-multilingual-uncased` »), multilingue, pour toute autre langue (l'espagnol pour l'uspanteko, en l'occurrence).

De plus, pour ces modèles, deux niveaux d'alignements sont possibles : au niveau des mots ou des sous-mots. S'il est plus avantageux de considérer le second pour les expériences de (Jalili Sabet et al., 2020), nous avons remarqué, dans les expériences préliminaires, que le premier, qui agrège les représentations des sous-mots pour avoir celle des mots, convenait davantage à notre cas. En effet, comme nous avons observé avec l'exemple de la figure 6.5, une large proportion des unités alignées correspondent exactement et cette configuration garantit qu'une glose lexicale décomposée n'est associée qu'à un seul mot au plus.

Couche du modèle BERT Le choix se pose ensuite sur la couche utilisée par le modèle pré-entraîné pour représenter les mots par SimAlign. Par défaut, la couche 8 est utilisée d'après les meilleurs résultats d'alignements dans (Jalili Sabet et al., 2020). De fait, il s'agit d'une situation intermédiaire, plus contextualisée que les premières couches, mais suffisamment générale par rapport aux couches ultérieures, trop spécifiques. Cette partie vérifiera, par conséquent, si ce constat s'applique bien dans le cadre de notre tâche.

En effet, SimAlign a été conçu principalement pour les alignements de phrases naturelles, dans des langues différentes ; ici, nous souhaitons un alignement entre une suite de gloses lexicales et une phrase, dans la même langue. Nous pouvons alors constater que la séquence de gloses lexicales constitue une « phrase » particulière par son absence de mots outils et par l'ordre parfois très différent par rapport à la traduction. Afin d'estimer la qualité des alignements obtenus en fonction des différentes couches, nous avons donc créé des gloses synthétiques en tsez, car nous ne disposons pas d'alignements de référence entre gloses et traduction.

Nous générons ces gloses synthétiques à partir des phrases traduites en anglais du corpus tsez en suivant la méthodologie suivante. Dans un premier temps, les mots de la phrase traduite sont lemmatisés avec spaCy [4]. Puis, les lemmes correspondant aux mots outils sont supprimés ; nous considérons comme tels, les déterminants, les particules et les pronoms, en se basant sur les étiquettes PoS identifiées par spaCy (respectivement, « DET », « PART » ou « PRON »). Enfin, les lemmes restants sont permutés aléatoirement. Cette approche nous permet de reproduire quelques particularités des gloses lexicales, en les assimilant à des lemmes de mots pleins de la traduction dans un ordre différent. La figure 6.6 présente les gloses synthétiques obtenues pour une phrase en anglais. Nous les noterons `syn1`.

gloses synthétiques	*day see one rainbow forest in*
traduction	*one day she saw a rainbow in the forest*

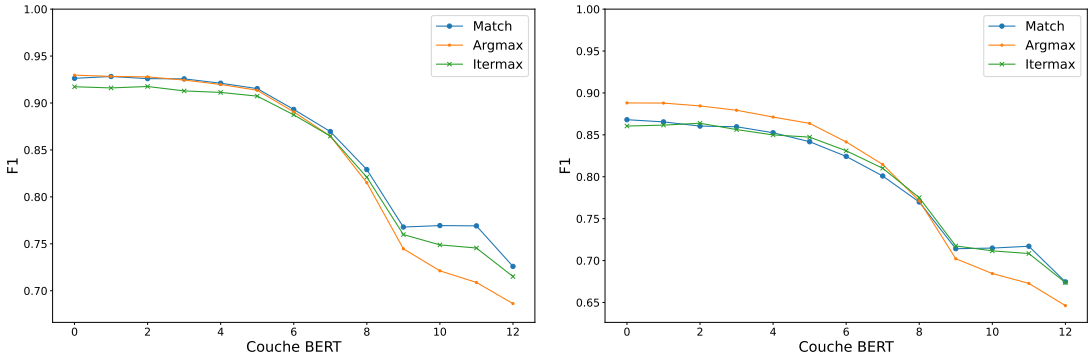
FIGURE 6.6 – Exemple de gloses lexicales synthétiques (`syn1`) obtenues à partir de la traduction en anglais d'une phrase tsez.

Cette procédure peut toutefois être améliorée en utilisant des synonymes de lemmes ; nous utilisons à cette fin, WordNet à travers NLTK (Bird et al., 2009). Si le synonyme est constitué de plusieurs mots, comme les verbes à particule en anglais, nous le considérons comme les gloses lexicales composées (telle « get.up »). De plus, la distorsion générée par les permutations aléatoires peut être contrôlée, afin d'avoir une certaine cohérence dans la succession des gloses lexicales. Cette deuxième version, sensiblement plus réaliste, sera appelée `syn2`.

Grâce à ces gloses synthétiques, nous pouvons désormais avoir une première estimation de la qualité des alignements de SimAlign. En effet, la référence peut être déterminée en retraçant

4. https://spacy.io/.

leur génération. Pour l'évaluation, les liens sont ordinairement évalués selon le taux d'erreur d'alignement (en anglais, *Alignment Error Rate*, AER) ; or, ici, nous choisissons de ne pas distinguer les alignements sûrs et possibles. C'est pourquoi la figure 6.7 présente l'évolution du score F1 en fonction de la couche BERT utilisée pour les trois méthodes de SimAlign.



(a) Pour les gloses `syn1`

(b) Pour les gloses `syn2`

FIGURE 6.7 – Évolution du score F1 en fonction de la couche du modèle BERT utilisée dans SimAlign pour l'alignement des gloses lexicales synthétiques avec les mots de la traduction en anglais des données tsez.

Nous observons que pour la tâche d'alignement de gloses, la huitième couche n'est pas optimale ; les performances sont déjà dégradées à ce stade. De plus, les courbes des trois méthodes suivent une allure similaire : descendante vers les couches les plus élevées. Par conséquent, le meilleur score F1 est obtenu par la méthode Argmax à la couche 0, c'est-à-dire celle qui ne tient compte d'aucun contexte. Il apparaît donc que la contextualisation des plongements est néfaste dans notre cas.

Entre les différentes méthodes d'alignement, nous remarquons peu de différences : avec `syn1`, elles paraissent toutes équivalentes, avec un score légèrement moins bon pour Itermax ; Argmax semble plus performant que les deux autres sur `syn2`.

Bien qu'il s'agisse d'approximations de gloses lexicales, nous avons pu constater quelques tendances dans les impacts des choix de configuration de SimAlign. Pour la suite de nos expériences, nous utilisons donc la couche 0 de BERT (ou mBERT) dans SimAlign et considérons uniquement les méthodes `Argmax` et `Match`, comme annoncé précédemment ; Itermax est en effet un compromis entre les deux approches et n'est pas concluant, au vu des résultats pour notre tâche.

6.3.3 Étude des alignements obtenus

Nous continuons notre analyse des alignements automatiques, en évaluant leur qualité sur les données à notre disposition, avant de les employer dans un processus complet d'alignement de gloses. En effet, cette dernière repose sur l'hypothèse principale [H] qui suppose que les gloses lexicales puissent être retrouvées à travers la traduction. Cette section s'attachera donc à la vérification de cette hypothèse, à travers le corpus tsez.

Tout d'abord, le tableau 6.6 présente quelques statistiques sur les alignements obtenus avec SimAlign sur les données tsez à l'échelle des phrases. Les comportements théoriques évoqués en section 6.3.1 et les tendances observées pour les gloses synthétiques en section 6.3.2 semblent se confirmer sur notre corpus. En effet, nous obtenons moins d'alignements avec la méthode `Argmax` que `Match`, qui parvient à aligner la quasi totalité des morphèmes avec un mot de la traduction.

	`Argmax`	`Match`
Nombre d'alignements par phrase	8,0 (± 3,9)	10,1 (± 5,2)
$\#_{alignements} - \#_{gloses}$ par phrase	−2,12 (± 2,1)	−0,02 (± 0,2)
Nombre de phrases où certaines gloses *ne* sont *pas* alignées	1 554	31

TABLE 6.6 – Quelques statistiques à l'échelle des phrases sur les alignements automatiques obtenus entre les gloses lexicales tsez et les mots de la traduction en anglais.

Notons que les 31 phrases concernées par le manque d'alignement avec `Match` sont les seules qui comportent davantage de gloses lexicales que de mots dans la traduction.

Couverture des alignements Si la méthode `Match` permet d'assurer, en théorie, un alignement avec un mot de la traduction pour chaque glose lexicale, ce n'est pas le cas pour l'alignement `Argmax`, comme nous venons de le voir. Nous étudions donc la couverture des alignements automatiques vis-à-vis des gloses, plus en détail, au niveau des morphèmes. À titre comparatif, nous introduisons la méthode `ExactMatch` qui associe deux unités lorsque les lemmes, obtenues avec spaCy, correspondent, une approche de base déjà utilisée par Georgi (2016). Le tableau 6.7 présente le nombre et la proportion de gloses lexicales *non alignées* pour ces trois méthodes.

	base		+ dictionnaire							
Taille du corpus	/		200		500		1 000		1 600	
`ExactMatch`	6 530	(35,0 %)	-		-		-		-	
`Argmax`	3 615	(19,4 %)	1 223	(6,6 %)	858	(4,6 %)	733	(3,9 %)	627	(3,4 %)
`Match`	35	(0,2 %)	18	(0,1 %)	16	(0,1 %)	9	(0,0 %)	9	(0,0 %)

TABLE 6.7 – Occurrences (et proportion) de gloses lexicales *non alignées* par `ExactMatch` et SimAlign, où les alignements sont éventuellement complétés par un dictionnaire pour les données tsez.

Nous observons tout d'abord que la méthode `ExactMatch`, simple à mettre en œuvre, parvient déjà à aligner une partie conséquente des gloses, environ les deux tiers. L'utilisation de SimAlign permet de couvrir une plus grande proportion : 20 % restent non alignées avec `Argmax` et `Match` parvient à associer un mot de la traduction pour la quasi totalité des mots. Les différences entre les deux approches sont encore une fois visibles : l'une produit des alignements plus parcimonieux, tandis que l'autre garantit un mot cible pour toute glose.

Comme la proportion de gloses lexicales non alignées par `Argmax` notamment ne peut pas être négligée, nous avons recours à un dictionnaire afin de compléter ces liens manquants, comme indiqué dans l'illustration de notre approche en figure 6.4. Ce dernier est constitué en associant à tout morphème source déjà aligné, le lemme du mot cible qui lui est le plus fréquemment associé. Par exemple, la traduction peut ne pas répéter les mots d'une phrase à l'autre en employant un pronom (comme « *Le corbeau* voit un arbre. *Il* s'envole ») ; à travers un dictionnaire, nous pouvons alors retrouver le mot originel. Dans notre expérience, nous faisons varier la taille du corpus utilisé pour créer le dictionnaire, de 200 à 1 600 phrases. Ici, il faut souligner que le dictionnaire est créé uniquement à partir des alignements automatiques et non des gloses de référence, afin d'atteindre les limites d'une approche complètement non supervisée.

Si l'usage de ce lexique n'impacte que très peu la méthode `Match` par définition, `Argmax` voit une baisse nette des gloses lexicales non alignées, avec moins de 5 % dès 500 phrases de supervision. Nous pouvons ainsi réduire l'écart entre les deux méthodes par le biais du dictionnaire.

Correspondance entre les gloses Comme nous obtenons des alignements automatiques parvenant à couvrir la plupart des gloses lexicales, il s'agit maintenant d'évaluer leur qualité. Malheureusement, encore une fois, aucun des corpus étudiés ne comporte les liens de référence entre les gloses lexicales et les mots de la traduction. Nous nous intéressons donc à une « borne inférieure » de la pertinence de ces alignements, en nous focalisant uniquement sur les correspondances *exactes*. Le tableau 6.8 compare alors les trois méthodes, en considérant aussi les versions complétées par le dictionnaire.

	base	+ dictionnaire			
Taille du corpus	/	200	500	1 000	1 600
`ExactMatch`	59,1	-	-	-	-
`Argmax`	58,3	66,2	67,8	67,6	68,2
`Match`	60,1	60,1	60,1	60,1	60,2

TABLE 6.8 – Correspondances *exactes* (%) entre les gloses *lexicales* de référence et celles obtenues par alignement pour les données tsez.

Dans l'ensemble, nous remarquons qu'environ 60 % des mots de la traduction alignés correspondent à la glose de référence, dans le cas de base ; autrement dit, ce sont des liens triviaux pour les méthodes. Si `Argmax` obtient moins de correspondance que `Match`, voire même que `ExactMatch`, l'utilisation du dictionnaire permet d'améliorer nettement la correspondance entre les gloses obtenues par alignement et la référence. Nous rappelons ici que le dictionnaire est constitué en se basant sur les alignements automatiques calculés sur le corpus d'entraînement et non sur la référence. À l'inverse, au vu du nombre de liens d'alignement supplémentaires identifiés par `Match`, cette méthode semble moins efficace avec un score plus élevé de seulement deux points dans le cas de base ; une partie non négligeable des alignements identifiés par `Match` sont donc erronés, comme dans le cas de l'exemple 6.5.

Notons que la somme des proportions de gloses non alignées (tableau 6.7) et de gloses correspondant à la référence (tableau 6.8) n'est pas égale à 1 pour la méthode `ExactMatch` à cause des gloses composées. En effet, pour la glose de référence be.NPRS, l'alignement donne la glose be, par exemple.

Finalement, il faut souligner que les valeurs présentées ici indiquent uniquement les correspondances *exactes* ; les paires de synonymes, comme khan/*king* dans la figure 6.5, sont considérées comme fausses. Dans un cas réel d'utilisation cependant, la prédiction d'un tel lemme d'un mot de la traduction peut être amplement suffisant comme glose candidat. Nous avons donc ici un aperçu sous-estimé de la qualité des alignements automatiques.

6.3.4 Annotation manuelle des alignements

Enfin, nous avons également effectué une annotation manuelle des alignements automatiques, afin d'évaluer leur correspondance avec le jugement humain. Nous avons réparti les 200 phrases de test en tsez en paquets de 50, décalés de 25 phrases pour pouvoir mesurer l'accord inter-annotateur

sur les parties en commun. Trois annotateurs (A0, A1 et A2) ont corrigé manuellement les alignements de SimAlign obtenus avec la méthode `Match`.

Chaque phrase du paquet est présentée et annotée, comme illustré en figure 6.8 :

— Les informations générales sur la phrase avec son identifiant (code ISO + numéro de la phrase dans le corpus d'origine, ici 1910) et la traduction de la phrase pour en comprendre le sens ;
— Les gloses lexicales uniquement, précédées de leur indice (les gloses grammaticales sont ignorées pour l'alignement) ;
— L'alignement original donné par la méthode `Match` de SimAlign ;
— La correction manuelle effectuée par l'annotateur, où chaque glose lexicale doit avoir un et un seul mot cible aligné ; l'utilisation de l'astérisque indique que la glose peut difficilement être alignée (comme ici pour « go ») ;
— La traduction de la phrase où chacun des mots est précédé de son indice ;
— Un champ libre pour les remarques, en particulier concernant les alignements avec astérisques.

```
Sentence ddo1910: and the fox carried it to the crow
Gloss: 00_carry 01_fox 02_crow 03_go
Original: 00-03 01-02 02-07 03-05
Correction: 00-03 01-02 02-07 03-*
Translation: 00_and 01_the 02_fox 03_carried 04_it 05_to 06_the 07_crow
@comments: go (3) unaligned or aligned with to (5)?
```

FIGURE 6.8 – Exemple d'annotation manuelle d'un alignement pour une phrase tsez.

Paquet	P0 (000-050)	P1 (025-075)	P2 (050-100)	P3 (075-125)	P4 (100-150)
Annotateur	A0	A1	A2	A0	A1
Exactitude	67,1	68,6	67,3	71,3	67,9

TABLE 6.9 – Exactitudes des alignements automatiques par rapport au jugement humain sur des paquets de 50 phrases en tsez.

Paquets (i & j)	Exactitude i	Exactitude j	Accord inter-annotateur	κ de Cohen
0 & 1	69,0	67,2	86,4	0,86
1 & 2	70,0	65,0	85,6	0,83
2 & 3	69,6	78,2	86,8	0,78
3 & 4	64,6	66,2	90,9	0,88

TABLE 6.10 – Accord inter-annotateur sur les parties en commun des paquets, avant discussion entre annotateurs.

Le tableau 6.9 présente les scores d'exactitude obtenus pour chacun de ces paquets, en comparant les liens d'alignement de SimAlign avec ceux des annotateurs. Les alignements automatiques atteignent donc en moyenne une correspondance de 68,4, ce qui est déjà meilleur que les valeurs

de base obtenues dans la section précédente au tableau 6.8. Bien que plusieurs récits tirés du folklore se succèdent sur l'ensemble de ces phrases, la qualité des alignements semble relativement stable.

En outre, dans le tableau 6.10, nous nous intéressons de plus près aux phrases en commun entre annotateurs ; nous reportons les exactitudes selon chacun séparément puis calculons les correspondances entre les deux annotations manuelles. Nous constatons, dans l'ensemble, un accord inter-annotateur élevé, comme en témoignent les valeurs des κ de Cohen, et ce, alors que les trois annotateurs ne se sont pas accordés sur les cas ambigus. En effet, à ce stade, seules les grandes lignes étaient fixées comme l'utilisation de l'astérisque en cas de doute. Ceci suggère donc qu'une grande partie des annotations sont faciles à établir ; nous pouvons penser à la proportion de correspondance exacte mais aussi aux synonymes.

Nous remarquons ainsi que les alignements automatiques de SimAlign paraissent de qualité satisfaisante, malgré quelques erreurs persistantes pour la méthode `Match`. Nous aboutissons à la même conclusion que (Georgi, 2016) et l'hypothèse [H] semble vérifiée : les gloses lexicales peuvent bien être retrouvées à partir de la traduction.

6.4 Modèle statistique pour la prédiction de gloses

L'autre volet de notre approche repose sur la ré-utilisation d'un modèle à base de CRF permettant de spécifier l'ensemble des étiquettes possibles localement, à savoir à l'échelle de chaque phrase traitée. De cette manière, nous pouvons aborder le problème de la variété des gloses lexicales, sans rendre le calcul excessivement coûteux, en proposant dynamiquement des étiquettes pertinentes depuis les mots de la traduction.

6.4.1 Lost, un modèle de traduction statistique à l'origine

Nous avons choisi d'utiliser Lost [5] (Lavergne et al., 2011, 2013), initialement un modèle de traduction statistique, pour notre tâche, car il effectue un étiquetage de séquences en permettant d'adapter les étiquettes possibles à chaque phrase. Nous pouvons alors obtenir des espaces de recherche différents pour les paires de phrases source-cible $(\boldsymbol{x}, \boldsymbol{z})$, $\mathcal{Y}(\boldsymbol{x}, \boldsymbol{z})^T$, comme défini en section 6.2.4. En effet, la génération de gloses peut être considérée comme une sorte de traduction de la langue source vers une « langue » des gloses, ce qui a par ailleurs motivé l'utilisation de modèles de traduction neuronales pour la tâche (voir section 2.3.3).

Le modèle s'appuie sur la même théorie que les CRF, présentée en section A.2 ; ici, nous nous focaliserons sur ses points spécifiques, en mettant en exergue ce qui concerne la tâche de génération de gloses, plus simple que la traduction automatique statistique. En effet, certains problèmes propres à cette dernière, comme la segmentation de la phrase source, ne se posent pas dans notre cas ; les unités sources sont déjà segmentées et à un morphème est nécessairement affectée une et une seule étiquette.

Malgré les différences, la tâche d'étiquetage de gloses peut correspondre à une traduction de la phrase source vers une phrase glosée dans une autre langue, avec une grande variété dans les étiquettes possibles ; c'est ce qui a motivé l'approche neuronale de Zhao et al. (2020). De plus, les problématiques liées aux différences dans l'ordre des unités sont également communes, du fait des divergences linguistiques (Dorr, 1994).

5. Disponible sur `https://github.com/shuokabe/gloss_lost/tree/main/lost`.

D'un point de vue théorique, comme il s'agit d'un modèle de traduction basé sur les n-grammes, Lost détermine pour une phrase source $\boldsymbol{x}$, une phrase cible $\boldsymbol{y}$ maximisant la probabilité conditionnelle $p_\theta(\boldsymbol{y}|\boldsymbol{x})$. Pour ce faire, Lost tient aussi compte de l'ensemble des dérivations possibles $\boldsymbol{d}$ permettant d'obtenir $\boldsymbol{y}$ à partir de $\boldsymbol{x}$, comme le ré-ordonnancement des unités sources. Nous obtenons alors pour le moment, l'équation (6.5), en notant $\boldsymbol{G}$ le vecteur des caractéristiques et $\boldsymbol{\theta}$ ses paramètres associés, d'après l'équation caractéristique des CRF (6.1) :

$$p_{\boldsymbol{\theta}}(\boldsymbol{y}|\boldsymbol{x}) = \sum_{\boldsymbol{d}\in\mathcal{D}(\boldsymbol{x})} p_{\boldsymbol{\theta}}(\boldsymbol{y},\boldsymbol{d}|\boldsymbol{x}) = \sum_{\boldsymbol{d}\in\mathcal{D}(\boldsymbol{x})} \frac{\exp\left\{\boldsymbol{\theta}^T\boldsymbol{G}(\boldsymbol{y},\boldsymbol{d},\boldsymbol{x})\right\}}{\sum\limits_{\substack{\boldsymbol{d'}\in\mathcal{D}(\boldsymbol{x})\\ \boldsymbol{y'}\in\mathcal{Y}(\boldsymbol{d'},\boldsymbol{x})}} \exp\left\{\boldsymbol{\theta}^T\boldsymbol{G}(\boldsymbol{y'},\boldsymbol{d'},\boldsymbol{x})\right\}} \tag{6.5}$$

Ici, $\mathcal{D}(\boldsymbol{x})$ correspond à l'ensemble des dérivations possibles pour la phrase source $\boldsymbol{x}$ et $\mathcal{Y}(\boldsymbol{d'},\boldsymbol{x})$ à l'ensemble des traductions possibles pour $\boldsymbol{x}$ avec la dérivation $\boldsymbol{d'}$. On remarquera, par ailleurs, les similitudes avec l'équation de notre modèle (6.4), où notre source est la paire de phrases ($\boldsymbol{x}$, $\boldsymbol{z}$) et la variable latente, a.

On obtient alors la règle suivante pour l'inférence :

$$\boldsymbol{y}^* = \arg\max_{\boldsymbol{y},\boldsymbol{d}} \; p_\theta(\boldsymbol{y},\boldsymbol{d}|\boldsymbol{x}) \tag{6.6}$$

Nous avons en réalité présenté les modèles n-grammes jusqu'à présent ; la différence principale de Lost par rapport à ses équivalents réside dans l'aspect intégré de l'apprentissage de ses paramètres. En effet, le modèle effectue une maximisation de la log-vraisemblance conditionnelle ; en se basant sur l'équation (6.5), nous pouvons donc écrire, avec ($\boldsymbol{x}^n$, $\boldsymbol{y}^n$) représentant une paire de phrases source-cible :

$$\mathcal{L}(\boldsymbol{\theta}) = \sum_n \left[\log \sum_{\boldsymbol{d}\in\mathcal{D}(\boldsymbol{x}^n)} \exp\left\{\boldsymbol{\theta}^T\boldsymbol{G}(\boldsymbol{y}^n,\boldsymbol{d},\boldsymbol{x}^n)\right\} - \log \sum_{\substack{\boldsymbol{d'}\in\mathcal{D}(\boldsymbol{x}^n)\\ \boldsymbol{y'}\in\mathcal{Y}(\boldsymbol{d'},\boldsymbol{x}^n)}} \exp\left\{\boldsymbol{\theta}^T\boldsymbol{G}(\boldsymbol{y'},\boldsymbol{d'},\boldsymbol{x}^n)\right\} \right] \tag{6.7}$$

Contrairement aux CRF classiques, l'équation (6.7), où nous observons une différence de deux *log-sum-exp*, n'est plus convexe à cause de la variable latente $\boldsymbol{d}$ (Sutton et McCallum, 2007). Malgré cela, nous pouvons effectuer une descente de gradient, qui se calcule comme en équation (6.8), même si cela aboutit à un optimum local.

$$\frac{\partial\mathcal{L}(\boldsymbol{\theta})}{\partial\theta_k} = \sum_n \left[\frac{\sum\limits_{\boldsymbol{d}\in\mathcal{D}(\boldsymbol{x}^n)} G_k(\boldsymbol{y}^n,\boldsymbol{d},\boldsymbol{x}^n)\exp\left\{\boldsymbol{\theta}^T\boldsymbol{G}(\boldsymbol{y}^n,\boldsymbol{d},\boldsymbol{x}^n)\right\}}{\sum\limits_{\boldsymbol{d}\in\mathcal{D}(\boldsymbol{x}^n)} \exp\left\{\boldsymbol{\theta}^T\boldsymbol{G}(\boldsymbol{y}^n,\boldsymbol{d},\boldsymbol{x}^n)\right\}} - \sum_{\substack{\boldsymbol{d'}\in\mathcal{D}(\boldsymbol{x}^n)\\ \boldsymbol{y'}\in\mathcal{Y}(\boldsymbol{d'},\boldsymbol{x}^n)}} G_k(\boldsymbol{y},\boldsymbol{d},\boldsymbol{x}^n)p_\theta(\boldsymbol{y'},\boldsymbol{d'}|\boldsymbol{x}^n) \right] \tag{6.8}$$

En l'état, le calcul de ce gradient est coûteux, mais Lost met en œuvre plusieurs simplifications. Tout d'abord, l'approche de Lavergne et al. (2013) a été de recourir à une approximation qui considère seulement les n meilleures hypothèses à l'inférence, en se basant sur un espace de recherche, représentant par un graphe l'ensemble des étiquetages possibles. De plus, il utilise des caractéristiques G_k unigrammes et bigrammes uniquement. Cette restriction à une portée locale permet entre autres d'employer un algorithme *forward-backward* modifié et reste raisonnable, dans la mesure où les CRF reposent également sur une telle dépendance. En outre, aux étiquettes impossibles, à savoir, dans notre cas, aux gloses lexicales qui ne font pas partie de $\mathcal{Y}(\boldsymbol{x},\boldsymbol{z})$, est

attribué un poids infiniment négatif, empêchant leur prédiction par le modèle car complètement absentes de l'espace de recherche. Enfin, l'optimisation de cette descente de gradient se fait avec Rprop (Riedmiller et Braun, 1993), qui permet entre autres de gérer les configurations avec une grande quantité d'étiquettes.

Atteignabilité des références Un problème récurrent observé par Liang et al. (2006) pour les modèles de traduction statistiques est la non atteignabilité des références. Il s'agit de situations où la phrase de référence ne peut pas être générée par le modèle, notamment dû aux étiquettes manquantes car trop peu ou jamais observées. Pour y remédier, différentes stratégies existent ; nous n'utilisons pas la meilleure identifiée par (Lavergne et al., 2013) qui consiste en l'utilisation de pseudo-références oracles définies sur l'espace de recherche en entier, mais l'alternative plus simple et déjà efficace qui complète localement l'espace de recherche avec les étiquettes de références. En pratique, dans notre cas, cela revient simplement à ajouter les gloses de référence dans l'espace de recherche, car l'unité de base est uniquement le morphème ; nous n'avons pas de phrase source à segmenter, ce qui n'impacte pas outre mesure la complexité du graphe.

Paramétrage de Lost Dans nos expériences, nous gardons le paramétrage par défaut, suffisante pour nos expériences. Nous arrêtons l'apprentissage à 15 itérations ; nous n'avons pas observé d'améliorations notables au-delà.

De plus, si Lost propose l'utilisation d'une régularisation *elastic-net* (Zou et Hastie, 2005), combinaison linéaire des pénalités l_1 et l_2, nous choisissons la régularisation l_1, d'après nos expériences préliminaires ; elle incite de fait à une sélection des caractéristiques les plus pertinentes (Tibshirani, 1996). Le poids associé est de $0{,}5$; aucune tendance claire n'a été observée en modifiant la valeur de ce poids, entre $0{,}3$ et $0{,}7$.

6.4.2 Définir les étiquettes

Notre modèle a naturellement pour objectif de générer les étiquettes, pour une paire de phrase source-cible $(\boldsymbol{x}, \boldsymbol{z})$, depuis l'ensemble $\mathcal{Y}(\boldsymbol{x}, \boldsymbol{z})$, comme indiqué en section 6.2.4. Cependant, au-delà de l'étiquette principale g, qui correspond soit directement à la glose (depuis $\mathcal{Y}_G$ ou $\mathcal{Y}_L(\boldsymbol{x})$) soit à l'indice d'un mot de la traduction, nous prédisons trois informations supplémentaires : le type binaire b de la glose, grammaticale (GRAM) ou lexicale (LEX), l'étiquette PoS p associée ainsi que sa position dans la phrase cible si aligné. L'idée est de compenser la grande variété des étiquettes, dont certaines ne seraient pas suffisamment observées, voire inconnues, et d'apprendre des caractéristiques plus générales, donc plus robustes. Selon nos expériences préliminaires, l'utilisation de ces étiquettes structurées obtient un impact notablement positif sur les prédictions par rapport à la version de base avec seulement g. Le tableau 6.11 présente un exemple pour chaque provenance d'étiquettes.

En pratique, si le type de glose b s'obtient de manière déterministe à partir de l'étiquette principale, l'étiquette PoS p peut être plus complexe. Nous attribuons pour toutes les gloses grammaticales, l'étiquette « GRAM » et pour les mots de la traduction, nous avons recours à spaCy. Quant aux étiquettes provenant de $\mathcal{Y}_L(\boldsymbol{x})$, le dictionnaire contient en réalité la partie du discours associée à l'étiquette la plus fréquente ; à travers le lien king/*khan*, nous pouvons projeter la partie du discours NOUN de « *khan* » sur la glose de référence « king ». Nous remarquerons ici que le dictionnaire, contrairement à la section précédente, est constitué des gloses de référence ; ce changement est dû à la volonté de prédire des gloses les plus proches de l'annotation originelle.

provenance	ensemble	étiquette principale g	type de glose b	étiquette PoS p	indice l
glose grammaticale	$\mathcal{Y}_G$	GEN1	GRAM	GRAM	-
dictionnaire	$\mathcal{Y}_L(\boldsymbol{x})$	king	LEX	NOUN	-
traduction	z_2	khan	LEX	NOUN	2
référence	y_1	khan	LEX	NOUN	2

TABLE 6.11 – Exemple d'étiquettes à prédire pour chaque provenance d'étiquettes, en utilisant la phrase exemple de 6.5. La référence est donnée uniquement lors de l'entraînement, si elle est absente de l'ensemble des étiquettes possibles

L'indice l présenté ici est théorique : nous utilisons dans Lost une approximation en la discrétisant. Malgré la perte d'information, cette approche permet de limiter l'effet de la longueur de la phrase et de comparer les positions des mots plus généralement. Nous divisons la phrase en quatre parties (donc quatre valeurs, 1/4, 2/4, 3/4 et 4/4) et affectons des valeurs distinctes -1 pour les étiquettes du dictionnaire et -2 pour les gloses grammaticales.

Nous avons ainsi défini les différents types d'étiquettes, qui sont utilisées pour constituer l'espace de recherche, rendu alors spécifique à chaque paire de phrases étudiée.

6.4.3 Caractéristiques unigrammes et bigrammes dans Lost

Lost étant basé sur la théorie des CRF, il repose sur l'utilisation de caractéristiques. Nous présentons ici les principales informations utilisées pour les constituer. L'entrée du modèle est uniquement le morphème source m, à partir duquel nous obtenons :

- sa position dans le mot, t, représentée par une valeur numérique pour les mots avec plusieurs morphèmes ou « F » pour les mots-morphèmes ;
- sa longueur en nombre de caractères, l ;
- ses trois premières et dernières lettres, d et e ;
- une variable binaire indiquant s'il est directement présent dans la traduction, cs ;
- sa position dans la phrase, ps, ici aussi discrétisée en la divisant en quatre parties ;
- sa représentation en consonnes (C) et voyelles (V), dl ;
- la longueur du mot en nombre de morphèmes, ml.

Les trois dernières caractéristiques correspondent à des finalités spécifiques. La variable de copie cs est active lorsqu'un morphème source se retrouve tel quel dans la phrase traduite, notamment afin de traiter les noms propres. Comme il s'agit d'une correspondance lettre à lettre, les alphabets des langues source et cible doivent être identiques ; sinon, la translittération peut être une solution. La position (discrète) ps permet de comparer les distances relatives des unités entre les deux phrases. Le but est d'empêcher les associations de morphèmes sources et de mots de la traduction ayant une trop grande distorsion, inspiré par les modèles d'alignements statistiques. Enfin, la représentation délexicalisée dl s'inscrit dans la continuité de l'idée de généraliser les morphèmes ; en allant au-delà de la forme orthographique, nous avons accès aux patrons de morphèmes. Ceci ouvre également la possibilité vers une approche multilingue, discutée en section 6.6.

Pour les sorties, nous avons présenté les étiquettes structurées dans la partie précédente (section 6.4.2). Nous avons recours ici aussi à une variable de copie ct qui signale si l'étiquette est

littéralement dans la phrase source, avec une valeur -1 pour les gloses grammaticales, ainsi qu'à pt qui correspond à la version discrète de la position dans la phrase cible, détaillée plus tôt.

La figure 6.9 explicite la représentation des données dans Lost pour une phrase d'exemple. Nous présentons ici les sorties de référence ; c'est pourquoi, l'étiquette principale g correspond aux vraies gloses et les parties du discours ainsi que les positions dans la phrase sont accessibles grâce à l'alignement automatique.

	entrée	quelques caractéristiques								sorties et caractéristiques				
i	m	t	l	d	e	cs	ps	dl	ml	g	b	p	ct	pt
0	nesi	0	4	nes	esi	0	1/4	CVCV	2	he.OBL	LEX	PRON	0	1/4
1	s	1	1	s	s	0	1/4	C	2	GEN1	GRAM	GRAM	-1	-2
2	ɬˤono	F	5	ɬˤo	ono	0	2/4	CCVCV	1	three	LEX	NUM	0	3/4
3	uži	F	3	uži	uži	0	2/4	VCV	1	son	LEX	NOUN	0	4/4
4	zow	0	3	zow	zow	0	3/4	CVC	2	be.NPRS	LEX	VERB	0	2/4
5	n	1	1	n	n	0	4/4	C	2	PST.UNW	GRAM	GRAM	-1	-2

FIGURE 6.9 – Exemple d'entrée, d'étiquettes de sortie et de caractéristiques associées pour une phrase tsez. L'intitulé des colonnes est détaillé ci-dessus.

Le tableau 6.12 récapitule les patrons de caractéristiques unigrammes (i) et bigrammes ($i-1$ et i) utilisés dans Lost, illustrés par un exemple pour une position de la figure 6.9. Nous avons regroupé les patrons : la première partie présente les caractéristiques de base pour l'étiquette (principale) de gloses. La deuxième correspond aux fonctions impliquant les deux autres informations à prédire, b et p, avec, en particulier, les tests bigrammes concernant l'une des deux avec la glose (comme les patrons 16 ou 19). La troisième correspond aux caractéristiques reposant sur la copie d'unités et la position relative. Enfin, la dernière se concentre sur l'aspect délexicalisé, comme premier pas vers une généralisation plus poussée du modèle.

6.4.4 Configurations explorées

Nous comparons principalement trois configurations de Lost, afin d'évaluer les apports de chaque ensemble d'étiquettes correspondant aux gloses lexicales.

— La première, `S1`, ne considère que les gloses lexicales du dictionnaire ($\mathcal{Y}_L(\boldsymbol{x})$), en référence aux systèmes déjà existants qui sauvegardent les précédentes occurrences observées pour les étiquetages futurs ;

— La deuxième, `S2`, à l'inverse, utilise uniquement les lemmes des mots de la traduction $\boldsymbol{z}$ (correspondant aux indices $\{1, \ldots, |\boldsymbol{z}|\}$) et, en ce sens, se rapproche du modèle cible (TRS) dans l'approche de McMillan-Major (2020), qui repose sur la traduction pour prédire les gloses possibles ;

— La troisième, `S3`, combine les deux jeux d'étiquettes possibles afin de bénéficier des deux sources d'informations, ce qui comble les faiblesses des deux modèles précédents ; nous avons de ce fait des étiquettes corroborées par les données de supervision, à travers le dictionnaire, et la possibilité de traiter les morphèmes inconnus grâce à la traduction.

Dans le dernier cas, il est possible que la même glose lexicale soit présente dans les deux sources d'étiquettes. Si la partie du discours est différente (comme (return, VERB) et (return, NOUN)), il s'agit de deux étiquettes distinctes ; sinon, nous ne conservons que celle provenant de la traduction. En effet, elle contient une information supplémentaire sur la position au sein de la phrase.

ID	Caractéristiques	Test	Exemple (figure 6.9, $i = 4$)
1	uni-gloss	$\mathbb{1}(g_i = g)$	PST.UNW
2	bi-gloss	$\mathbb{1}(g_i = g) \wedge \mathbb{1}(g_{i-1} = g')$	(be.NPRS, PST.UNW)
3	uni-gloss-morph	$\mathbb{1}(g_i = g) \wedge \mathbb{1}(m_i = m)$	(PST.UNW, n)
4	uni-gloss-bi-morph	$\mathbb{1}(g_i = g) \wedge \mathbb{1}(m_{i-1} = m') \wedge \mathbb{1}(m_i = m)$	(PST.UNW, zow, n)
5	uni-gloss-position	$\mathbb{1}(g_i = g) \wedge \mathbb{1}(t_i = t)$	(PST.UNW, 1)
6	uni-gloss-length	$\mathbb{1}(g_i = g) \wedge \mathbb{1}(l_i = l)$	(PST.UNW, 1)
7	uni-gloss-start	$\mathbb{1}(g_i = g) \wedge \mathbb{1}(d_i = d)$	(PST.UNW, n)
8	uni-gloss-end	$\mathbb{1}(g_i = g) \wedge \mathbb{1}(e_i = e)$	(PST.UNW, n)
9	bi-gloss-morph	$\mathbb{1}(g_i = g) \wedge \mathbb{1}(g_{i-1} = g') \wedge \mathbb{1}(m_i = m)$	(be.NPRS, PST.UNW, n)
10	*uni/bi-bin	$\mathbb{1}(b_i = b)$ $(\wedge\mathbb{1}(b_{i-1} = b'))$	GRAM ((LEX, GRAM))
11	uni/bi-pos	$\mathbb{1}(p_i = p)$ $(\wedge\mathbb{1}(p_{i-1} = p'))$	GRAM ((VERB, GRAM))
12	uni-bin-morph	$\mathbb{1}(b_i = b) \wedge \mathbb{1}(m_i = m)$	(GRAM, n)
13	*uni-bin-position/length	$\mathbb{1}(b_i = b) \wedge \mathbb{1}(t_i = t)/\mathbb{1}(l_i = l)$	(GRAM, 1) / (GRAM, 1)
14	uni-bin-start/end	$\mathbb{1}(b_i = b) \wedge \mathbb{1}(d_i = d)/\mathbb{1}(e_i = e)$	(GRAM, n) / (GRAM, n)
15	*uni-bin-bi-position	$\mathbb{1}(b_i = b) \wedge \mathbb{1}(t_i = t) \wedge \mathbb{1}(t_{i-1} = t')$	(GRAM, 0, 1)
16	bi-bin-gloss	$\mathbb{1}(g_i = g) \wedge \mathbb{1}(b_{i-1} = b')$	(LEX, PST.UNW)
17	bi-gloss-bin	$\mathbb{1}(b_i = b) \wedge \mathbb{1}(g_{i-1} = g')$	(be.NPRS, GRAM)
18	uni-pos-morph	$\mathbb{1}(p_i = p) \wedge \mathbb{1}(m_i = m)$	(GRAM, n)
19	bi-pos-gloss	$\mathbb{1}(g_i = g) \wedge \mathbb{1}(p_{i-1} = p')$	(VERB, PST.UNW)
20	bi-gloss-pos	$\mathbb{1}(p_i = p) \wedge \mathbb{1}(g_{i-1} = g')$	(be.NPRS, GRAM)
21	uni-pos-start/end	$\mathbb{1}(p_i = p) \wedge \mathbb{1}(d_i = d)/\mathbb{1}(e_i = e)$	(GRAM, n) / (GRAM, n)
22	uni-copy-trg	$\mathbb{1}(ct_i = ct)$	-1
23	uni-copy-trg-src	$\mathbb{1}(ct_i = ct) \wedge \mathbb{1}(cs_i = cs)$	(-1, 0)
24	uni-posi-ts	$\mathbb{1}(pt_i = pt) \wedge \mathbb{1}(ps_i = ps)$	(-2, 4/4)
25	uni-gloss-morph-pts	$\mathbb{1}(g_i = g) \wedge \mathbb{1}(pt_i = pt)$ $\wedge\mathbb{1}(m_i = m) \wedge \mathbb{1}(ps_i = ps)$	(PST.UNW, -2, n, 4/4)
26	uni-gloss-delex	$\mathbb{1}(g_i = g) \wedge \mathbb{1}(dl_i = dl)$	(PST.UNW, C)
27	*uni-bin-delex	$\mathbb{1}(b_i = b) \wedge \mathbb{1}(dl_i = dl)$	(GRAM, C)
28	uni-pos-delex	$\mathbb{1}(p_i = p) \wedge \mathbb{1}(dl_i = dl)$	(GRAM, C)
29	*uni-bin-bi-delex	$\mathbb{1}(b_i = b) \wedge \mathbb{1}(dl_{i-1} = dl') \wedge \mathbb{1}(dl_i = dl)$	(GRAM, CVC, C)
30	*bi-bin-bi-delex	$\mathbb{1}(b_{i-1} = b') \wedge \mathbb{1}(b_i = b)$ $\wedge\mathbb{1}(dl_{i-1} = dl') \wedge \mathbb{1}(dl_i = dl)$	(LEX, PST.UNW, CVC, C)
31	*uni-bin-rel-morph-position	$\mathbb{1}(b_i = b) \wedge \mathbb{1}(t_i = t) \wedge \mathbb{1}(ml_i = ml)$	(GRAM, 1, 2)

TABLE 6.12 – Patrons de caractéristiques **uni**grammes et **bi**grammes utilisées dans Lost, avec un exemple pour la phrase de la figure 6.9. L'astérisque dénote les caractéristiques multilingues, principalement utilisées à la section 6.6.

Enfin, dans toutes les configurations, la référence est ajoutée si elle est absente de l'ensemble des étiquettes, afin d'assurer l'atteignabilité lors de l'apprentissage (voir section 6.4.1).

6.5 Résultats expérimentaux et analyses

Dans cette section, nous présentons essentiellement les résultats obtenus par notre modèle lors du défi partagé SIGMORPHON 2023 sur la génération automatique des gloses, dont les modalités ont été présentées en section 2.3.2. Par la nature même de notre approche d'étiquetage de séquence, notre participation concerne uniquement la piste ouverte, où la segmentation en morphèmes de la phrase source est donnée. Cinq des sept langues proposées, décrites en section 3.2.4, ont été étudiées.

6.5.1 Conditions expérimentales

Modèles de base Nous présentons tout d'abord deux modèles simples pour pouvoir situer les performances de nos modèles.

— Le premier, `maj`, similaire à celui de la section 6.2.2, affecte l'étiquette majoritaire (grammaticale ou lexicale cette fois) pour tout morphème observé dans les données d'entraînement. Une étiquette « UNK » est affectée à tous les morphèmes inconnus restants.

— Le deuxième modèle est une combinaison d'un CRF étiquetant soit avec la glose grammaticale, soit l'étiquette unique « lex », à l'instar du modèle CRF présenté en 6.2.2, et donc inspiré de Moeller et Hulden (2018). Dans un second temps, les étiquettes prédites comme lexicales sont identifiées en utilisant le même principe que `maj`. Nous noterons ce modèle en cascade `CRF+maj`.

Cette dernière approche est motivée par la proportion plus élevée de morphèmes grammaticaux ambigus par rapport aux morphèmes lexicaux, pour lesquels une approche systématique à l'aide d'un dictionnaire peut convenir.

Métriques d'évaluation Nous utilisons principalement les scores d'exactitude au niveau des mots et des morphèmes définis comme la proportion de mots (respectivement morphèmes) correctement glosés parmi l'ensemble des mots (respectivement morphèmes). Si, dans une approche d'étiquetage de morphèmes, nous nous sommes concentrés sur la métrique au niveau des morphèmes, le défi partagé privilégiait le score au niveau des mots.

Par ailleurs, ce défi étudie également des métriques complémentaires telles que le score BLEU (Papineni et al., 2002), dû aux méthodes inspirées des modèles de traduction automatique tels Zhao et al. (2020), ainsi que les précisions, rappels et scores F1 différenciés pour les gloses grammaticales et lexicales.

6.5.2 Résultats pour les langues du défi partagé

Le tableau 6.13 présente les résultats officiels pour les exactitudes au niveau des mots (gauche) ou des morphèmes (droite) pour cinq langues : le tsez (`ddo`), le gitksan (`git`), le lezghien (`lez`), le natugu (`ntu`) et l'uspanteko (`usp`). Nos systèmes sont comparés au modèle de base [6] (BASE_SIG) (Ginn, 2023) ainsi qu'aux meilleurs résultats publiés [7] (BEST_SIG), présentés plus en détails en section 2.3.3. Nous reportons aussi les performances de notre système utilisé lors du défi partagé (`S2` dans Okabe et Yvon, 2023b), indiqué par CRF_OFF. Celui-ci emploie toutes les caractéristiques présentées dans le tableau 6.12, hormis la dernière catégorie (26-31). Le nombre de caractéristiques actives pour le modèle `S3` est donné à titre indicatif en annexe A.4.

Nous constatons, en premier lieu, que le modèle de base `maj` est déjà une méthode suffisamment performante, malgré sa simplicité. En effet, pour la plupart des langues, cette stratégie obtient de meilleures performances que le modèle de base du défi partagé BASE_SIG. L'utilisation d'un CRF pour identifier les gloses grammaticales en amont, avec `CRF+maj` permet de gagner quelques points d'exactitude supplémentaires ; notons que dans le cas du tsez (`ddo`), le nombre total d'étiquettes est rédhibitoire pour le CRF standard.

Entre les trois configurations présentées en section 6.4.4, nous remarquons que l'utilisation conjointe des deux sources (`S3`), traduction et dictionnaire, est bénéfique, si ce n'est nécessaire

6. https://github.com/sigmorphon/2023glossingST/tree/main/baseline.
7. https://github.com/sigmorphon/2023glossingST/blob/main/results.md.

modèle	ddo	git	lez	ntu	usp
maj	65,3	28,1	81,2	81,5	72,8
CRF+maj	-	29,4	84,9	88,1	76,2
BASE_ST	75,7	16,4	34,5	41,1	76,6
BEST_ST	**85,8**	31,5	**85,4**	**89,3**	78,5
CRF_OFF	85,5	31,5	83,0	**89,3**	76,7
CRF (S1)	36,2	25,5	65,3	59,0	53,2
CRF (S2)	51,5	29,9	52,8	65,0	63,3
CRF (S3)	85,6	**33,6**	82,8	89,1	**78,9**

modèle	ddo	git	lez	ntu	usp
maj	79,1	51,2	85,8	87,1	79,5
CRF+maj	-	51,1	**88,3**	92,3	82,5
BASE_ST	85,3	25,3	51,8	49,0	82,5
BEST_ST	**92,0**	**52,4**	87,6	**92,8**	**84,5**
CRF_OFF	91,8	51,1	87,0	**92,8**	82,7
CRF (S1)	60,1	47,8	73,5	73,0	65,5
CRF (S2)	70,2	45,3	65,0	76,8	73,3
CRF (S3)	91,9	**52,4**	87,0	**92,8**	84,4

TABLE 6.13 – Exactitudes calculées au niveau des mots (à gauche) et des morphèmes (à droite) pour les modèles de base et nos systèmes sur les corpus de cinq langues du défi partagé SIGMORPHON 2023.

pour avoir un modèle compétitif. Les moindres scores des deux systèmes reposant sur des ensembles d'étiquettes définis partiellement soulignent l'insuffisance des propositions : recourir uniquement aux gloses lexicales vues à l'entraînement (S1), à travers le lexique ($\mathcal{Y}_L(\boldsymbol{x})$), est trop restrictif et conduit aux pires résultats. De même, la traduction seule (S2) ne suffit pas, bien qu'elle permette d'atteindre des valeurs d'exactitude plus proches de maj.

Par ailleurs, si nous comparons les performances entre les langues, nous notons que le gitksan présente des scores considérablement plus bas, quel que soit le modèle. Nous attribuons ceci principalement à la taille des données d'entraînement, 31 phrases seulement, contre 701 pour la deuxième langue la moins dotée de notre étude, le lezghien. Hormis ce cas particulier et complexe du gitksan, notre modèle parvient à obtenir des résultats encourageants, aux alentours de 80 au niveau des mots et davantage pour les morphèmes, et ce, même pour les corpus de moins de mille phrases. En outre, ses scores sont souvent très proches des meilleures valeurs du défi partagé, voire équivalents, en particulier pour les langues avec le moins de données.

Si ce constat positif peut laisser croire que de tels modèles puissent être utilisés assez tôt dans le processus de documentation, il faut souligner que dans un corpus multilingue glosé comme IMTVault (Nordhoff et Krämer, 2022), seulement 16 langues possèdent plus de 700 phrases ; l'obtention de phrases entièrement annotées pour la supervision est de fait très coûteuse et difficile. De plus, nous relevons ici que l'uspanteko, qui bénéficie du plus large nombre d'occurrences de mots en supervision dans nos langues, obtient le deuxième score le plus bas, suggérant que la taille des données n'est pas le seul facteur décisif pour la qualité des prédictions. Néanmoins, les performances sur le gitksan, à savoir des conditions expérimentales difficiles, sont prometteuses et semblent être une première étape vers l'automatisation.

6.5.3 Traitement des morphèmes inconnus

Le défi majeur posé par les gloses lexicales se traduit par la difficulté dans le traitement des morphèmes inconnus lors de l'inférence. Le tableau 6.14 en reporte différentes statistiques sur chaque corpus étudié. Les valeurs reportées diffèrent sensiblement de celles de Ginn et al. (2023) (tableau 3) ; nous ne considérons ici que les morphèmes étiquetés par une glose lexicale et la proportion est calculée sur cet ensemble restreint.

Tout d'abord, concernant les quantités de morphèmes inconnus lors de l'inférence, nous constatons que, hormis pour le gitksan, la plupart des corpus n'en présente qu'une petite proportion, aux alentours de 10 % ou moins. Notons ici que, si la taille du corpus d'entraînement permet effecti-

langue	ddo	git	lez	ntu	usp
nombre de morphèmes inconnus mi	44	200	64	47	181
proportion parmi les gloses lexicales du test (%)	1,02	74,9	9,65	6,16	10,9
dont ceux absents de la traduction at	15	143	42	24	117
proportion (at/mi) (%)	34,1	71,5	65,6	51,1	64,6
exactitude (gloses de référence)	18,2	12,0	4,7	29,8	29,3
exactitude (gloses obtenues par alignement)	15,9	18,0	10,9	40,4	56,9

TABLE 6.14 – Statistiques sur les morphèmes *lexicaux* inconnus dans les données de test. L'exactitude présentée est calculée au niveau des morphèmes par rapport aux gloses de référence ou celles obtenues via alignement comme en section 6.3.

vement d'expliquer en partie ces taux, encore une fois, ce n'est pas le facteur décisif en général. Dans notre cas, les données pour le gitksan étaient clairement divisées en différentes histoires, dont une a été utilisée pour le test ; les autres langues n'observent pas une telle distinction dans le contenu, car elles suivent la séparation standard 80/10/10.

Parmi les morphèmes lexicaux inconnus, nous avons noté qu'une partie non négligeable, allant d'un tiers pour le tsez jusqu'à 70 % en gitksan, n'était pas même présent dans la traduction. Il s'agit alors de cas impossibles à prédire pour les modèles automatiques. En tenant compte de ces chiffres, nous nous intéressons maintenant à l'exactitude obtenue sur ces morphèmes lexicaux inconnus, présentée dans la partie inférieure du tableau 6.14. Lorsque nous comparons les prédictions à la référence, nous obtenons dans l'absolu des valeurs basses, moins de 30, mais il faut les mettre en perspective au vu du nombre de gloses absentes de la phrase cible. Pour le natugu, par exemple, la valeur maximale théorique possible pour l'exactitude est 48,9 ($100 - 51{,}1$), ce qui relativise le score de 29,8.

En complément, nous avons utilisé les alignements automatiques `Match`, afin d'obtenir des gloses « approximées », comme dans la figure 6.5, et par là, contourner le problème des étiquettes impossibles. Hormis pour le tsez, nous observons alors des scores plus élevés, indiquant la prédiction de lemmes issus de la traduction, qui peuvent notamment correspondre à des synonymes des gloses de référence. Néanmoins, rappelons que la méthode `Match` introduit une certaine part de bruit.

6.5.4 Efficacité des modèles statistiques

Un des atouts de notre modèle réside dans sa capacité à mettre à profit efficacement les données d'entraînement, comme il repose sur des CRF, notablement meilleurs même à partir de petites quantités de données. Si nous avons déjà eu un aperçu de cet avantage dans les résultats du lezghien ou du natugu dans le tableau 6.13, nous étudions plus en détail cette caractéristique, en considérant une seule langue, le tsez, mais pour un nombre variable de phrases de supervision.

Le tableau 6.15 présente alors l'évolution de l'exactitude pour différentes tailles de données d'entraînement sur le corpus tsez. Nous remarquons que notre modèle est systématiquement meilleur que la méthode `maj` qui reproduit la configuration actuelle de certains outils d'annotation linguistique, basée sur un dictionnaire. De fait, l'écart de 5 points environ dès 50 phrases se creuse au fur et à mesure que le quantité de phrases de supervision augmente, jusqu'à atteindre plus de

entraînement	50	200	700	1 000	2 000	complet
`maj`	61,0	72,4	76,7	77,6	78,7	79,1
CRF (S3)	66,9	80,6	87,5	89,2	90,7	91,9
Δ CRF (S3) - `maj`	5,9	8,2	10,8	11,6	12,0	12,8

TABLE 6.15 – Exactitude au niveau des morphèmes en tsez pour différentes quantités de phrases données à l'entraînement.

12 points lorsque l'entièreté du corpus est utilisé (voir ligne Δ). Les données supplémentaires sont donc mieux exploitées avec l'approche CRF, réussissant à généraliser à des cas inconnus.

modèle	`maj`		CRF (S3)	
type de glose	gram	lex	gram	lex
50	69,3	51,9	78,1	54,6
200	71,8	73,1	86,7	73,7
1 000	72,3	83,7	91,0	87,2
2 000	72,5	85,9	91,8	89,3
complet	72,5	86,7	92,3	91,3

TABLE 6.16 – Scores F1 différenciés pour les gloses **gram**maticales et **lex**icales pour différentes tailles du corpus tsez.

Le tableau 6.16 présente le score F1 obtenu en séparant les gloses grammaticales des gloses lexicales selon la référence, pour les deux modèles précédents. Nous observons que pour `maj`, les étiquettes grammaticales posent davantage problème que les étiquettes lexicales. Nous attribuons cette tendance aux gloses grammaticales, qui sont volontiers plus ambiguës, avec plusieurs possibilités pour un même morphème par exemple, et aux gloses lexicales avec qu'une seule étiquette possible. La difficulté due à la variété des gloses lexicales est amoindrie en augmentant la taille des données d'entraînement, ce qui permet d'observer plus de morphèmes sources et gloses correspondantes. À l'inverse, le score F1 pour les gloses grammaticales stagne assez tôt, avec moins de moins d'un point de différence entre 200 et 3558 phrases.

De plus, notre modèle `S3` obtient de meilleurs scores par rapport à ce modèle de base, sur les deux niveaux. Nous remarquons que l'avantage principal de l'apprentissage est la désambiguïsation des gloses grammaticales, comme en témoignent les scores plus élevés, de 20 points environ lorsque l'entièreté du corpus est utilisé. Quant aux étiquettes lexicales, l'utilisation de plus grands corpus améliore également les résultats.

6.6 Vers le multilinguisme

Jusque là, nous n'avons eu recours qu'aux données de supervision de la langue étudiée, ce qui rend notre modèle fortement dépendant de l'avancée de la documentation de la langue. Il existe pourtant des corpus de textes glosés, au format IGT, comme présenté en section 3.1.2, ou des langues de la même famille, plus documentées. Les techniques de transfert entre langues en recourant à un pré-entraînement multilingue (Conneau et al., 2020) sont de plus en plus communes en TAL et ont été utilisées dans d'autres contextes, comme l'analyse morphologique de langues

peu dotées (Anastasopoulos et Neubig, 2019). Utiliser directement de telles ressources, à la manière d'autres tâches de TAL, est cependant difficile du fait des différences entre les langues et la variabilité des annotations, comme énoncé en section 2.3.4. Nous nous intéressons donc ici à une approche permettant d'apprendre des caractéristiques généralisées à travers plusieurs langues, dans l'optique d'améliorer la qualité des prédictions sur les langues en cours de documentation. Nous étudions alors dans quelle mesure le modèle parvient à apprendre des caractéristiques multilingues.

6.6.1 Pré-entraînement sur un corpus multilingue

Une première approche est d'avoir recours à un corpus multilingue de gloses pour pré-entraîner notre modèle. Nous choisissons IMTVault (Nordhoff et Krämer, 2022), corpus plus petit et récent qu'ODIN mais contenant moins de bruit (voir section 3.1.2 pour plus de détails). Parmi toutes les langues présentes, nous considérons uniquement celles qui sont clairement identifiées par un code et de plus de 30 phrases, en reproduisant les conditions du gitksan pour lequel nous avons 31 phrases d'entraînement. Une autre raison derrière ce filtrage est que les gloses sont fortement liées aux langues, rendant l'apport de ces quelques phrases peu utile dans notre cas, voire nuisible. Il nous reste alors 173 langues et 34 815 phrases, dont nous utilisons les 30 000 premières pour superviser notre modèle, puis 2 000 phrases pour le développement et le test respectivement.

En présence de différentes langues et phénomènes grammaticaux, l'étude des gloses directes, grammaticales ou lexicales, paraît peu pertinente ; les étiquettes grammaticales seules constituent également un nombre trop élevé. C'est pourquoi, l'objectif de notre modèle est de prédire les étiquettes binaires GRAM et LEX, suffisamment générales, afin d'apprendre des tendances partagées par plusieurs langues.

Pour ce faire, le modèle est pré-entraîné en employant uniquement les caractéristiques multilingues, indiquées par un astérisque dans le tableau récapitulatif des caractéristiques 6.12. Il s'agit donc de tests locaux sur les morphèmes en ignorant leur forme orthographique. Une fois le pré-entraînement effectué, notre modèle utilise les poids appris sur le corpus multilingue comme point de départ, ce qui n'affecte pas les autres caractéristiques ; S3 est par la suite entraîné comme dans les expériences précédentes, avec l'entièreté des patrons présentés en section 6.4.3. Le tableau 6.17 présente les résultats ainsi obtenus sur les cinq langues étudiées.

niveau	mot					morphème				
langue	ddo	git	lez	ntu	usp	ddo	git	lez	ntu	usp
CRF (S3)	85,6	33,6	82,8	89,1	78,9	91,9	52,4	87,0	92,8	84,4
+ IMT	85,3	33,1	83,4	89,0	79,0	91,8	52,8	87,1	92,5	84,5

TABLE 6.17 – Exactitude au niveau des mots et des morphèmes avec et sans pré-entraînement multilingue sur IMTVault.

Les améliorations sont dans l'ensemble décevantes, négligeables dans le meilleur des cas, dommageables sinon. Il semble donc que les caractéristiques multilingues, généralisables, sont déjà apprises par le modèle pour la langue étudiée à travers les données de supervision, et ce, même à partir de quelques centaines de phrases. Comme la variable binaire b est relativement facile à prédire, en fin de compte, le pré-entraînement n'apporte qu'un effet marginal dans l'ensemble.

6.6.2 Transfert cross-lingue entre langues de la même famille

Une autre approche de pré-entraînement consiste en l'utilisation de transfert de connaissances entre les langues. Parmi nos corpus, nous avons des données concernant deux langues de la même famille nakho-daghestanienne : le tsez et le lezghien. Étant donné la quantité de données, nous utilisons le tsez pour le pré-entraînement et expérimentons sur le lezghien.

Notons qu'il s'agit là d'une configuration déjà expérimentée chez Zhao et al. (2020), qui observaient une amélioration nette des métriques dans le même sens tsez-lezghien. Dans leurs travaux, le lezghien est translitteré afin de bénéficier davantage du transfert cross-lingue, pour mieux identifier notamment des mots de même racine. Ici, nous n'effectuons pas ce traitement étant donné que nous n'utilisons pas la forme orthographique, mais des caractéristiques délexicalisées. Il s'agit en revanche d'une piste future à considérer, où le tsez serait impliqué directement dans la prédiction du lezghien translitteré, par exemple.

En adoptant la même méthodologie que dans la section précédente pour le pré-entraînement, nous observons une exactitude de 83,3 pour les mots et de 87,1 pour les morphèmes. Les scores sont donc très proches de ceux obtenus avec IMTVault, bien que les données de pré-entraînement du tsez soient de plus petite taille.

6.6.3 Avec très peu de ressources

Enfin, nous étudions le cas où les phrases d'entraînement sont disponibles en très petite quantité, notamment au début de la documentation. Intuitivement, ces premières étapes semblent être les mieux à même de bénéficier de ressources externes, pour compenser le manque de données de supervision. Nous réduisons donc le nombre de phrases données en entraînement à 50 pour toutes les langues, sauf pour le cas particulier du gitksan, que nous ne traitons pas ici, et comparons l'apport du pré-entraînement dans cette configuration. Le tableau 6.18 présente les résultats associés.

niveau	mot				morphème			
langue	ddo	lez	ntu	usp	ddo	lez	ntu	usp
CRF (S3)	47,6	53,7	64,9	45,9	66,9	63,3	74,7	57,5
+ IMT	48,0	54,0	65,6	48,3	67,3	63,4	75,3	59,2
Δ	0,4	0,3	0,7	2,4	0,4	0,1	0,6	1,7

TABLE 6.18 – Exactitude au niveau des mots et des morphèmes avec et sans pré-entraînement multilingue sur les jeux de test réduits à 50 phrases.

Nous remarquons que le bénéfice du pré-entraînement apparaît de manière sensiblement plus visible ici, avec des améliorations allant de 0,3 à 2,4 points au niveau des mots et de 0,1 à 1,7 pour les morphèmes. Il faut noter que nombre d'unités présentes à l'entraînement semble être un facteur influent dans l'effet de ce procédé. De fait, l'uspanteko en bénéficie davantage parce qu'il n'y a que 4,3 mots par phrases, contrairement aux trois autres langues, avec plus de 10 mots par phrases. Si le choix de comparer les langues sur le critère du nombre de phrases est donc discutable dans cette expérience, il s'agit de l'unité naturelle de constitution de corpus. Cela permet entre autres d'observer qu'avec très peu de mots disponibles, l'apport du pré-entraînement est d'autant plus visible.

6.7 Conclusion

Nous avons abordé la tâche de génération automatique des gloses à travers une approche classique d'étiquetage de séquences, basée sur les CRF, en introduisant progressivement les difficultés. Si la classification binaire de la nature des morphèmes est aisée, la prédiction des gloses grammaticales ajoute une complexité, principalement par la taille du jeu d'étiquettes. La proximité avec l'étiquetage en parties du discours, tâche plus connue, a révélé une complexité similaire.

Le défi principal de la tâche provient de la variété des gloses lexicales. En s'appuyant sur les travaux antérieurs fondés sur des CRF, nous avons abordé la génération de gloses par une approche hybride entre les tâches d'étiquetage de séquences classique et d'alignement. Elle repose sur l'hypothèse, déjà présente chez McMillan-Major (2020) et Zhao et al. (2020), que les gloses lexicales puissent être retrouvées à partir de la traduction dans la langue de documentation.

Comme les alignements de référence entre ces deux unités ne sont pas disponibles, nous avons recours à un modèle d'alignement neuronal, SimAlign, afin de pouvoir superviser l'entraînement. Pour nous assurer de leur utilité et pertinence, nous avons évalué les liens d'alignement automatiques obtenus, d'une part, vis-à-vis de leur couverture du corpus et, d'autre part, par rapport à leur concordance avec la référence. Pour cette dernière, nous les avons, en outre, estimés manuellement et elles paraissent proches d'un jugement humain en général.

Notre approche de génération de gloses consiste en la réutilisation de Lost, un modèle de traduction statistique à l'origine, basé sur la théorie des CRF. Il permet notamment de moduler localement l'ensemble des étiquettes possibles, ce qui nous laisse choisir des étiquettes candidats depuis les mots de la traduction ainsi que depuis un dictionnaire de la langue, afin de bénéficier des deux sources d'informations lexicales. À travers des étiquettes structurées, faisant intervenir la nature de la glose ou les parties du discours, ainsi que des caractéristiques généralisables, nous avons œuvré pour augmenter la robustesse du modèle afin de traiter au mieux les morphèmes peu voire jamais observés.

Cette thèse présente les résultats de ce modèle, obtenus sur les cinq langues du défi partagé SIGMORPHON 2023, que sont le tsez (`ddo`), le gitksan (`git`), le lezghien (`lez`), le natugu (`ntu`) et l'uspanteko (`usp`), aux corpus de tailles variées. Le modèle avec lequel nous avons participé a atteint les meilleurs résultats au niveau des mots pour deux de ces langues, avec relativement peu de données. La version mise à jour ici présente des résultats sensiblement plus élevés pour certaines langues.

En s'appuyant sur des caractéristiques indépendantes de la langue, comme la longueur en caractères ou le patron consonne-voyelle du morphème, nous avons également étudié le pré-entraînement de notre modèle, dans une optique ouvrant vers le transfert cross-lingue, basé sur les corpus multilingues comme IMTVault (Nordhoff et Krämer, 2022) ou des données de langues de la même famille. Les résultats sont pour le moment décevants dans l'absolu, mais semblent être une première étape pour compenser le manque de données au début de la documentation. En effet, dans l'ensemble, nous obtenons des prédictions qui restent meilleures que les gloses proposées par un système de lexique, utilisé dans les outils d'annotation actuellement, et ce, à partir de très peu de données d'entraînement. L'intégration dans les logiciels utilisés par les linguistes semble donc être une option pertinente pour le futur.

Un des leviers pour améliorer le modèle repose sur les alignements. Malgré un impact qui ne sera probablement pas déterminant, car les liens obtenus sont déjà majoritairement suffisants, une meilleure qualité permet de mieux prédire les informations connexes, comme la partie du discours,

et une extension aux gloses grammaticales, comme initié par Georgi (2016), pourrait augmenter leur robustesse.

Un point faible de notre approche réside toujours dans les gloses lexicales, à travers la contrainte locale des caractéristiques bigrammes. L'affectation des étiquettes se fait donc notamment sur une fenêtre de deux gloses et n'observe pas les prédictions antérieures. C'est pourquoi le modèle obtient, en particulier avec très peu de données comme en gitksan, des étiquettes lexicales répétées au sein d'une même phrase. L'utilisation d'un post-traitement déterministe pour réaffecter certaines étiquettes lexicales, afin d'éviter ces doublons, est par exemple une piste à explorer.

Chapitre 7

Conclusion

Dilegua, o notte! Tramontate, stelle!
All'alba vincerò! Vincerò!

Acte III, *Turandot*, Puccini

7.1 Bilan

Face au risque de voir disparaître plus de la moitié des langues dans les décennies à venir, les linguistes se consacrent à leur enregistrement, annotation et archivage à travers un processus principalement manuel, nécessitant également une expertise linguistique et surtout un engagement des locuteurs, et donc coûteux. La documentation automatique des langues vise alors à faciliter et accélérer le traitement de ces données. C'est dans ce contexte que nous avons étudié deux tâches fondamentales qui s'inscrivent dans le processus standard de documentation linguistique : la segmentation de séquences (en mots et en morphèmes) ainsi que la production automatique de gloses. La première intervient après une transcription (presque) phonétique de l'enregistrement, afin d'obtenir dans un premier temps des phrases segmentées en mots. La tâche de segmentation en morphèmes constitue l'étape suivante pour avoir une phrase en langue source entièrement segmentée. L'autre tâche génère ensuite automatiquement des annotations linguistiques pour chacun de ces morphèmes. Celles-ci se divisent en deux catégories : soit elles indiquent son rôle grammatical à partir d'un jeu d'étiquettes pré-défini, soit elles expriment sa signification dans la langue de documentation, permettant de rendre la phrase compréhensible au plus grand nombre de lecteurs.

Comme les langues en cours de documentation sont fréquemment à tradition orale et peu présentes sur Internet, l'accès aux données se fait principalement en s'appuyant sur les travaux des linguistes. Nous avons donc présenté les collections de ressources disponibles à l'heure actuelle, en soulignant leurs particularités.

Dans l'optique d'améliorer les performances des modèles de segmentation en mots, nous avons intégré des ressources accessibles de manière réaliste lors des projets de documentation de langues. En effet, il arrive souvent que les linguistes aient déjà travaillé sur la langue et disposent de données, comme des phrases annotées ou des lexiques, qui étaient ignorées jusque là dans les modèles de segmentation complètement non supervisés. Pour les langues très peu dotées, la meilleure approche de segmentation s'appuie sur des modèles bayésiens non paramétriques ; nous réimplémentons donc un des modèles aux résultats stables, `dpseg`, et incorporons ces ressources additionnelles à travers différentes méthodes de supervision faible. Dans le cas où des phrases déjà segmentées sont disponibles, l'utilisation d'un échantillonnage de Gibbs semi-supervisé permet d'améliorer significativement la qualité de segmentation. Nous avons par la même occasion

observé que l'utilisation des informations de frontières seules ne suffit pas ; le meilleur scénario de supervision semble être de regrouper les informations de frontières et de non frontières sur quelques phrases, à travers une annotation dense. De plus, lorsqu'un lexique est disponible, la méthode optimale est d'adapter le modèle de langue, en se basant sur les successions bigrammes de caractères dans les mots de supervision, et d'augmenter la probabilité de génération de ces derniers dans `dpseg`.

En outre, en prévision d'une intégration dans un outil d'annotation, nous avons également simulé une situation d'apprentissage incrémental, où un expert de la langue corrigerait progressivement les segmentations prédites, améliorant par là le modèle. Nous avons observé une baisse régulière du taux d'erreur et constaté l'importance d'un meilleur modèle de langue, qui permet d'augmenter plus rapidement la qualité de segmentation.

Enfin, nous avons analysé la nature des unités identifiées par `dpseg` : ces dernières semblent en effet à mi-chemin entre des mots et des morphèmes. Si l'utilisation de la supervision faible peut pallier partiellement cette tendance à sur-segmenter, inhérente au modèle, ces limites nous amènent tout de même à considérer l'introduction d'un second niveau de segmentation pour permettre une différenciation entre ces deux types d'unités.

En nous basant sur le même modèle, `dpseg`, nous avons donc conçu plusieurs types d'approches de segmentation simultanée en mots et en morphèmes. Au-delà d'un modèle en cascade segmentant d'abord en mots puis en morphèmes, nous avons développé deux catégories de modèles. La première met en parallèle deux modèles de segmentation, un par niveau, et repose sur une règle pour assurer la cohérence aux deux niveaux. La seconde est une approche hiérarchique, où la probabilité de générer un mot est obtenue par la probabilité de chacun de ses morphèmes constitutifs. De plus, à la lumière des résultats obtenus précédemment par l'incorporation de ressources supplémentaires par `dpseg`, nous appliquons les meilleures stratégies de supervision faible.

Toutefois, dans toutes les situations testées, nous n'observons pas d'amélioration suffisamment notable, ce qui nous amène à nous interroger sur les limites de nos méthodes dans la configuration employée. Nous constatons en effet que les corpus issus de la documentation des langues présentent des distributions d'unités particulières qui ne correspondent pas exactement aux hypothèses implicites des modèles de segmentation que nous utilisons, à savoir une distribution en loi de puissance. Ceci tient de leur nature même, où les mots sont volontiers plus répétés que dans un corpus spontané, par exemple. Dans tous les cas, les expériences soulignent la difficulté de déduire les mots par rapport aux morphèmes sur la seule base des statistiques de fréquences.

À cette étape, en théorie, nous sommes en présence d'une phrase source segmentée en mots et en morphèmes ; la phase suivante s'attelle à l'annotation linguistique de chacun de ses morphèmes : la génération de gloses. Il s'agit d'un processus coûteux en temps et en expertise : le linguiste effectue cette tâche avec peu d'automatisation pour le moment. Nos expériences ont suggéré la similarité de la tâche avec celle d'étiquetage en parties du discours.

Dans le cas principal où nous prédisons également les gloses lexicales, nous les choisissons notamment à partir de la traduction, où nous supposons, comme dans certains travaux antérieurs, qu'elles y sont présentes. À ce sujet, comme nous n'observons pas les liens entre les morphèmes sources et la traduction, nous avons recours à des alignements automatiques, lors de l'apprentissage, en reposant sur les gloses comme pont entre les deux langues. Nous avons donc étudié les liens d'alignement obtenus entre les gloses lexicales d'une part et la traduction d'autre part, afin de vérifier si notre hypothèse était bien valable. De manière générale, nous avons constaté que les alignements automatiques permettent de recouvrir la plupart des gloses lexicales dans la phrase traduite, soit directement, soit par un synonyme, avec plus ou moins de bruits selon la méthode

employée. Enfin, en ré-utilisant un modèle de traduction statistique, Lost, étendant les CRF, nous avons pu restreindre l'ensemble des étiquettes possibles pour l'adapter dynamiquement à chaque phrase. En effet, nous proposons, au-delà des gloses grammaticales, communes à tout le corpus, des étiquettes lexicales issues de la traduction ainsi que d'un dictionnaire, constitué des précédentes occurrences observées pour les morphèmes sources.

En expérimentant ce modèle sur les langues du défi partagé SIGMORPHON 2023, nous obtenons des résultats compétitifs, voire supérieurs aux meilleurs scores publiés officiellement. En particulier, pour les langues avec le moins de données supervisées, notre approche statistique semble plus performante. En nous appuyant sur la traduction comme entrée supplémentaire, nous pouvons également retrouver une partie des mots jamais observés durant l'apprentissage. En complément, nous implémentons aussi un système de pré-entraînement où des données glosées multilingues ou d'une langue de la même famille sont utilisées. Lorsque très peu de phrases de supervision sont disponibles, nous observons alors une amélioration grâce au pré-entraînement, ce qui est un résultat prometteur pour l'application de cette méthode, à un stade relativement précoce de documentation.

Cette thèse confirme ainsi l'impact et surtout l'intérêt d'utiliser les ressources auxiliaires disponibles au cours de la documentation, comme les phrases déjà annotées, les dictionnaires ou les données d'autres langues notamment. Comme la quantité de données reste le facteur limitant majeur dans l'application des techniques de TAL pour les langues en cours de documentation, le recours à une supervision faible permet d'atténuer ce problème.

Enfin, pour bien clore cette thèse, il aurait été envisageable, en théorie, de participer également à la piste fermée du défi partagé, à travers une approche en cascade de segmentation en morphèmes puis d'étiquetage de gloses. Toutefois, elle n'a pas été effectuée du fait des risques élevés de propagation d'erreurs ; du fait des biais identifiés pour `dpseg`, notamment la génération insuffisante de segments rares, l'analyse en morphèmes obtenue n'aurait pas été de qualité convaincante pour la prédiction des gloses.

Limites Tout d'abord, nos approches pour les deux tâches ont un coût de calcul non négligeable vis-à-vis de la quantité de données. Par exemple, `dpseg` doit itérer plusieurs dizaines de milliers de fois sur tout le corpus ; plus il y a de phrases, plus le temps de calcul devient long. Il existe des méthodes pour accélérer ce calcul, comme présenté dans la section 2.2.4, avec l'échantillonnage de Gibbs bloqué, par exemple. Nous soulignons cependant que les méthodes présentées ici sont principalement destinées aux étapes initiales de documentation, lorsque la quantité de données n'est pas encore suffisante pour entraîner les modèles neuronaux. En effet, une fois ce seuil franchi, ces derniers semblent plus aptes à exploiter les données et prendront le relais des méthodes présentées ici.

En outre, pour la tâche de segmentation en particulier, il est également possible d'appliquer nos approches sur des modèles plus sophistiqués, comme les *Adaptor Grammars* (Johnson et al., 2007), en se basant sur certains travaux récents. De fait, à la manière des méthodes faiblement supervisées de (Sirts et Goldwater, 2013) ou de (Eskander et al., 2021), les données de supervision que nous avons considérées, aussi bien sous forme de phrases segmentées que de listes d'unités, peuvent être utilisées pour fournir un lexique d'affixes et éventuellement de racines. Précisons que pour le moment, nos expériences se sont focalisées sur des ressources extraites directement depuis le corpus à segmenter, ce qui est une situation idéale : tous les mots présents dans le dictionnaire le sont également dans le texte et aucun autre mot n'y apparaît. Il est donc souhaitable d'observer l'impact d'un lexique constitué à partir d'autres ressources pour évaluer la robustesse de la

méthode. Ce constat rejoint également le point mentionné en section 4.3.2 concernant l'utilisation d'une liste de formes fléchies et non un dictionnaire de lemmes.

En ce qui concerne la segmentation, les métriques indiquent une performance assez faible, en particulier pour les types. Dans un cadre d'utilisation réel, les segments peuvent donc ne pas être pertinents, à cause de la sur-segmentation entre autres. De plus, notre approche suppose une chaîne de caractères de référence, intégralement correcte, mais non segmentée ; or les outils automatiques de transcription de la parole auront inévitablement une part de bruit. La robustesse de notre modèle face aux variations, même minimes, semble relativement compromise ; Godard et al. (2018c) observent notamment des performances plus dégradées avec `dpseg` qu'une approche neuronale, lorsque des sorties d'un système de reconnaissance d'unité acoustique sont utilisées. De même, ces erreurs sont également la raison pour laquelle une approche intégrée serait à privilégier dans un cadre réaliste d'utilisation ; il est peu probable qu'un linguiste ne corrige que les phonèmes sans simultanément identifier les frontières de mots.

7.2 Perspectives

Vers un usage réel La finalité de la documentation automatique des langues est d'aider les annotations des linguistes, en pré-traitant les données. Pour ce faire, une première manière est alors d'intégrer nos modèles dans des outils utilisés pour l'annotation comme ELAN (Wittenburg et al., 2006). L'apprentissage incrémental, qui a été expérimenté en vue d'une possible utilisation réelle au chapitre 4, convient en ce sens. La qualité de segmentation est effectivement améliorée, au fur et à mesure des corrections. Notons que notre modèle est relativement peu coûteux et se prête bien à ce type d'approche, avec uniquement des itérations d'échantillonnage de Gibbs à effectuer régulièrement pour accélérer la propagation des corrections.

Il reste toutefois des questions à élucider, notamment sur le choix des phrases à présenter à l'annotateur, ce qui impacte de manière non négligeable les performances. Dans le cadre d'un apprentissage actif, Palmer et al. (2009) observent entre autres que l'approche séquentielle, à savoir de suivre l'ordre des phrases du corpus, reste l'option la moins optimale, par rapport à celle aléatoire (que nous avons utilisée) ou celle présentant les phrases les plus difficiles pour le modèle.

De plus, concernant les gloses, notre modèle de génération automatique permet d'obtenir une première version d'annotation, pour être éditée manuellement par un expert dans un second temps. Elle pourrait donc être également intégrée aux outils d'annotations et, par exemple, agir en complément du système de dictionnaire déjà implémenté.

Il serait donc utile d'évaluer qualitativement les prédictions de nos modèles, afin d'estimer leur pertinence concrète lors de la documentation. Dans quelle mesure nos pré-traitements (aussi bien pour la segmentation que la génération des gloses) constituent une aide en termes de temps d'annotation mais également de pertinence ?

Ce travail s'inscrit ainsi dans la continuité des travaux de collaboration entre le domaine du TAL et les linguistes. L'intérêt est mutuel, entre d'une part, des systèmes prometteurs pour assister les linguistes dans leurs travaux de documentation et d'autre part, des ressources ainsi que des connaissances ciblées pour les langues étudiées afin d'améliorer les modèles.

La nature des unités Notons ici que l'automatisation permet également d'obtenir un autre point de vue que celui des linguistes. En fin de chapitre 5 émerge particulièrement la question sur la nature des unités segmentées : qu'est-ce qu'un mot ? Les segments obtenus sur la seule base

d'informations statistiques peuvent éventuellement apporter une vision différente et complémentaire sur ce point. En effet, à la manière de (Godard et al., 2018b), où les résultats du modèle permettent de soutenir l'incorporation des consonnes complexes dans la phonologie de la langue, des travaux de TAL peuvent servir de supplément pour mettre en évidence des phénomènes linguistiques.

Par ailleurs, dans un autre registre, une piste d'amélioration serait d'incorporer dans les modèles de segmentation, la distinction entre les différents types d'unités, à la manière de certains travaux cités en section 2.2 comme (Löser et Allauzen, 2016) ou l'*Adaptor Grammar*. En effet, pour le moment, nos modèles ne différencient pas les verbes des noms ou les affixes des racines.

Au-delà des transcriptions D'un point de vue pratique, au vu du succès récent des approches de transcription, même pour les langues très peu dotées comme nous avons vu en section 2.2.7, un modèle intégré de transcription et de segmentation de mots peut être préférable, à la manière de (Guillaume et al., 2022). En ayant accès directement aux signaux sonores, la prédiction des frontières de mots semble plus aisée, bien que toutes les unités ne bénéficient pas d'une pause ou d'une prosodie marquée. Ce type d'approche permet également d'être plus facilement déployé dans un cas réel, car il est peu fréquent d'avoir une transcription corrigée manuellement mais sans segmentation, comme nous l'avons mentionné dans les limites de nos travaux.

Par ailleurs, comme les langues traitées sont principalement orales, il existe également des initiatives pour moins reposer sur les transcriptions et mieux utiliser les ressources disponibles, orales comme écrites (notamment les traductions et les lexiques), comme le « modèle de transcription éparse » (*sparse transcription model* en anglais) de (Bird, 2021). La motivation principale est d'aborder le problème du « goulot d'étranglement dû aux transcriptions » (*transcription bottleneck* en anglais) qui est jusque là causé, selon lui, par trois postulats : la nécessité de transcrire des phones, de manière complète et avant toute chose. En envisageant le problème sous un autre angle, Bird (2021) propose une transcription ciblée sur quelques unités acoustiques dans les enregistrements, en fonction de l'intérêt notamment linguistique. Cette démarche permet également une collaboration plus efficace avec les locuteurs, qui peuvent aider à améliorer la reconnaissance de certains mots par exemple (Le Ferrand et al., 2020).

Approches neuronales À l'ère des grands modèles de langues et des réseaux de neurones, cette thèse n'aborde pas cet aspect, en se concentrant principalement sur des approches statistiques. S'il est bien difficile d'utiliser directement les architectures neuronales sur des données de langues en cours de documentation, principalement du fait de leur taille, des travaux ont déjà abordé la question, comme nous l'avons vu, en section 2.2.6 pour la segmentation en mots et en section 2.3.3 pour la génération automatique de gloses. Pour le moment, les résultats y étaient moins performants que les approches statistiques, particulièrement lorsque les données sont en petite quantité. Néanmoins, les avancées récentes sur les méthodes neuronales pour les langues très peu dotées sont prometteuses ; par exemple, ImaniGooghari et al. (2023) créent un grand modèle de langue pré-entraîné pour 511 langues, pour lesquelles il existe au moins 30 000 phrases. Ce point rejoint également un autre levier utilisé en TAL qui est de recourir à des modèles multilingues, comme les langues que nous traitons sont très peu dotées. L'impact du transfert entre les langues, en ce sens, semble bénéfique dans la tâche de génération de gloses (Zhao et al., 2020) et reste une piste d'amélioration privilégiée.

De plus, certaines langues, assez avancées dans la documentation par rapport à d'autres, présentent des corpus et des ressources suffisamment conséquentes pour permettre leur utilisation :

pour la génération de gloses, c'est notamment le cas de l'arapaho (arp), qui dispose du plus grand jeu d'entraînement dans le défi partagé SIGMORPHON (Ginn et al., 2023), ou le chintang (ctn), à travers le *Chintang language research program*. De fait, au fil de la documentation, les données s'accroissent et pour améliorer l'adaptabilité des modèles, les approches neuronales constituent alors une option privilégiée à considérer, surtout pour le stade après avoir utilisé les modèles statistiques.

Enfin, comme observé en section 2.3.1, les gloses peuvent servir de pont entre la langue source et cible, permettant notamment de projeter la structure d'une phrase à une autre. Ces alignements peuvent alors être utilisés pour constituer des lexiques de manière plus aisée. En recourant par exemple à la méthode de (Wang et al., 2022), il serait alors possible d'étendre des modèles pré-entraînés multilingues pour les langues peu dotées, en se basant sur ces lexiques (monolingues et bilingues) induits.

Bibliographie

Emile Aarts et Jan Korst. 1989. *Simulated Annealing and Boltzmann Machines : A Stochastic Approach to Combinatorial Optimization and Neural Computing*. John Wiley & Sons, Inc., USA. *[page 20]*

Asen' K. Abdulaev et I. K. Abdulaev. 2010. *Cezjas fol'klor : (gíurus mecrek°iorno butirno) = Dido (Tsez) folklore = Didojskij (cezskij) fol'klor*. Lotos, Leipzig. *[page 50]*

Asen' K. Abdulaev, I.K. Abdullaev, André Müller, Evgeniya Zhivotova et Bernard Comrie. 2022. The Tsez Annotated Corpus Project. *[page 51]*

Oliver Adams. 2017. *Automatic Understanding of Unwritten Languages*. Ph.D. thesis, The University of Melbourne. *[page 4]*

Oliver Adams, Trevor Cohn, Graham Neubig, Hilaria Cruz, Steven Bird et Alexis Michaud. 2018. Evaluation phonemic transcription of low-resource tonal languages for language documentation. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA). *[pages 3 et 4]*

Oliver Adams, Trevor Cohn, Graham Neubig et Alexis Michaud. 2017. Phonemic transcription of low-resource tonal languages. In *Proceedings of the Australasian Language Technology Association Workshop 2017*, pages 53–60, Brisbane, Australia. *[page 3]*

Oliver Adams, Benjamin Galliot, Guillaume Wisniewski, Nicholas Lambourne, Ben Foley, Rahasya Sanders-Dwyer, Janet Wiles, Alexis Michaud, Séverine Guillaume, Laurent Besacier, Christopher Cox, Katya Aplonova, Guillaume Jacques et Nathan Hill. 2021. User-friendly automatic transcription of low-resource languages: Plugging ESPnet into elpis. In *Proceedings of the 4th Workshop on the Use of Computational Methods in the Study of Endangered Languages Volume 1 (Papers)*, pages 51–62, Online. Association for Computational Linguistics. *[pages 4 et 50]*

Gilles Adda, Sebastian Stüker, Martine Adda-Decker, Odette Ambouroue, Laurent Besacier, David Blachon, Hélène Bonneau-Maynard, Pierre Godard, Fatima Hamlaoui, Dmitri Idiatov, Guy-Noël Kouarata, Lori Lamel, Emmanuel-Moselly Makasso, Annie Rialland, Mark Van de Velde, François Yvon et Sabine Zerbian. 2016. Breaking the Unwritten Language Barrier: The Bulb Project. In *Proceedings of SLTU (Spoken Language Technologies for Under-Resourced Languages)*, Yogyakarta, Indonesia. *[pages 4, 11, 15, 22, 25, 26, 43 et 48]*

Mohamed Afify, Ruhi Sarikaya, Hong-Kwang Jeff Kuo, Laurent Besacier et Yuqing Gao. 2006. On the use of morphological analysis for dialectal Arabic speech recognition. In *Proc. Interspeech 2006*, pages paper 1444–Mon2A2O.2. *[page 28]*

Célestin Amboulou. 1998. *Le Mbochi, langue bantou du Congo-Brazzaville : étude descriptive*. Ph.D. thesis, INALCO. Thèse de doctorat dirigée par Gérard Philippson Linguistique Paris. *[page 48]*

Jonathan D. Amith, Jiatong Shi et Rey Castillo García. 2021. End-to-end automatic speech recognition: Its impact on the workflowin documenting yoloxóchitl Mixtec. In *Proceedings of the First Workshop on Natural Language Processing for Indigenous Languages of the Americas*, pages 64–80, Online. Association for Computational Linguistics. *[page 27]*

Antonios Anastasopoulos. 2019. *Computational Tools for Endangered Language Documentation*. Ph.D. thesis, University of Notre Dame. *[pages 4, 44 et 49]*

Antonios Anastasopoulos, Sameer Bansal, David Chiang, Sharon Goldwater et Adam Lopez. 2017. Spoken term discovery for language documentation using translations. In *Proceedings of the Workshop on Speech-Centric Natural Language Processing*, pages 53–58, Copenhagen, Denmark. Association for Computational Linguistics. *[page 4]*

Antonios Anastasopoulos et David Chiang. 2018. Leveraging Translations for Speech Transcription in Low-resource Settings. In *Proc. Interspeech 2018*, pages 1279–1283. *[pages 4 et 49]*

Antonios Anastasopoulos et Graham Neubig. 2019. Pushing the limits of low-resource morphological inflection. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 984–996, Hong Kong, China. Association for Computational Linguistics. *[page 124]*

Galen Andrew et Jianfeng Gao. 2007. Scalable training of l1-regularized log-linear models. In *Proceedings of the 24th International Conference on Machine Learning*, ICML '07, page 33–40, New York, NY, USA. Association for Computing Machinery. *[page 82]*

Peter K. Austin et Julia Sallabank. 2011. *The Cambridge Handbook of Endangered Languages*. Cambridge Handbooks in Language and Linguistics. Cambridge University Press, Cambridge. *[page 1]*

R. Harald Baayen. 2001. *Word Frequency Distributions*, volume 18 of *Text, Speech and Language Technology*. Springer Netherlands, Dordrecht. *[pages 93 et 94]*

Dzmitry Bahdanau, Kyunghyun Cho et Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR'15)*. *[pages 29 et 31]*

Jason Baldridge et Alexis Palmer. 2009. How well does active learning *actually* work? Time-based evaluation of cost-reduction strategies for language documentation. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 296–305, Singapore. Association for Computational Linguistics. *[pages 13, 36, 40 et 42]*

Timothy Baldwin, William Croft, Joakim Nivre et Agata Savary. 2021. Universals of Linguistic Idiosyncrasy in Multilingual Computational Linguistics (Dagstuhl Seminar 21351). *Dagstuhl Reports*, 11(7) :89–138. *[page 98]*

Diego Barriga Martínez, Victor Mijangos et Ximena Gutierrez-Vasques. 2021. Automatic interlinear glossing for Otomi language. In *Proceedings of the First Workshop on Natural Language Processing for Indigenous Languages of the Americas*, pages 34–43, Online. Association for Computational Linguistics. *[pages 37, 38 et 40]*

Khuyagbaatar Batsuren, Gábor Bella, Aryaman Arora, Viktor Martinovic, Kyle Gorman, Zdeněk Žabokrtský, Amarsanaa Ganbold, Šárka Dohnalová, Magda Ševčíková, Kateřina Pelegrinová, Fausto Giunchiglia, Ryan Cotterell et Ekaterina Vylomova. 2022a. The SIGMORPHON 2022 shared task on morpheme segmentation. In *Proceedings of the 19th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 103–116, Seattle, Washington. Association for Computational Linguistics. *[page 28]*

Khuyagbaatar Batsuren, Omer Goldman, Salam Khalifa, Nizar Habash, Witold Kieraś, Gábor Bella, Brian Leonard, Garrett Nicolai, Kyle Gorman, Yustinus Ghanggo Ate, Maria Ryskina, Sabrina Mielke, Elena Budianskaya, Charbel El-Khaissi, Tiago Pimentel, Michael Gasser, William Abbott Lane, Mohit Raj, Matt Coler, Jaime Rafael Montoya Samame, Delio Siticonatzi Camaiteri, Esaú Zumaeta Rojas, Didier López Francis, Arturo Oncevay, Juan López Bau-

tista, Gema Celeste Silva Villegas, Lucas Torroba Hennigen, Adam Ek, David Guriel, Peter Dirix, Jean-Philippe Bernardy, Andrey Scherbakov, Aziyana Bayyr-ool, Antonios Anastasopoulos, Roberto Zariquiey, Karina Sheifer, Sofya Ganieva, Hilaria Cruz, Ritván Karahóǧa, Stella Markantonatou, George Pavlidis, Matvey Plugaryov, Elena Klyachko, Ali Salehi, Candy Angulo, Jatayu Baxi, Andrew Krizhanovsky, Natalia Krizhanovskaya, Elizabeth Salesky, Clara Vania, Sardana Ivanova, Jennifer White, Rowan Hall Maudslay, Josef Valvoda, Ran Zmigrod, Paula Czarnowska, Irene Nikkarinen, Aelita Salchak, Brijesh Bhatt, Christopher Straughn, Zoey Liu, Jonathan North Washington, Yuval Pinter, Duygu Ataman, Marcin Wolinski, Totok Suhardijanto, Anna Yablonskaya, Niklas Stoehr, Hossep Dolatian, Zahroh Nuriah, Shyam Ratan, Francis M. Tyers, Edoardo M. Ponti, Grant Aiton, Aryaman Arora, Richard J. Hatcher, Ritesh Kumar, Jeremiah Young, Daria Rodionova, Anastasia Yemelina, Taras Andrushko, Igor Marchenko, Polina Mashkovtseva, Alexandra Serova, Emily Prud'hommeaux, Maria Nepomniashchaya, Fausto Giunchiglia, Eleanor Chodroff, Mans Hulden, Miikka Silfverberg, Arya D. McCarthy, David Yarowsky, Ryan Cotterell, Reut Tsarfaty et Ekaterina Vylomova. 2022b. UniMorph 4.0: Universal Morphology. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 840–855, Marseille, France. European Language Resources Association. *[page 47]*

Roch Paulin Beapami, Ruth Chatfield, Guy-Noël Kouarata et Andrea Embengue Waldschmidt. 2000. *Dictionnaire Mbochi-Français*. SIL-Congo Publishers, Brazzaville. *[pages 48 et 49]*

Diego Bear et Paul Cook. 2022. Evaluating unsupervised approaches to morphological segmentation for wolastoqey. In *Proceedings of the 1st Annual Meeting of the ELRA/ISCA Special Interest Group on Under-Resourced Languages*, pages 155–160, Marseille, France. European Language Resources Association. *[page 30]*

Emily M. Bender, Joshua Crowgey, Michael Wayne Goodman et Fei Xia. 2014. Learning grammar specifications from IGT: A case study of chintang. In *Proceedings of the 2014 Workshop on the Use of Computational Methods in the Study of Endangered Languages*, pages 43–53, Baltimore, Maryland, USA. Association for Computational Linguistics. *[page 34]*

Emily M. Bender, Michael Wayne Goodman, Joshua Crowgey et Fei Xia. 2013. Towards creating precision grammars from interlinear glossed text: Inferring large-scale typological properties. In *Proceedings of the 7th Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*, pages 74–83, Sofia, Bulgaria. Association for Computational Linguistics. *[page 34]*

Yoshua Bengio, Nicholas Léonard et Aaron Courville. 2013. Estimating or propagating gradients through stochastic neurons for conditional computation. *[page 39]*

Ryan Bennett, Jessica Coon et Robert Henderson. 2016. Introduction to Mayan linguistics. *Language and Linguistics Compass*, 10(10) :455–468. *[page 51]*

Vincent Berment. 2004. *Méthodes pour informatiser les langues et les groupes de langues " peu dotées "*. Theses, Université Joseph-Fourier - Grenoble I. *[page 9]*

H. Russell Bernard. 1992. Preserving language diversity. *Human Organization*, 51(1) :82–89. *[page 1]*

H. Russell Bernard. 1997. *Language preservation and publishing*, pages 139–156. De Gruyter Mouton, Berlin, Boston. *[page 1]*

Mathieu Bernard, Roland Thiolliere, Amanda Saksida, Georgia R. Loukatou, Elin Larsen, Mark Johnson, Laia Fibla, Emmanuel Dupoux, Robert Daland, Xuan Nga Cao et Alejandrina Cristia. 2020. Wordseg: Standardizing unsupervised word form segmentation from text. *Behavior Research Methods*, 52(1) :264–278. *[page 24]*

José M. Bernardo et Adrian F. M. Smith. 1994. *Bayesian Theory*. Wiley, New York. *[page 19]*

Laurent Besacier, Etienne Barnard, Alexey Karpov et Tanja Schultz. 2014. Automatic speech recognition for under-resourced languages: A survey. *Speech Communication*, 56 :85–100. *[page 3]*

Mat Bettinson et Steven Bird. 2017. Developing a suite of mobile applications for collaborative language documentation. In *Proceedings of the 2nd Workshop on the Use of Computational Methods in the Study of Endangered Languages*, pages 156–164, Honolulu. Association for Computational Linguistics. *[page 2]*

Balthazar Bickel, Bernard. Comrie et Martin Haspelmath. 2008. The Leipzig Glossing Rules: Conventions for interlinear morpheme-by-morpheme glosses. Leipzig : Max Planck Institute for Evolutionary Anthropology, Department of Linguistics. `https://www.eva.mpg.de/lingua/resources/glossing-rules.php`. *[pages 32 et 41]*

Frédéric Bimbot, Sabine Deligne et François Yvon. 1995. Unsupervised decomposition of phoneme strings into variable-length sequences, by multigrams. In *International Conference of Phonetic Sciences (ICPHS)*, Stockholm, Sweden. *[page 16]*

Steven Bird. 2010. A scalable method for preserving oral literature from small languages. In *The Role of Digital Libraries in a Time of Global Change*, pages 5–14, Berlin, Heidelberg. Springer Berlin Heidelberg. *[page 2]*

Steven Bird. 2020. Decolonising speech and language technology. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3504–3519, Barcelona, Spain (Online). International Committee on Computational Linguistics. *[pages 5, 55, 60 et 74]*

Steven Bird. 2021. Sparse Transcription. *Computational Linguistics*, 46(4) :713–744. *[page 133]*

Steven Bird et David Chiang. 2012. Machine translation for language preservation. In *Proceedings of COLING 2012 : Posters*, pages 125–134, Mumbai, India. The COLING 2012 Organizing Committee. *[page 3]*

Steven Bird, Florian R. Hanke, Oliver Adams et Haejoong Lee. 2014. Aikuma: A mobile app for collaborative language documentation. In *Proceedings of the 2014 Workshop on the Use of Computational Methods in the Study of Endangered Languages*, pages 1–5, Baltimore, Maryland, USA. Association for Computational Linguistics. *[pages 2 et 11]*

Steven Bird, Ewan Klein et Edward Loper. 2009. *Natural Language Processing with Python*. O'Reilly Media, Inc. *[page 109]*

David Blachon, Elodie Gauthier, Laurent Besacier, Guy-Noël Kouarata, Martine Adda-Decker et Annie Rialland. 2016. Parallel speech collection for under-resourced language studies using the lig-aikuma mobile device app. *Procedia Computer Science*, 81 :61–66. SLTU-2016 5th Workshop on Spoken Language Technologies for Under-resourced languages 09-12 May 2016 Yogyakarta, Indonesia. *[page 11]*

Brenda Boerger. 2022. A Grammar Sketch of Natqgu [ntu]: An Oceanic language of Santa Cruz, Solomon Islands. *[page 51]*

Paul Boersma et David Weenink. 1992–2022. Praat: Doing Phonetics by Computer. [Computer program]. Retrieved 23 January 2022. *[page 2]*

Marcely Zanon Boito, Alexandre Bérard, Aline Villavicencio et Laurent Besacier. 2017. Unwritten languages demand attention too! word discovery with encoder-decoder models. In *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 458–465. *[page 25]*

Marcely Zanon Boito, Aline Villavicencio et Laurent Besacier. 2019. Empirical Evaluation of Sequence-to-Sequence Models for Word Discovery in Low-Resource Settings. In *Proc. Interspeech 2019*, pages 2688–2692. *[page 27]*

Marcely Zanon Boito, Bolaji Yusuf, Lucas Ondel, Aline Villavicencio et Laurent Besacier. 2022. Unsupervised word segmentation from discrete speech units in low-resource settings. In *Proceedings of the 1st Annual Meeting of the ELRA/ISCA Special Interest Group on Under-Resourced Languages*, pages 1–9, Marseille, France. European Language Resources Association. *[pages 27 et 49]*

Piotr Bojanowski, Edouard Grave, Armand Joulin et Tomas Mikolov. 2017. Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics*, 5 :135–146. *[page 108]*

Gilles Boulianne. 2022. Phoneme transcription of endangered languages: an evaluation of recent ASR architectures in the single speaker scenario. In *Findings of the Association for Computational Linguistics : ACL 2022*, pages 2301–2308, Dublin, Ireland. Association for Computational Linguistics. *[page 27]*

Luc Bouquiaux et Jacqueline M. C. Thomas, editors. 1976. *Enquête et description des langues à tradition orale*. SELAF, Paris, France. *[page 49]*

Caren Brinckmann. 2009. Transcription bottleneck of speech corpus exploitation. In *LULCL II 2008 - Proceedings of the Second Colloquium on Lesser Used Languages and Computer Linguistics*, pages 165–179, Bozen-Bolzano. EURAC. *[page 2]*

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra et Robert L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2) :263–311. *[pages 107 et 108]*

Gina Bustamante, Arturo Oncevay et Roberto Zariquiey. 2020. No data to crawl? monolingual corpus creation from PDF files of truly low-resource languages in Peru. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 2914–2923, Marseille, France. European Language Resources Association. *[page 47]*

Bernard Caron. 2015. Mettouchi, Amina, Martine Vanhove & Dominique Caubet (eds) (2012). 'The CorpAfroAs Corpus'. ANR CorpAfroAs: a Corpus for Afro-Asiatic languages. Document électronique. Esquisse grammaticale du zaar (langue tchadique du Nigéria). *[pages 50 et 103]*

Ann Clifton et Anoop Sarkar. 2011. Combining morpheme-based machine translation with post-processing morpheme prediction. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics : Human Language Technologies*, pages 32–42, Portland, Oregon, USA. Association for Computational Linguistics. *[page 28]*

CNRS-LLACAN. 2023. ELAN-CorpA (Version 6.0) [computer software]. Villejuif : CNRS-LLACAN (Langage, langues et cultures d'Afrique). *[pages 33 et 102]*

Edith Coates. 2023. An ensembled encoder-decoder system for interlinear glossed text. In *Proceedings of the 20th SIGMORPHON workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 217–221, Toronto, Canada. Association for Computational Linguistics. *[pages 39 et 40]*

Michael Collins et Brian Roark. 2004. Incremental parsing with the perceptron algorithm. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pages 111–118, Barcelona, Spain. *[page 37]*

Bernard Comrie et Maria Polinsky. à paraître. Tsez. In Yuri Koryakov, Yury Lander and Timur Maisak (eds.) The Caucasian Languages. An International Handbook. Mouton. HSK series. *[page 50]*

Alexis Conneau, Alexei Baevski, Ronan Collobert, Abdelrahman Mohamed et Michael Auli. 2021. Unsupervised Cross-Lingual Representation Learning for Speech Recognition. In *Proc. Interspeech 2021*, pages 2426–2430. *[page 27]*

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer et Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics. *[pages 39, 108 et 123]*

Ryan Cotterell, Tim Vieira et Hinrich Schütze. 2016. A joint model of orthography and morphological segmentation. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, pages 664–669, San Diego, California. Association for Computational Linguistics. *[pages 28 et 77]*

Mathias Creutz et Krista Lagus. 2002. Unsupervised discovery of morphemes. In *Proceedings of the ACL-02 Workshop on Morphological and Phonological Learning*, pages 21–30. Association for Computational Linguistics. *[pages 17, 28, 66 et 81]*

Mathias Creutz et Krista Lagus. 2007. Unsupervised models for morpheme segmentation and morphology learning. 4(1). *[pages 17, 28, 30, 66 et 81]*

Ziggy Cross, Michelle Yun, Ananya Apparaju, Jata MacCabe, Garrett Nicolai et Miikka Silfverberg. 2023. Glossy bytes: Neural glossing using subword encoding. In *Proceedings of the 20th SIGMORPHON workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 222–229, Toronto, Canada. Association for Computational Linguistics. *[pages 39 et 40]*

David Crystal. 2000. *Language Death*. Cambridge University Press. *[page 1]*

Sabine Deligne et Frédéric Bimbot. 1995. Language modeling by variable length sequences: theoretical formulation and evaluation of multigrams. In *1995 International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 169–172 vol.1. *[page 16]*

Jacob Devlin, Ming-Wei Chang, Kenton Lee et Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics. *[page 108]*

Charles Donet. 2014. The importance of verb salience in the followability of lezgi oral narratives. Master's thesis, Graduate Institute of Applied Linguistics, Dallas, TX. *[page 51]*

Bonnie J. Dorr. 1994. Machine translation divergences: A formal description and proposed solution. *Computational Linguistics*, 20(4) :597–633. *[pages 34 et 114]*

James Owen Dorsey. 1890. *The Čegiha Language*, volume 6 of *Contributions to North American Ethnology*. U.S. Government, Washington, D.C. *[page 32]*

Zi-Yi Dou et Graham Neubig. 2021. Word alignment by fine-tuning embeddings on parallel corpora. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics : Main Volume*, pages 2112–2128, Online. Association for Computational Linguistics. *[page 108]*

C. Downey, Shannon Drizin, Levon Haroutunian et Shivin Thukral. 2022a. Multilingual unsupervised sequence segmentation transfers to extremely low-resource languages. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, pages 5331–5346, Dublin, Ireland. Association for Computational Linguistics. *[pages 25 et 26]*

C. M. Downey, Fei Xia, Gina-Anne Levow et Shane Steinert-Threlkeld. 2021. A masked segmental language model for unsupervised natural language segmentation. *CoRR*, abs/2104.07829. *[page 25]*

C.m. Downey, Fei Xia, Gina-Anne Levow et Shane Steinert-Threlkeld. 2022b. A masked segmental language model for unsupervised natural language segmentation. In *Proceedings of the 19th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 39–50, Seattle, Washington. Association for Computational Linguistics. *[page 47]*

Matthew S. Dryer et Martin Haspelmath, editors. 2013. *WALS Online (v2020.3)*. Zenodo. *[page 34]*

Ewan Dunbar, Xuan Nga Cao, Juan Benjumea, Julien Karadayi, Mathieu Bernard, Laurent Besacier, Xavier Anguera et Emmanuel Dupoux. 2017. The zero resource speech challenge 2017. In *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 323–330. *[pages 5 et 26]*

Long Duong, Antonios Anastasopoulos, David Chiang, Steven Bird et Trevor Cohn. 2016. An attentional model for speech translation without transcription. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, pages 949–959, San Diego, California. Association for Computational Linguistics. *[page 4]*

David M. Eberhard, Gary F. Simons et Charles D. Fennig. 2023. *Ethnologue: Languages of the World*, 26th edition. SIL International, Dallas, Texas. *[pages 1, 31 et 48]*

Abteen Ebrahimi, Manuel Mager, Shruti Rijhwani, Enora Rice, Arturo Oncevay, Claudia Baltazar, María Cortés, Cynthia Montaño, John E. Ortega, Rolando Coto-solano, Hilaria Cruz, Alexis Palmer et Katharina Kann. 2023. Findings of the AmericasNLP 2023 shared task on machine translation into indigenous languages. In *Proceedings of the Workshop on Natural Language Processing for Indigenous Languages of the Americas (AmericasNLP)*, pages 206–219, Toronto, Canada. Association for Computational Linguistics. *[page 47]*

Georges Martial Embanga Aborobongui. 2013. *Processus segmentaux et tonals en Mbondzi - (variété de la langue embosi C25) -*. Theses, Université de la Sorbonne nouvelle - Paris III. *[pages 48 et 49]*

Ramy Eskander, Francesca Callejas, Elizabeth Nichols, Judith Klavans et Smaranda Muresan. 2020. MorphAGram, evaluation and framework for unsupervised morphological segmentation. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 7112–7122, Marseille, France. European Language Resources Association. *[pages 30 et 81]*

Ramy Eskander, Judith Klavans et Smaranda Muresan. 2019. Unsupervised morphological segmentation for low-resource polysynthetic languages. In *Proceedings of the 16th Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 189–195, Florence, Italy. Association for Computational Linguistics. *[page 30]*

Ramy Eskander, Cass Lowry, Sujay Khandagale, Francesca Callejas, Judith Klavans, Maria Polinsky et Smaranda Muresan. 2021. Minimally-supervised morphological segmentation using Adaptor Grammars with linguistic priors. In *Findings of the Association for Computational Linguistics : ACL-IJCNLP 2021*, pages 3969–3974, Online. Association for Computational Linguistics. *[pages 30, 46 et 131]*

Ramy Eskander, Owen Rambow et Tianchun Yang. 2016. Extending the use of Adaptor Grammars for unsupervised morphological segmentation of unseen languages. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics : Technical Papers*, pages 900–910, Osaka, Japan. The COLING 2016 Organizing Committee. *[page 29]*

Nicholas Evans. 2008. Book review: Essentials of language documentation. *Language Documentation & Conservation*, 2(2) :340–350. *[page 14]*

Daniel S. Fabricant et Norman R. Farnsworth. 2001. The value of plants used in traditional medicine for drug discovery. *Environmental Health Perspectives*, 109 :69–75. *[page 2]*

Ben Foley, Josh Arnold, Rolando Coto-Solano, Gautier Durantin, T. Mark Ellison, Daan van Esch, Scott Heath, František Kratochvíl, Zara Maxwell-Smith, David Nash, Ola Olsson, Mark Richards, Nay San, Hywel Stoakes, Nick Thieberger et Janet Wiles. 2018. Building Speech Recognition Systems for Language Documentation: The CoEDL Endangered Language Pipeline and Inference System (ELPIS). In *Proc. 6th Workshop on Spoken Language Technologies for Under-Resourced Languages (SLTU 2018)*, pages 205–209. *[page 4]*

Clarissa Forbes, Henry Davis, Michael Schwan et the UBC Gitksan Research Laboratory. 2017. Three Gitksan texts. In *Papers for the 52nd International Conference on Salish and Neighbouring Languages*, pages 47–89. UBC Working Papers in Linguistics. *[page 51]*

Clarissa Forbes, Garrett Nicolai et Miikka Silfverberg. 2021. An FST morphological analyzer for the gitksan language. In *Proceedings of the 18th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 188–197, Online. Association for Computational Linguistics. *[page 51]*

Joerg Franke, Markus Mueller, Fatima Hamlaoui, Sebastian Stueker et Alex Waibel. 2016. Phoneme boundary detection using deep bidirectional lstms. In *Speech Communication ; 12. ITG Symposium*, pages 1–5, Paderborn, Germany. *[pages 15 et 26]*

Zvi Galil. 1986. Efficient algorithms for finding maximum matching in graphs. *ACM Comput. Surv.*, 18(1) :23–38. *[page 107]*

Benjamin Galliot, Guillaume Wisniewski, Séverine Guillaume, Laurent Besacier, Guillaume Jacques, Alexis Michaud, Solange Rossato, Minh-Châu Nguyên et Maxime Fily. 2021. Deux corpus audio transcrits de langues rares (japhug et na) normalisés en vue d'expériences en traitement du signal. In *Journées scientifiques du Groupement de recherche "Linguistique informatique, formelle et de terrain" (GDR LIFT)*, Journées GDR LIFT 2021, Grenoble, France. *[pages 44 et 50]*

Benjamin Galliot, Guillaume Wisniewski, Séverine Guillaume, Guillaume Jacques et Alexis Michaud. 2022. Faciliter l'accès des praticiens du Traitement Automatique des Langues à des jeux de données de langues rares : un deuxième point d'étape. In *Journées Jointes des Groupements de Recherche Linguistique Informatique, Formelle et de Terrain (LIFT) et Traitement Automatique des Langues (TAL)*, Marseille, France. CNRS. *[page 50]*

Andrew Gelman, John Carlin, Hal Stern et Donald Rubin. 2004. *Bayesian Data Analysis*. Chapman & Hall/CRC, New York. *[page 19]*

Stuart Geman et Donald Geman. 1984. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6(6) :721–741. *[page 19]*

Ryan Georgi, Fei Xia et William Lewis. 2012. Improving dependency parsing with interlinear glossed text and syntactic projection. In *Proceedings of COLING 2012 : Posters*, pages 371–380, Mumbai, India. The COLING 2012 Organizing Committee. *[page 34]*

Ryan Georgi, Fei Xia et William Lewis. 2015. Enriching interlinear text using automatically constructed annotators. In *Proceedings of the 9th SIGHUM Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities (LaTeCH)*, pages 58–67, Beijing, China. Association for Computational Linguistics. *[page 34]*

Ryan Alden Georgi. 2016. *From Aari to Zulu: Massively Multilingual Creation of Language Tools using Interlinear Glossed Text*. Ph.D. thesis, University of Washington. *[pages 33, 105, 111, 114 et 127]*

Kim Gerdes, Bruno Guillaume, Sylvain Kahane et Guy Perrier. 2018. SUD or surface-syntactic Universal Dependencies: An annotation scheme near-isomorphic to UD. In *Proceedings of the Second Workshop on Universal Dependencies (UDW 2018)*, pages 66–74, Brussels, Belgium. Association for Computational Linguistics. *[page 12]*

Kim Gerdes, Bruno Guillaume, Sylvain Kahane et Guy Perrier. 2019. Improving surface-syntactic Universal Dependencies (SUD): MWEs and deep syntactic features. In *Proceedings of the 18th International Workshop on Treebanks and Linguistic Theories (TLT, SyntaxFest 2019)*, pages 126–132, Paris, France. Association for Computational Linguistics. *[page 12]*

Michael Ginn. 2023. Sigmorphon 2023 shared task of interlinear glossing: Baseline model. *[pages 38, 39, 40 et 120]*

Michael Ginn, Sarah Moeller, Alexis Palmer, Anna Stacey, Garrett Nicolai, Mans Hulden et Miikka Silfverberg. 2023. Findings of the SIGMORPHON 2023 shared task on interlinear glossing. In *Proceedings of the 20th SIGMORPHON workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 186–201, Toronto, Canada. Association for Computational Linguistics. *[pages 36, 38, 51, 52, 121 et 134]*

Leander Girrbach. 2023. Tü-CL at SIGMORPHON 2023: Straight-through gradient estimation for hard attention. In *Proceedings of the 20th SIGMORPHON workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 171–185, Toronto, Canada. Association for Computational Linguistics. *[pages 39 et 40]*

Pierre Godard. 2019. *Unsupervised word discovery for computational language documentation*. Thèses, Université Paris-Saclay. *[pages 4, 11, 17, 21, 22, 55, 56, 58 et 59]*

Pierre Godard, Gilles Adda, Martine Adda-Decker, Alexandre Allauzen, Laurent Besacier, Hélène Bonneau-Maynard, Guy-Noël Kouarata, Kevin Löser, Annie Rialland et François Yvon. 2016. Preliminary Experiments on Unsupervised Word Discovery in Mboshi. In *Proc. Interspeech 2016*, pages 3539–3543. *[pages 4, 16, 21, 22 et 49]*

Pierre Godard, Gilles Adda, Martine Adda-Decker, Juan Benjumea, Laurent Besacier, Jamison Cooper-Leavitt, Guy-Noel Kouarata, Lori Lamel, Hélène Maynard, Markus Mueller, Annie Rialland, Sebastian Stueker, François Yvon et Marcely Zanon-Boito. 2018a. A very low resource language speech corpus for computational language documentation experiments. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA). *[pages 16, 26, 43 et 49]*

Pierre Godard, Laurent Besacier et François Yvon. 2019. Controlling utterance length in NMT-based word segmentation with attention. In *Proceedings of the 16th International Conference on Spoken Language Translation*, Hong Kong. Association for Computational Linguistics. *[pages 25, 49 et 61]*

Pierre Godard, Laurent Besacier, François Yvon, Martine Adda-Decker, Gilles Adda, Hélène Maynard et Annie Rialland. 2018b. Adaptor Grammars for the linguist: Word segmentation experiments for very low-resource languages. In *Proceedings of the Fifteenth Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 32–42, Brussels, Belgium. Association for Computational Linguistics. *[pages 16, 24, 25, 46, 49, 61, 80 et 133]*

Pierre Godard, Marcely Zanon Boito, Lucas Ondel, Alexandre Berard, François Yvon, Aline Villavicencio et Laurent Besacier. 2018c. Unsupervised Word Segmentation from Speech with Attention. In *Proc. Interspeech 2018*, pages 2678–2682. *[pages 4, 26, 27, 49 et 132]*

Pierre Godard, Kevin Loser, Alexandre Allauzen, Laurent Besacier et Francois Yvon. 2018d. Unsupervised Learning of Word Segmentation: Does Tone Matter? In *Proceedings of the 19th*

International Conference on Computational Linguistics and Intelligent Text Processing (CICLING), Hanoi, Vietnam. *[pages 4 et 49]*

Dirk Goldhahn, Thomas Eckart et Uwe Quasthoff. 2012. Building large monolingual dictionaries at the Leipzig corpora collection: From 100 to 200 languages. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 759–765, Istanbul, Turkey. European Language Resources Association (ELRA). *[page 91]*

Omer Goldman, Khuyagbaatar Batsuren, Salam Khalifa, Aryaman Arora, Garrett Nicolai, Reut Tsarfaty et Ekaterina Vylomova. 2023. SIGMORPHON–UniMorph 2023 shared task 0: Typologically diverse morphological inflection. In *Proceedings of the 20th SIGMORPHON workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 117–125, Toronto, Canada. Association for Computational Linguistics. *[page 39]*

Sharon Goldwater. 2006. *Nonparametric Bayesian Models of Lexical Acquisition*. Ph.D. thesis, Brown University. *[pages 17, 20, 21, 22, 57 et 58]*

Sharon Goldwater, Thomas L. Griffiths et Mark Johnson. 2005. Interpolating between types and tokens by estimating power-law generators. In *Proceedings of the 18th International Conference on Neural Information Processing Systems*, NIPS'05, page 459–466, Cambridge, MA, USA. MIT Press. *[pages 29 et 91]*

Sharon Goldwater, Thomas L. Griffiths et Mark Johnson. 2006. Contextual dependencies in unsupervised word segmentation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 673–680, Sydney, Australia. Association for Computational Linguistics. *[pages 19 et 24]*

Sharon Goldwater, Thomas L. Griffiths et Mark Johnson. 2009. A Bayesian framework for word segmentation: Exploring the effects of context. *Cognition*, 112(1) :21–54. *[pages 17, 18, 19, 20, 21, 42, 55, 56, 58, 59, 64, 71, 91 et 159]*

Sharon Goldwater, Thomas L. Griffiths et Mark Johnson. 2011. Producing power-law distributions and damping word frequencies with two-stage language models. *Journal of Machine Learning Research*, 12 :2335–2382. *[pages 19 et 21]*

Alex Graves, Santiago Fernández, Faustino Gomez et Jürgen Schmidhuber. 2006. Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, page 369–376, New York, NY, USA. Association for Computing Machinery. *[page 39]*

Joseph Greenberg. 1963. Some universals of grammar with particular reference to the order of meaningful elements. In Joseph Greenberg, editor, *Universals of Language*, pages 73–113. MIT Press, London. *[page 34]*

Stig-Arne Grönroos, Sami Virpioja, Peter Smit et Mikko Kurimo. 2014. Morfessor FlatCat: An HMM-based method for unsupervised and semi-supervised learning of morphology. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics : Technical Papers*, pages 1177–1185, Dublin, Ireland. Dublin City University and Association for Computational Linguistics. *[page 31]*

Séverine Guillaume, Guillaume Wisniewski, Cécile Macaire, Guillaume Jacques, Alexis Michaud, Benjamin Galliot, Maximin Coavoux, Solange Rossato, Minh-Châu Nguyên et Maxime Fily. 2022. Fine-tuning pre-trained models for automatic speech recognition, experiments on a fieldwork corpus of japhug (trans-himalayan family). In *Proceedings of the Fifth Workshop on the Use of Computational Methods in the Study of Endangered Languages*, pages 170–178, Dublin, Ireland. Association for Computational Linguistics. *[pages 27, 28, 50 et 133]*

Vishwa Gupta et Gilles Boulianne. 2020a. Automatic transcription challenges for Inuktitut, a low-resource polysynthetic language. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 2521–2527, Marseille, France. European Language Resources Association. *[page 27]*

Vishwa Gupta et Gilles Boulianne. 2020b. Speech transcription challenges for resource constrained indigenous language Cree. In *Proceedings of the 1st Joint Workshop on Spoken Language Technologies for Under-resourced languages (SLTU) and Collaboration and Computing for Under-Resourced Languages (CCURL)*, pages 362–367, Marseille, France. European Language Resources association. *[page 27]*

Vishwa Gupta et Gilles Boulianne. 2022. Progress in multilingual speech recognition for low resource languages Kurmanji Kurdish, Cree and inuktut. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 6420–6428, Marseille, France. European Language Resources Association. *[page 27]*

Malcolm Guthrie. 1948. *The Classification of the Bantu Languages*. Oxford University Press for the International African Institute, London. *[page 43]*

Ken Hale, Michael Krauss, Lucille J. Watahomigie, Akira Y. Yamamoto, Colette Craig, LaVerne Masayesva Jeanne et Nora C. England. 1992. Endangered languages. *Language*, 68(1) :1–42. *[pages 1 et 2]*

Fatima Hamlaoui, Emmanuel-Moselly Makasso, Markus Müller, Jonas Engelmann, Gilles Adda, Alex Waibel et Sebastian Stüker. 2018. BULBasaa: A bilingual basaa-French speech corpus for the evaluation of language documentation tools. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA). *[page 43]*

Florian R. Hanke et Steven Bird. 2013. Large-scale text collection for unwritten languages. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 1134–1138, Nagoya, Japan. Asian Federation of Natural Language Processing. *[page 11]*

Zellig S. Harris. 1955. From phoneme to morpheme. *Language*, 31(2) :190–222. *[page 16]*

Martin Haspelmath. 1993. *A Grammar of Lezgian*. De Gruyter Mouton, Berlin, Boston. *[page 51]*

Taiqi He, Lindia Tjuatja, Nathaniel Robinson, Shinji Watanabe, David R. Mortensen, Graham Neubig et Lori Levin. 2023. SigMoreFun submission to the SIGMORPHON shared task on interlinear glossing. In *Proceedings of the 20th SIGMORPHON workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 209–216, Toronto, Canada. Association for Computational Linguistics. *[pages 39 et 40]*

Nikolaus P. Himmelmann. 1998. Documentary and descriptive linguistics. *Linguistics*, 36(1) :161–196. *[page 2]*

Nikolaus P. Himmelmann. 2006. *Language documentation: What is it and what is it good for?*, pages 1–30. De Gruyter Mouton, Berlin, New York. *[page 3]*

Leanne Hinton. 2011. *Revitalization of endangered languages*, Cambridge Handbooks in Language and Linguistics, page 291–311. Cambridge University Press. *[page 14]*

Rebecca Hwa, Philip Resnik, Amy Weinberg et Okan Kolak. 2002. Evaluating translational correspondence using annotation projection. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 392–399, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics. *[page 34]*

Ayyoob ImaniGooghari, Peiqin Lin, Amir Hossein Kargaran, Silvia Severini, Masoud Jalili Sabet, Nora Kassner, Chunlan Ma, Helmut Schmid, André Martins, François Yvon et Hinrich

Schütze. 2023. Glot500: Scaling multilingual corpora and language models to 500 languages. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, pages 1082–1117, Toronto, Canada. Association for Computational Linguistics. *[page 133]*

Guillaume Jacques. 2015. Dictionnaire japhug-chinois-français, version 1.0. Paris : Projet HimalCo. http://himalco.huma-num.fr/dictionaries/index.htm. *[page 50]*

Guillaume Jacques. 2021. *A grammar of Japhug*. Number 1 in Comprehensive Grammar Library. Language Science Press, Berlin. *[pages ix, 10, 33, 49 et 50]*

Masoud Jalili Sabet, Philipp Dufter, François Yvon et Hinrich Schütze. 2020. SimAlign: High quality word alignments without parallel training data using static and contextualized embeddings. In *Findings of the Association for Computational Linguistics : EMNLP 2020*, pages 1627–1643, Online. Association for Computational Linguistics. *[pages 107, 108 et 109]*

Jesin James, Vithya Yogarajan, Isabella Shields, Catherine Watson, Peter Keegan, Keoni Mahelona et Peter-Lucas Jones. 2022. Language models for code-switch detection of te reo Māori and English in a low-resource setting. In *Findings of the Association for Computational Linguistics : NAACL 2022*, pages 650–660, Seattle, United States. Association for Computational Linguistics. *[page 15]*

Robbie Jimerson et Emily Prud'hommeaux. 2018. ASR for documenting acutely under-resourced indigenous languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA). *[page 3]*

Zhihui Jin et Kumiko Tanaka-Ishii. 2006. Unsupervised segmentation of Chinese text by use of branching entropy. In *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, pages 428–435, Sydney, Australia. Association for Computational Linguistics. *[page 25]*

Mark Johnson. 2008a. Unsupervised word segmentation for Sesotho using Adaptor Grammars. In *Proceedings of the Tenth Meeting of ACL Special Interest Group on Computational Morphology and Phonology*, pages 20–27, Columbus, Ohio. Association for Computational Linguistics. *[pages 24, 77 et 81]*

Mark Johnson. 2008b. Using Adaptor Grammars to identify synergies in the unsupervised acquisition of linguistic structure. In *Proceedings of ACL-08 : HLT*, pages 398–406, Columbus, Ohio. Association for Computational Linguistics. *[page 24]*

Mark Johnson, Anne Christophe, Emmanuel Dupoux et Katherine Demuth. 2014. Modelling function words improves unsupervised word segmentation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, pages 282–292, Baltimore, Maryland. Association for Computational Linguistics. *[page 24]*

Mark Johnson et Sharon Goldwater. 2009. Improving nonparameteric Bayesian inference: experiments on unsupervised word segmentation with adaptor grammars. In *Proceedings of Human Language Technologies : The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 317–325, Boulder, Colorado. Association for Computational Linguistics. *[pages 22 et 24]*

Mark Johnson, Thomas L. Griffiths et Sharon Goldwater. 2007. Adaptor Grammars: a Framework for Specifying Compositional Nonparametric Bayesian Models. In *Advances in Neural Information Processing Systems 19*, pages 641–648, Cambridge, MA. MIT Press. *[pages 22, 23, 29, 56, 80 et 131]*

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali et Monojit Choudhury. 2020. The state and fate of linguistic diversity and inclusion in the NLP world. In *Proceedings of the 58th*

Annual Meeting of the Association for Computational Linguistics, pages 6282–6293, Online. Association for Computational Linguistics. *[page 9]*

Daniel Jurafsky et James H. Martin. 2009. *Speech and Language Processing : An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, 2nd edition. Prentice-Hall. *[page 65]*

Sylvain Kahane, Santiago Herrera, Bruno Guillaume et Kim Gerdes. 2023. Autogramm : développement simultané de treebanks et de grammaires à partir de corpus. *[page 12]*

Katharina Kann, Jesus Manuel Mager Hois, Ivan Vladimir Meza-Ruiz et Hinrich Schütze. 2018. Fortification of neural morphological segmentation models for polysynthetic minimal-resource languages. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 1 (Long Papers)*, pages 47–57, New Orleans, Louisiana. Association for Computational Linguistics. *[pages 30 et 31]*

Kazuya Kawakami, Chris Dyer et Phil Blunsom. 2019. Learning to discover, ground and use words with segmental neural language models. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6429–6441, Florence, Italy. Association for Computational Linguistics. *[page 25]*

Sujay Khandagale, Yoann Léveillé, Samuel Miller, Derek Pham, Ramy Eskander, Cass Lowry, Richard Compton, Judith Klavans, Maria Polinsky et Smaranda Muresan. 2022. Towards unsupervised morphological analysis of polysynthetic languages. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 2 : Short Papers)*, pages 334–340, Online only. Association for Computational Linguistics. *[page 30]*

Yoshiaki Kitagawa et Mamoru Komachi. 2018. Long short-term memory for Japanese word segmentation. In *Proceedings of the 32nd Pacific Asia Conference on Language, Information and Computation*, Hong Kong. Association for Computational Linguistics. *[page 25]*

Jordan Kodner, Salam Khalifa, Khuyagbaatar Batsuren, Hossep Dolatian, Ryan Cotterell, Faruk Akkus, Antonios Anastasopoulos, Taras Andrushko, Aryaman Arora, Nona Atanalov, Gábor Bella, Elena Budianskaya, Yustinus Ghanggo Ate, Omer Goldman, David Guriel, Simon Guriel, Silvia Guriel-Agiashvili, Witold Kieraś, Andrew Krizhanovsky, Natalia Krizhanovsky, Igor Marchenko, Magdalena Markowska, Polina Mashkovtseva, Maria Nepomniashchaya, Daria Rodionova, Karina Scheifer, Alexandra Sorova, Anastasia Yemelina, Jeremiah Young et Ekaterina Vylomova. 2022. SIGMORPHON–UniMorph 2022 shared task 0: Generalization and typologically diverse morphological inflection. In *Proceedings of the 19th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 176–203, Seattle, Washington. Association for Computational Linguistics. *[page 47]*

Oskar Kohonen, Sami Virpioja et Krista Lagus. 2010. Semi-supervised learning of concatenative morphology. In *Proceedings of the 11th Meeting of the ACL Special Interest Group on Computational Morphology and Phonology*, pages 78–86, Uppsala, Sweden. Association for Computational Linguistics. *[page 31]*

Kimmo Koskenniemi. 1983. *Two-level morphology: A general computational model for word-form recognition and production*, volume 11. University of Helsinki, Department of General Linguistics Helsinki, Finland. *[page 77]*

Guy Noël Kouarata. 2014. *Variations de formes dans la langue Mbochi (Bantu C25)*. Ph.D. thesis, Université Lumière Lyon 2. *[page 49]*

Michael Krauss. 1992. The world's languages in crisis. *Language*, 68(1) :4–10. *[page 2]*

Taku Kudo. 2018. Subword regularization: Improving neural network translation models with multiple subword candidates. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, pages 66–75, Melbourne, Australia. Association for Computational Linguistics. *[pages 16 et 66]*

Mikko Kurimo, Sami Virpioja, Ville Turunen et Krista Lagus. 2010. Morpho challenge 2005-2010: Evaluations and results. In *Proceedings of the 11th Meeting of the ACL Special Interest Group on Computational Morphology and Phonology*, pages 87–95, Uppsala, Sweden. Association for Computational Linguistics. *[page 28]*

John D. Lafferty, Andrew McCallum et Fernando C. N. Pereira. 2001. Conditional random fields : Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, page 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc. *[pages 28, 36, 82, 100 et 161]*

William Lane et Steven Bird. 2020. Bootstrapping techniques for polysynthetic morphological analysis. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6652–6661, Online. Association for Computational Linguistics. *[page 13]*

Michel Launey. 1994. *Une grammaire omniprédicative : Essai sur la morphosyntaxe du nahuatl classique*. Editions du CNRS, Paris. *[page 32]*

Thomas Lavergne, Alexandre Allauzen, Josep Maria Crego et François Yvon. 2011. From n-gram-based to CRF-based translation models. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 542–553, Edinburgh, Scotland. Association for Computational Linguistics. *[page 114]*

Thomas Lavergne, Alexandre Allauzen et François Yvon. 2013. Un cadre d'apprentissage intégralement discriminant pour la traduction statistique. In *Actes de la 20ème Conférence sur le Traitement Automatique des Langues Naturelles*, pages 450–463, Les Sables d'Olonne, France. ATALA. *[pages 114, 115 et 116]*

Thomas Lavergne, Olivier Cappé et François Yvon. 2010. Practical very large scale CRFs. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 504–513, Uppsala, Sweden. Association for Computational Linguistics. *[page 82]*

Thomas Lavergne et François Yvon. 2017. Learning the structure of variable-order CRFs: a finite-state perspective. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 433–439, Copenhagen, Denmark. Association for Computational Linguistics. *[page 102]*

Eric Le Ferrand, Steven Bird et Laurent Besacier. 2020. Enabling interactive transcription in an indigenous community. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3422–3428, Barcelona, Spain (Online). International Committee on Computational Linguistics. *[pages 13 et 133]*

Christian Lehmann. 1982. Directions for interlinear morphemic translations. *Folia Linguistica - FOLIA LINGUIST*, 16(1-4) :199–224. *[pages 31 et 32]*

Marika Lekakou, Valeria Baldissera et Antonis Anastasopoulos. 2013. Documentation and analysis of an endangered language: Aspects of the grammar of griko. University of Ioannina. *[page 44]*

M. Paul Lewis et Gary F. Simons. 2010. Assessing endangerment: Expanding fishman's gids. *Revue Roumaine de Linguistique*, 55 :103–20. *[page 1]*

William D. Lewis. 2006. Odin: A model for adapting and enriching legacy infrastructure. In *2006 Second IEEE International Conference on e-Science and Grid Computing (e-Science'06)*, pages 137–137. *[pages 12, 34, 41 et 44]*

William D. Lewis et Fei Xia. 2008. Automatically identifying computationally relevant typological features. In *Proceedings of the Third International Joint Conference on Natural Language Processing : Volume-II*. *[pages 34 et 98]*

William D. Lewis et Fei Xia. 2010. Developing ODIN: A Multilingual Repository of Annotated Language Data for Hundreds of the World's Languages. *Literary and Linguistic Computing*, 25(3) :303–319. *[pages 12, 34, 41 et 44]*

Percy Liang, Alexandre Bouchard-Côté, Dan Klein et Ben Taskar. 2006. An end-to-end discriminative approach to machine translation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 761–768, Sydney, Australia. Association for Computational Linguistics. *[page 116]*

Patrick Littell, Anna Kazantseva, Roland Kuhn, Aidan Pine, Antti Arppe, Christopher Cox et Marie-Odile Junker. 2018. Indigenous language technologies in Canada: Assessment, challenges, and successes. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2620–2632, Santa Fe, New Mexico, USA. Association for Computational Linguistics. *[page 14]*

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer et Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *[page 38]*

Zoey Liu, Robert Jimerson et Emily Prud'hommeaux. 2021. Morphological segmentation for Seneca. In *Proceedings of the First Workshop on Natural Language Processing for Indigenous Languages of the Americas*, pages 90–101, Online. Association for Computational Linguistics. *[page 31]*

Kevin Löser et Alexandre Allauzen. 2016. Une méthode non-supervisée pour la segmentation morphologique et l'apprentissage de morphotactique à l'aide de processus de Pitman-Yor (an unsupervised method for joint morphological segmentation and morphotactics learning using Pitman-Yor processes). In *Actes de la conférence conjointe JEP-TALN-RECITAL 2016. volume 2 : TALN (Articles longs)*, pages 207–220, Paris, France. AFCP - ATALA. *[pages 22, 49, 56 et 133]*

Cécile Macaire. 2021. Recognizing lexical units in low-resource language contexts with supervised and unsupervised neural networks. Research report, LACITO (UMR 7107). *[page 50]*

Manuel Mager, Ximena Gutierrez-Vasques, Gerardo Sierra et Ivan Meza-Ruiz. 2018. Challenges of language technologies for the indigenous languages of the Americas. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 55–69, Santa Fe, New Mexico, USA. Association for Computational Linguistics. *[page 43]*

Manuel Mager, Arturo Oncevay, Abteen Ebrahimi, John Ortega, Annette Rios, Angela Fan, Ximena Gutierrez-Vasques, Luis Chiruzzo, Gustavo Giménez-Lugo, Ricardo Ramos, Ivan Vladimir Meza Ruiz, Rolando Coto-Solano, Alexis Palmer, Elisabeth Mager-Hois, Vishrav Chaudhary, Graham Neubig, Ngoc Thang Vu et Katharina Kann. 2021. Findings of the AmericasNLP 2021 shared task on open machine translation for indigenous languages of the Americas. In *Proceedings of the First Workshop on Natural Language Processing for Indigenous Languages of the Americas*, pages 202–217, Online. Association for Computational Linguistics. *[page 26]*

Pierre Magistry. 2012. Segmentation non supervisée : le cas du mandarin (unsupervized word segmentation) [in French]. In *Proceedings of the Joint Conference JEP-TALN-RECITAL 2012, volume 3 : RECITAL*, pages 1–13, Grenoble, France. ATALA/AFCP. *[page 25]*

Pierre Magistry et Benoît Sagot. 2012. Unsupervized word segmentation: the case for Mandarin Chinese. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2 : Short Papers)*, pages 383–387, Jeju Island, Korea. Association for Computational Linguistics. *[page 25]*

Angelina McMillan-Major. 2020. Automating gloss generation in interlinear glossed text. In *Proceedings of the Society for Computation in Linguistics 2020*, pages 355–366, New York, New York. Association for Computational Linguistics. *[pages 36, 38, 40, 104, 105, 118 et 126]*

Devansh Mehta, Sebastin Santy, Ramaravind Kommiya Mothilal, Brij Mohan Lal Srivastava, Alok Sharma, Anurag Shukla, Vishnu Prasad, Venkanna U, Amit Sharma et Kalika Bali. 2020. Learnings from technological interventions in a low resource language: A case-study on Gondi. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 2832–2838, Marseille, France. European Language Resources Association. *[page 15]*

Maite Melero, Sakriani Sakti et Claudia Soria, editors. 2022. *Proceedings of the 1st Annual Meeting of the ELRA/ISCA Special Interest Group on Under-Resourced Languages*. European Language Resources Association, Marseille, France. *[page 13]*

Francois Meyer et Jan Buys. 2022. Subword segmental language modelling for nguni languages. In *Findings of the Association for Computational Linguistics : EMNLP 2022*, pages 6636–6649, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics. *[page 29]*

Susanne Maria Michaelis, Philippe Maurer, Martin Haspelmath et Magnus Huber, editors. 2013. *APiCS Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig. *[page 46]*

Alexis Michaud, Oliver Adams, Trevor Anthony Cohn, Graham Neubig et Séverine Guillaume. 2018. Integrating automatic transcription into the language documentation workflow : Experiments with na data and the persephone toolkit. *Language Documentation & Conservation*, 12 :393–429. *[page 53]*

Alexis Michaud, Séverine Guillaume, Guillaume Jacques, Đăng-Khoa Mạc, Michel Jacobson, Thu-Hà Phạm et Matthew Deo. 2016. Contribuer au progrès solidaire des recherches et de la documentation : la Collection Pangloss et la Collection AuCo. In *Journées d'Etude de la Parole 2016*, volume 1 of *Actes de la conférence conjointe JEP-TALN-RECITAL 2016, volume 1 : Journées d'Etude de la Parole*, pages 155–163, Paris, France. Association Francophone de la Communication Parlée. *[page 46]*

Daichi Mochihashi, Takeshi Yamada et Naonori Ueda. 2009. Bayesian unsupervised word segmentation with nested Pitman-Yor language modeling. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 100–108, Suntec, Singapore. Association for Computational Linguistics. *[pages 21, 22, 25, 56, 60, 64, 65 et 80]*

Sarah Moeller et Mans Hulden. 2018. Automatic glossing in a low-resource setting for language documentation. In *Proceedings of the Workshop on Computational Modeling of Polysynthetic Languages*, pages 84–93, Santa Fe, New Mexico, USA. Association for Computational Linguistics. *[pages 37, 38, 40, 51, 82, 102, 104 et 120]*

Sarah Moeller, Martha Palmer, Andrew Cowell, Alexis Palmer et Katharina Kann. 2021. *Integrating Machine Learning into Language Documentation and Description*. Ph.D. thesis, USA. AAI28415172. *[page 4]*

Tumi Moeng, Sheldon Reay, Aaron Daniels et Jan Buys. 2021. Canonical and surface morphological segmentation for nguni languages. In *2nd AfricaNLP Workshop Proceedings, AfricaNLP@EACL 2021, Virtual Event, April 19, 2021*. *[page 29]*

David R. Mortensen, Ela Gulsen, Taiqi He, Nathaniel Robinson, Jonathan Amith, Lindia Tjuatja et Lori Levin. 2023. Generalized glossing guidelines: An explicit, human- and machine-readable, item-and-process convention for morphological annotation. In *Proceedings of the 20th SIGMORPHON workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 58–67, Toronto, Canada. Association for Computational Linguistics. *[page 32]*

Thomas Mueller, Helmut Schmid et Hinrich Schütze. 2013. Efficient higher-order CRFs for morphological tagging. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 322–332, Seattle, Washington, USA. Association for Computational Linguistics. *[page 102]*

Markus Müller, Jörg Franke, Alex Waibel et Sebastian Stüker. 2017a. Towards phoneme inventory discovery for documentation of unwritten languages. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5200–5204, New Orleans, LA, USA. *[pages 16 et 26]*

Markus Müller, Sebastian Stüker et Alex Waibel. 2017b. Dblstm based multilingual articulatory feature extraction for language documentation. In *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 417–423. *[pages 16 et 26]*

Hiroshi Nakagawa, Anna Bugaeva, Miki Kobayashi et Yoshimi Yoshikawa. 2016-2021. A glossed audio corpus of ainu folklore. Available from `https://ainu.ninjal.ac.jp/folklore/`. *[page 44]*

Jason Naradowski et Sharon Goldwater. 2009. Improving morphology induction by learning spelling rules. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI-09)*, pages 1–6. International Joint Conference on Artificial Intelligence ; Conference date : 11-07-2009. *[page 29]*

Ndongo Ibara. 2014. *Embosi-English Dictionary*. Peter Lang Verlag, Berlin, Germany. *[page 48]*

Daniel Nettle et Suzanne Romaine. 2000. *Vanishing Voices : The Extinction of the World's Languages*. Oxford University Press. *[page 1]*

Graham Neubig. 2014. Simple, correct parallelization for blocked gibbs sampling. In *Technical Report*. *[page 22]*

Graham Neubig, Patrick Littell, Chian-Yu Chen, Jean Lee, Zirui Li, Yu-Hsiang Lin et Yuyan Zhang. 2018. Towards a general-purpose linguistic annotation backend. *[page 4]*

Graham Neubig, Shruti Rijhwani, Alexis Palmer, Jordan MacKenzie, Hilaria Cruz, Xinjian Li, Matthew Lee, Aditi Chaudhary, Luke Gessler, Steven Abney, Shirley Anugrah Hayati, Antonios Anastasopoulos, Olga Zamaraeva, Emily Prud'hommeaux, Jennette Child, Sara Child, Rebecca Knowles, Sarah Moeller, Jeffrey Micher, Yiyuan Li, Sydney Zink, Mengzhou Xia, Roshan S Sharma et Patrick Littell. 2020. A summary of the first workshop on language technology for language documentation and revitalization. In *Proceedings of the 1st Joint Workshop on Spoken Language Technologies for Under-resourced languages (SLTU) and Collaboration and Computing for Under-Resourced Languages (CCURL)*, pages 342–351, Marseille, France. European Language Resources association. *[pages 12 et 14]*

Neasa Ní Chiaráin, Oisín Nolan, Madeleine Comtois, Neimhin Robinson Gunning, Harald Berthelsen et Ailbhe Ni Chasaide. 2022. Using speech and NLP resources to build an iCALL platform for a minority language, the story of an scéalaí, the Irish experience to date. In *Proceedings of the Fifth Workshop on the Use of Computational Methods in the Study of Endangered Languages*, pages 109–118, Dublin, Ireland. Association for Computational Linguistics. *[page 15]*

Sebastian Nordhoff. 2018. *Language Science Press business model*. Language Science Press, Berlin. *[page 45]*

Sebastian Nordhoff. 2020. From the attic to the cloud: mobilization of endangered language resources with linked data. In *Proceedings of the Workshop about Language Resources for the SSH Cloud*, pages 10–18, Marseille, France. European Language Resources Association. *[pages 46 et 53]*

Sebastian Nordhoff et Thomas Krämer. 2022. IMTVault: Extracting and enriching low-resource language interlinear glossed text from grammatical descriptions and typological survey articles. In *Proceedings of the 8th Workshop on Linked Data in Linguistics within the 13th Language Resources and Evaluation Conference*, pages 17–25, Marseille, France. European Language Resources Association. *[pages 41, 45, 50, 53, 121, 124 et 126]*

Åshild Næss et Brenda H. Boerger. 2008. Reefs-santa cruz as oceanic: Evidence from the verb complex. *Oceanic Linguistics*, 47(1) :185–212. *[page 51]*

Shu Okabe. 2021. Weakly supervised word segmentation. Technical report. *[page 56]*

Shu Okabe, Laurent Besacier et François Yvon. 2022. Weakly supervised word segmentation for computational language documentation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, pages 7385–7398, Dublin, Ireland. Association for Computational Linguistics. *[page 6]*

Shu Okabe et François Yvon. 2022a. Modèle-s bayés-ien-s pour la segment-ation à deux niveau-x faible-ment super-vis-é-e (Bayesian models for weakly supervised two-level segmentation). In *Actes de la 29e Conférence sur le Traitement Automatique des Langues Naturelles. Volume 1 : conférence principale*, pages 174–182, Avignon, France. ATALA. *[pages 6, 56 et 78]*

Shu Okabe et François Yvon. 2022b. Vers la génération automatique de gloses pour la documentation automatique des langues. In *Journées Jointes des Groupements de Recherche Linguistique Informatique, Formelle et de Terrain (LIFT) et Traitement Automatique des Langues (TAL)*, pages 198–203, Marseille, France. CNRS. *[pages 6 et 100]*

Shu Okabe et François Yvon. 2023a. Joint word and morpheme segmentation with Bayesian non-parametric models. In *Findings of the Association for Computational Linguistics : EACL 2023*, pages 640–654, Dubrovnik, Croatia. Association for Computational Linguistics. *[pages 7 et 78]*

Shu Okabe et François Yvon. 2023b. LISN @ SIGMORPHON 2023 shared task on interlinear glossing. In *Proceedings of the 20th SIGMORPHON workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 202–208, Toronto, Canada. Association for Computational Linguistics. *[pages 7, 100 et 120]*

Shu Okabe et François Yvon. 2023c. Production automatique de gloses interlinéaires à travers un modèle probabiliste exploitant des alignements. In *Actes de CORIA-TALN 2023. Actes de la 30e Conférence sur le Traitement Automatique des Langues Naturelles (TALN), volume 1 : travaux de recherche originaux – articles longs*, pages 262–274, Paris, France. ATALA. *[pages 7 et 100]*

Shu Okabe et François Yvon. 2023d. Towards multilingual interlinear morphological glossing. In *Findings of the Association for Computational Linguistics : EMNLP 2023*, pages 5958–5971, Singapore. Association for Computational Linguistics. *[page 7]*

Shu Okabe, François Yvon et Laurent Besacier. 2021. Segmentation en mots faiblement supervisée pour la documentation automatique des langues. In *Journées du Groupement de Recherche Linguistique Informatique, Formelle et de Terrain (LIFT)*, Grenoble, France. CNRS. *[pages 6 et 56]*

Lucas Ondel, Lukaš Burget et Jan Černocký. 2016. Variational inference for acoustic unit discovery. *Procedia Computer Science*, 81 :80–86. SLTU-2016 5th Workshop on Spoken Language Technologies for Under-resourced languages 09-12 May 2016 Yogyakarta, Indonesia. *[page 26]*

Alexis Palmer, Taesun Moon et Jason Baldridge. 2009. Evaluating automation strategies in language documentation. In *Proceedings of the NAACL HLT 2009 Workshop on Active Learning for Natural Language Processing*, pages 36–44, Boulder, Colorado. Association for Computational Linguistics. *[pages 13, 36, 40, 41, 51, 65, 72 et 132]*

Alexis Palmer, Taesun Moon, Jason Baldridge, Katrin Erk, Eric Campbell et Telma Can. 2010. Computational strategies for reducing annotation effort in language documentation: A case study in creating interlinear texts for uspanteko. *Linguistic Issues in Language Technology*, 3. *[page 51]*

Kishore Papineni, Salim Roukos, Todd Ward et Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics. *[page 120]*

Ludger Paschen, François Delafontaine, Christoph Draxler, Susanne Fuchs, Matthew Stave et Frank Seifart. 2020. Building a time-aligned cross-linguistic reference corpus from language documentation data (DoReCo). In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 2657–2666, Marseille, France. European Language Resources Association. *[pages 12 et 47]*

Ludger Paschen, Susanne Fuchs et Frank Seifart. 2022. Final lengthening and vowel length in 25 languages. *Journal of Phonetics*, 94 :101179. *[page 12]*

Jim Pitman et Marc Yor. 1997. The two-parameter poisson-dirichlet distribution derived from a stable subordinator. *The Annals of Probability*, 25(2) :855–900. *[pages 21 et 60]*

Telma Can Pixabaj, Miguel Angel Vicente Méndez, María Vicente Méndez et Oswaldo Ajcot Damián. 2007. Text Collections in Four Mayan Languages. Archived in The Archive of the Indigenous Languages of Latin America (AILLA). *[page 51]*

Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, Jan Silovsky, Georg Stemmer et Karel Vesely. 2011. The kaldi speech recognition toolkit. In *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*. IEEE Signal Processing Society. IEEE Catalog No. : CFP11SRW-USB. *[page 4]*

Gabriela Pérez Báez, Rachel Vogel et Eve Okura. 2018. Comparative Analysis in Language Revitalization Practices: Addressing the Challenge. In *The Oxford Handbook of Endangered Languages*. Oxford University Press. *[page 14]*

Gabriela Pérez Báez, Rachel Vogel et Uia Patolo. 2019. Global survey of revitalization efforts: A mixed methods approach to understanding language revitalization practices. *Language Documentation & Conservation*, 13 :446–513. *[page 14]*

Annie Rialland, Georges Aborobongui, Martine Adda-Decker et Lori Lamel. 2015. Dropping of the class-prefix consonant, vowel elision and automatic phonological mining in Embosi (Bantu C 25). In *Proceedings of the 44th Annual Conference on African Linguistics*, pages 221–230, Somerville. Cascadilla. *[page 49]*

Annie Rialland, Martine Adda-Decker, Guy-Noël Kouarata, Gilles Adda, Laurent Besacier, Lori Lamel, Elodie Gauthier, Pierre Godard et Jamison Cooper-Leavitt. 2018. Parallel corpora in Mboshi (Bantu C25, Congo-Brazzaville). In *Proceedings of the Eleventh International*

Conference on Language Resources and Evaluation (LREC 2018), Miyazaki, Japan. European Language Resources Association (ELRA). *[page 49]*

Martin Riedmiller et Heinrich Braun. 1993. A direct adaptive method for faster backpropagation learning: the rprop algorithm. In *IEEE International Conference on Neural Networks*, pages 586–591 vol.1. *[page 116]*

Bruce Rigsby. 1986. *Gitxsan Grammar*. University of Queensland, Australia. *[page 51]*

Bruce Rigsby. 1989. *A later view of Gitksan syntax*, pages 245–260. De Gruyter Mouton, Berlin, Boston. *[page 51]*

Shruti Rijhwani, Antonios Anastasopoulos et Graham Neubig. 2020. OCR Post Correction for Endangered Language Texts. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5931–5942, Online. Association for Computational Linguistics. *[page 45]*

Shruti Rijhwani, Daisy Rosenblum, Antonios Anastasopoulos et Graham Neubig. 2021. Lexically aware semi-supervised learning for OCR post-correction. *Transactions of the Association for Computational Linguistics*, 9 :1285–1302. *[page 45]*

Jorma Rissanen. 1989. *Stochastic Complexity in Statistical Inquiry*. Series in Computer Science - Vol 15. World Scientific Publishing. *[pages 16 et 28]*

Chris Rogers. 2010. Review of Fieldworks Language Explorer (FLEx) 3.0. In *Language Documentation & Conservation 4*, pages 78–84. *[pages 2, 33 et 102]*

Teemu Ruokolainen, Oskar Kohonen, Kairit Sirts, Stig-Arne Grönroos, Mikko Kurimo et Sami Virpioja. 2016. A Comparative Study of Minimally Supervised Morphological Segmentation. *Computational Linguistics*, 42(1) :91–120. *[page 28]*

Teemu Ruokolainen, Oskar Kohonen, Sami Virpioja et Mikko Kurimo. 2013. Supervised morphological segmentation in a low-resource learning setting using conditional random fields. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 29–37, Sofia, Bulgaria. Association for Computational Linguistics. *[page 30]*

Teemu Ruokolainen, Oskar Kohonen, Sami Virpioja et Mikko Kurimo. 2014. Painless semi-supervised morphological segmentation using conditional random fields. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, volume 2 : Short Papers*, pages 84–89, Gothenburg, Sweden. Association for Computational Linguistics. *[page 30]*

Jenny R. Saffran, Richard N. Aslin et Elissa L. Newport. 1996. Statistical learning by 8-month-old infants. *Science (New York, N.Y.)*, 274(5294) :1926–1928. *[page 17]*

Tanja Samardžić, Robert Schikowski et Sabine Stoll. 2015. Automatic interlinear glossing as two-level sequence classification. In *Proceedings of the 9th SIGHUM Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities (LaTeCH)*, pages 68–72, Beijing, China. Association for Computational Linguistics. *[pages 37, 38, 40, 102, 103 et 104]*

Suteera Seeha, Ivan Bilan, Liliana Mamani Sanchez, Johannes Huber, Michael Matuschek et Hinrich Schütze. 2020. ThaiLMCut: Unsupervised pretraining for Thai word segmentation. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 6947–6957, Marseille, France. European Language Resources Association. *[page 25]*

Frank Seifart, Nicholas Evans, Harald Hammarström et Stephen C. Levinson. 2018. Language documentation twenty-five years on. *Language*, 94(4) :e324–e345. *[pages 2, 3 et 33]*

Rico Sennrich, Barry Haddow et Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics. *[pages 16, 29, 39 et 66]*

Libin Shen, Giorgio Satta et Aravind Joshi. 2007. Guided learning for bidirectional sequence classification. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 760–767, Prague, Czech Republic. Association for Computational Linguistics. *[page 37]*

Kairit Sirts et Sharon Goldwater. 2013. Minimally-supervised morphological segmentation using Adaptor Grammars. *Transactions of the Association for Computational Linguistics*, 1 :255–266. *[pages 29, 30, 61 et 131]*

Peter Smit, Sami Virpioja, Stig-Arne Grönroos et Mikko Kurimo. 2014. Morfessor 2.0: Toolkit for statistical morphological segmentation. In *Proceedings of the Demonstrations at the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 21–24, Gothenburg, Sweden. Association for Computational Linguistics. *[pages 66 et 81]*

Felix Stahlberg, Tim Schlippe, Stephan Vogel et Tanja Schultz. 2012. Word segmentation through cross-lingual word-to-phoneme alignment. In *2012 IEEE Spoken Language Technology Workshop (SLT)*, pages 85–90. *[page 25]*

Zhiqing Sun et Zhi-Hong Deng. 2018. Unsupervised neural word segmentation for Chinese via segmental language modeling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4915–4920, Brussels, Belgium. Association for Computational Linguistics. *[page 25]*

Charles Sutton et Andrew McCallum. 2007. An introduction to conditional random fields for relational learning. In Lise Getoor et Ben Taskar, editors, *Introduction to Statistical Relational Learning*, chapter 4, pages 93–127. MIT Press. *[page 115]*

Kumiko Tanaka-Ishii. 2005. Entropy as an indicator of context boundaries : An experiment using a web search engine. In *Natural Language Processing – IJCNLP 2005*, pages 93–105, Berlin, Heidelberg. Springer Berlin Heidelberg. *[page 25]*

Allison Taylor-Adams. 2019. Recording to revitalize: Language teachers and documentation design. *Language Documentation & Conservation*, 13 :426–445. *[page 14]*

Yee Whye Teh. 2006a. A Bayesian interpretation of interpolated Kneser-Ney. Technical Report TRA2/06, School of Computing, National University of Singapore. *[pages 21 et 60]*

Yee Whye Teh. 2006b. A hierarchical Bayesian language model based on Pitman-Yor processes. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 985–992, Sydney, Australia. Association for Computational Linguistics. *[pages 21 et 60]*

Yee Whye Teh, Michael I Jordan, Matthew J Beal et David M Blei. 2006. Hierarchical dirichlet processes. *Journal of the American Statistical Association*, 101(476) :1566–1581. *[pages 20 et 21]*

Robert Tibshirani. 1996. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society : Series B (Methodological)*, 58(1) :267–288. *[page 116]*

Francis Tyers et Robert Henderson. 2021. A corpus of k'iche' annotated for morphosyntactic structure. In *Proceedings of the First Workshop on Natural Language Processing for Indigenous Languages of the Americas*, pages 10–20, Online. Association for Computational Linguistics. *[page 26]*

Kei Uchiumi, Hiroshi Tsukahara et Daichi Mochihashi. 2015. Inducing word and part-of-speech with Pitman-Yor hidden semi-Markov models. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1 : Long Papers)*, pages 1774–1782, Beijing, China. Association for Computational Linguistics. *[page 22]*

UNESCO. 2003. Vitalité et disparition des langues. International Expert Meeting on the UNESCO Programme Safeguarding of Endangered Languages, Paris. *[page 1]*

UNESCO. 2010. *Atlas des langues en danger dans le monde*. Mémoire des peuples. Paris, France. *[page 1]*

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser et Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc. *[pages 27, 29, 38 et 39]*

Sami Virpioja, Oskar Kohonen et Krista Lagus. 2011. Evaluating the effect of word frequencies in a probabilistic generative model of morphology. In *Proceedings of the 18th Nordic Conference of Computational Linguistics (NODALIDA 2011)*, pages 230–237, Riga, Latvia. Northern European Association for Language Technology (NEALT). *[page 95]*

Kilu von Prince et Sebastian Nordhoff. 2020. An empirical evaluation of annotation practices in corpora from language documentation. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 2778–2787, Marseille, France. European Language Resources Association. *[page 11]*

Lihao Wang et Xiaoqing Zheng. 2022. Unsupervised word segmentation with bi-directional neural language model. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, 22(1). *[page 25]*

Xinyi Wang, Sebastian Ruder et Graham Neubig. 2022. Expanding pretrained models to thousands more languages via lexicon-based adaptation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, pages 863–877, Dublin, Ireland. Association for Computational Linguistics. *[page 134]*

Shinji Watanabe, Takaaki Hori, Shigeki Karita, Tomoki Hayashi, Jiro Nishitoba, Yuya Unno, Nelson Enrique Yalta Soplin, Jahn Heymann, Matthew Wiesner, Nanxin Chen, Adithya Renduchintala et Tsubasa Ochiai. 2018. ESPnet: End-to-end speech processing toolkit. In *Proceedings of Interspeech*, pages 2207–2211. *[page 4]*

Søren Wichmann, Eric W. Holman et Cecil H. Brown. 2022. The asjp database (version 20). Online Database. *[page 47]*

P. Wittenburg, U. Mosel et A. Dwyer. 2002. Methods of language documentation in the DOBES project. In *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC'02)*, Las Palmas, Canary Islands - Spain. European Language Resources Association (ELRA). *[page 10]*

Peter Wittenburg, Hennie Brugman, Albert Russel, Alex Klassmann et Han Sloetjes. 2006. ELAN: a professional framework for multimodality research. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*, Genoa, Italy. European Language Resources Association (ELRA). *[pages 2, 3 et 132]*

Anthony C. Woodbury. 2003. Defining documentary linguistics. *Language Documentation and Description*, 1 :35–51. *[pages 2 et 11]*

Anthony C. Woodbury. 2011. *Language documentation*, Cambridge Handbooks in Language and Linguistics, page 159–186. Cambridge University Press. *[page 2]*

Madžid Š. Xalilov. 1999. *Cezsko-russkij slovar´ [Tsez-Russian Dictionary]*. DNC RAN. IJaLI, Makhachkala. *[page 50]*

Fei Xia, Michael Wayne Goodman, Ryan Georgi, Glenn Slayden et William D. Lewis. 2015. Enriching, editing, and representing interlinear glossed text. In *Computational Linguistics and Intelligent Text Processing*, pages 32–46, Cham. Springer International Publishing. *[page 35]*

Fei Xia et William Lewis. 2007. Multilingual structural projection across interlinear text. In *Human Language Technologies 2007 : The Conference of the North American Chapter of the Association for Computational Linguistics ; Proceedings of the Main Conference*, pages 452–459, Rochester, New York. Association for Computational Linguistics. *[pages 12, 34 et 44]*

Fei Xia, William Lewis, Michael Wayne Goodman, Joshua Crowgey et Emily M. Bender. 2014. Enriching ODIN. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 3151–3157, Reykjavik, Iceland. European Language Resources Association (ELRA). *[page 44]*

Jia Xu, Jianfeng Gao, Kristina Toutanova et Hermann Ney. 2008. Bayesian semi-supervised Chinese word segmentation for statistical machine translation. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 1017–1024, Manchester, UK. Coling 2008 Organizing Committee. *[pages 22 et 64]*

Liang Xu, Elaine Uí Dhonnchadha et Monica Ward. 2022. Faoi gheasa an adaptive game for Irish language learning. In *Proceedings of the Fifth Workshop on the Use of Computational Methods in the Study of Endangered Languages*, pages 133–138, Dublin, Ireland. Association for Computational Linguistics. *[page 15]*

Linting Xue, Aditya Barua, Noah Constant, Rami Al-Rfou, Sharan Narang, Mihir Kale, Adam Roberts et Colin Raffel. 2022. ByT5: Towards a token-free future with pre-trained byte-to-byte models. *Transactions of the Association for Computational Linguistics*, 10 :291–306. *[pages 39 et 40]*

Racquel-María Yamada. 2011. Integrating documentation and formal teaching of kari'nja: Documentary materials as pedagogical materials. *Language Documentation & Conservation*, 5 :1–30. *[page 14]*

Olga Zamaraeva, Kristen Howell et Emily M. Bender. 2019. Handling cross-cutting properties in automatic inference of lexical classes: A case study of chintang. In *Proceedings of the 3rd Workshop on the Use of Computational Methods in the Study of Endangered Languages Volume 1 (Papers)*, pages 28–38, Honolulu. Association for Computational Linguistics. *[page 34]*

Olga Zamaraeva, František Kratochvíl, Emily M. Bender, Fei Xia et Kristen Howell. 2017. Computational support for finding word classes: A case study of Abui. In *Proceedings of the 2nd Workshop on the Use of Computational Methods in the Study of Endangered Languages*, pages 130–140, Honolulu. Association for Computational Linguistics. *[page 34]*

Marcely Zanon Boito. 2021. *Models and resources for attention-based unsupervised word segmentation : an application to computational language documentation*. Theses, Université Grenoble Alpes [2020-....]. *[page 4]*

Roberto Zariquiey, Arturo Oncevay et Javier Vera. 2022. CLD^2 language documentation meets natural language processing for revitalising endangered languages. In *Proceedings of the Fifth Workshop on the Use of Computational Methods in the Study of Endangered Languages*, pages 20–30, Dublin, Ireland. Association for Computational Linguistics. *[page 46]*

Xingyuan Zhao, Satoru Ozaki, Antonios Anastasopoulos, Graham Neubig et Lori Levin. 2020. Automatic interlinear glossing for under-resourced languages leveraging translations. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5397–5408, Barcelona, Spain (Online). International Committee on Computational Linguistics. *[pages 33, 36, 38, 40, 50, 51, 104, 105, 114, 120, 125, 126 et 133]*

Zhong Zhou, Lori S. Levin, David R. Mortensen et Alex Waibel. 2019. Low-resource machine translation using interlinear glosses. *CoRR*, abs/1911.02709. *[page 35]*

George K. Zipf. 1935. *The Psycho-Biology of Language*. Houghton Mifflin, Boston. *[pages 17 et 18]*

Hui Zou et Trevor Hastie. 2005. Regularization and Variable Selection Via the Elastic Net. *Journal of the Royal Statistical Society Series B : Statistical Methodology*, 67(2) :301–320. *[page 116]*

Chapitre A

Annexes

A.1 Inférence dans dpseg et pypseg

Cette partie constitue un complément à la section 2.2.3. Elle provient du document de travail rédigé pour le projet CLD2025, qui reprend les travaux de (Goldwater et al., 2009).

Comme le modèle basé sur un processus de Dirichlet est équivalent à un modèle CRP à deux étages (Goldwater et al., 2009) (voir section 2.2.3), nous pouvons calculer la probabilité d'avoir un mot d'étiquette l, conditionnellement aux dispositions des tables ($\boldsymbol{z_{-i}}$) et des étiquettes ($l(\boldsymbol{z_{-i}})$) :

$$\begin{aligned}
P(w_i = l|\boldsymbol{z_{-i}}, l(\boldsymbol{z_{-i}}), \alpha) &= \sum_{k=1}^{K(\boldsymbol{z_{-i}})+1} P(w_i = l|z_i = k, l(\boldsymbol{z_{-i}})) * P(z_i = k|\boldsymbol{z_{-i}}, \alpha) \\
&= \sum_{k=1}^{K(\boldsymbol{z_{-i}})} P(w_i = l|z_i = k, l_k) * P(z_i = k|\boldsymbol{z_{-i}}, \alpha) \\
&\quad + P(w_i = l|z_i = K(\boldsymbol{z_{-i}}) + 1) * P(z_i = K(\boldsymbol{z_{-i}}) + 1|\boldsymbol{z_{-i}}, \alpha) \\
&= \sum_{k=1}^{K(\boldsymbol{z_{-i}})} \mathbb{1}_{(l_k=l)}(l) * \frac{n_k(\boldsymbol{z_{-i}})}{i-1+\alpha} + P_0(l) * \frac{\alpha}{i-1+\alpha}
\end{aligned}$$

où $P_0(l)$ correspond à la probabilité de générer un nouveau mot l, selon la distribution de base définie par l'équation (2.3), et $\mathbb{1}_{(l_k=l)}(l)$ à la fonction indicatrice valant 1 si l'étiquette de la table k est l, 0 sinon.

En notant $n_l^{(\boldsymbol{w_{-i}})}$, le nombre total de mot l dans le lexique ($\boldsymbol{w_{-i}}$), nous obtenons :

$$P(w_i = l|\boldsymbol{z_{-i}}, l(\boldsymbol{z_{-i}}), \alpha) = \frac{n_l^{(\boldsymbol{w_{-i}})} + \alpha P_0(l)}{i-1+\alpha} \tag{A.1}$$

Par conséquent, la probabilité que le client i s'assied à une table d'étiquette l, étant donné la disposition actuelle des mots $\boldsymbol{w_{-i}}$, s'écrit :

$$\begin{aligned}
P(w_i = l|\boldsymbol{w_{-i}}, \alpha) &= \sum_{\{\boldsymbol{z_{-i}}, l(\boldsymbol{z_{-i}})\}} P(w_i = l|\boldsymbol{z_{-i}}, l(\boldsymbol{z_{-i}}), \alpha) * P(\boldsymbol{z_{-i}}, l(\boldsymbol{z_{-i}})|\alpha) \\
&= \frac{n_l^{(\boldsymbol{w_{-i}})} + \alpha P_0(l)}{i-1+\alpha} * \underbrace{\sum_{\{\boldsymbol{z_{-i}}, l(\boldsymbol{z_{-i}})\}} P(\boldsymbol{z_{-i}}, l(\boldsymbol{z_{-i}})|\alpha)}_{=1}
\end{aligned}$$

C'est ainsi que nous retrouvons l'équation (2.5).

$$P(w_i = l|\boldsymbol{w_{-i}}, \alpha) = \frac{n_l^{(\boldsymbol{w_{-i}})} + \alpha P_0(l)}{i-1+\alpha}$$

Pour l'échantillonnage, à une position donnée de frontière b_i, il existe donc deux possibilités : soit il n'y a pas de frontière ($b_i = 0$), soit il y a une frontière ($b_i = 1$). Les deux hypothèses à considérer peuvent alors être formulées comme :

$h_1 = \gamma_p w_1 \gamma_s$ (absence de frontière) et $h_2 = \gamma_p w_2 w_3 \gamma_s$ (présence de frontière entre w_2 et w_3),

où γ_p représente les mots précédents et γ_s les mots suivants dans la phrase, par rapport à b_i. Nous noterons par la suite h^-, les mots dans γ_p et γ_s, c'est-à-dire tous les mots autres que w_1, w_2 et w_3.

Comme nous avons uniquement besoin du rapport relatif entre les deux probabilités pour sélectionner l'hypothèse, nous nous intéressons au numérateur des probabilités :

$$P(b_i = 0|h^-, d) = \frac{P(h_1|h^-, d)}{P(h_1|h^-, d) + P(h_2|h^-, d)} \propto P(h_1|h^-, d)$$
$$P(b_i = 1|h^-, d) = \frac{P(h_2|h^-, d)}{P(h_1|h^-, d) + P(h_2|h^-, d)} \propto P(h_2|h^-, d)$$

Avec la règle de Bayes, nous obtenons : $P(h_1|h^-, d) = \frac{P(d|h_1, h^-) * P(h_1|h^-)}{P(d|h^-)} = \frac{P(h_1|h^-)}{P(d|h^-)}$.
En effet, $P(d|h_1, h^-) = 1$.
Nous avons alors :

$$P(b_i = 0|h^-, d) \propto P(h_1|h^-)$$
$$P(b_i = 1|h^-, d) \propto P(h_2|h^-)$$

Dans le cas sans frontière, en utilisant également l'équation (2.6) :

$$\begin{aligned} P(h_1|h^-) &= P(w_1|h^-)P(u_{w_1}|h^-) \\ &= \frac{n_{w_1}^{(h^-)} + \alpha P_0(w_1)}{n^- + \alpha} * \frac{n_u^{(h^-)}(w) + \frac{\rho}{2}}{n^- + \rho} \end{aligned}$$

où n^- représente le nombre total de mots dans h^- (équivalent à $i-1$ dans l'équation (A.1)) car le modèle est échangeable. La valeur de $n_u^{(h^-)}(w)$ dépend de la nature de w : elle vaut $n_\$$, indiquant le nombre total de phrases, si le mot w finit la phrase, et $n^- - n_\$$ sinon.
Dans le cas avec une frontière,

$$\begin{aligned} P(h_2|h^-) &= P(w_2|h^-)P(u_{w_2}|h^-) * P(w_3|w_2, h^-)P(u_{w_3}|w_3, h^-) \\ &= \frac{n_{w_2}^{(h^-)} + \alpha P_0(w_2)}{n^- + \alpha} * \frac{n^- - n_\$ + \frac{\rho}{2}}{n^- + \rho} \\ &\quad * \frac{n_{w_3}^{(h^-)} + \mathbb{1}_{(w_2=w_3)}(w_2, w_3) + \alpha P_0(w_3)}{n^- + 1 + \alpha} * \frac{n_u^{(h^-)}(w) + \mathbb{1}_{(u_{w_2}=u_{w_3})}(u_{w_2}, u_{w_3}) + \frac{\rho}{2}}{n^- + 1 + \rho} \end{aligned}$$

En enlevant les termes en commun pour les deux hypothèses, nous obtenons finalement :

$$P(h_1|h^-) \propto n_{w_1}^{(h^-)} + \alpha P_0(w_1) \tag{A.2}$$

$$P(h_2|h^-) \propto (n_{w_2}^{(h^-)} + \alpha P_0(w_2)) * \frac{n_{w_3}^{(h^-)} + \mathbb{1}_{(w_2=w_3)}(w_2, w_3) + \alpha P_0(w_3)}{n^- + 1 + \alpha} * \frac{n^- - n_\$ + \frac{\rho}{2}}{n^- + 1 + \rho} \tag{A.3}$$

Enfin, la valeur de b_i est déterminée par la comparaison des expressions (A.2) et (A.3), en testant si un nombre aléatoire est inférieur à $P(h_2|h^-)$, ce qui indique la présence d'une frontière, le cas échéant.

Dans le cas Pitman-Yor En utilisant cette fois l'équation (2.8) à la place de (2.2), on obtient alors :

$$P(h_1|h^-) \propto \varphi(w_1, \alpha, d) \tag{A.4}$$

$$P(h_2|h^-) \propto \varphi(w_2, \alpha, d) * \frac{\varphi(w_3, \alpha, d) + \mathbb{1}_{(w_2=w_3)}(w_2, w_3)}{n^- + 1 + \alpha} * \frac{n^- - n_\$ + \frac{\rho}{2}}{n^- + 1 + \rho} \tag{A.5}$$

où, pour une meilleure lisibilité, nous notons $\varphi(w, \alpha, d)$ le numérateur obtenu lors du calcul, défini ci-dessous :

$$\varphi(w, \alpha, d) = n_w^{(\boldsymbol{w_{-i}})} - d \sum_{k=1}^{K(\boldsymbol{z_{-i}})} \mathbb{1}_{(l_k=w)}(w) + dK(\boldsymbol{z_{-i}})P_0(w) + \alpha P_0(w) \tag{A.6}$$

A.2 Rappel des champs aléatoires conditionnels (CRF)

Les champs aléatoires conditionnels (ou, en anglais, *Conditional Random Fields* ; CRF) (Lafferty et al., 2001) sont caractérisés par l'équation (A.7) suivante, définissant la probabilité de la séquence de T étiquettes $\boldsymbol{y} = (y_1, ..., y_T)$ pour une séquence en entrée $\boldsymbol{x} = (x_1, ..., x_T)$ de T éléments :

$$p_\theta(\boldsymbol{y}|\boldsymbol{x}) = \frac{\exp\left\{\sum_{k=1}^{K} \theta_k G_k(\boldsymbol{x}, \boldsymbol{y})\right\}}{\sum_{\boldsymbol{y'} \in \mathcal{Y}^T} \exp\left\{\sum_{k=1}^{K} \theta_k G_k(\boldsymbol{x}, \boldsymbol{y'})\right\}} = \frac{1}{Z_\theta(\boldsymbol{x})} \exp\left\{\sum_{k=1}^{K} \theta_k G_k(\boldsymbol{x}, \boldsymbol{y})\right\}, \tag{A.7}$$

où les $G_k, k \in [1, K]$ représentent les fonctions caractéristiques et $\theta_k \in \mathbb{R}$ le poids associé. Les fonctions caractéristiques évaluent des propriétés locales et peuvent se décomposer sous la forme $G_k(\boldsymbol{x}, \boldsymbol{y}) = \sum_i g_k(y_i, y_{i-1}, i, \boldsymbol{x})$, en testant par exemple l'étiquette de la position i (y_i ; unigramme) ou deux positions consécutives (le couple (y_{i-1}, y_i) ; bigramme). Enfin, la fonction de partition $Z_\theta(\boldsymbol{x})$ correspond à la constante de normalisation, qui somme sur toutes les séquences d'étiquettes possibles $\mathcal{Y}^T$, en notant $\mathcal{Y}$ l'ensemble des étiquettes.

L'inférence s'effectue alors en maximisant la log-vraisemblance ; dans le cas où seules des caractéristiques bigrammes sont considérées, la complexité du modèle est quadratique en nombre d'étiquettes.

A.3 Résultats de la segmentation à deux niveaux en tsez

Cette section présente les résultats *complets* de la segmentation à deux niveaux sur le corpus *tsez* correspondant aux sections 5.3.3 et 5.3.5. Dans l'ensemble, nous observons les mêmes tendances que sur le corpus japhug. Les modèles couplés ou hiérarchiques sont meilleurs que l'approche en cascade et la supervision faible est bénéfique, mais de nouveau de manière trop modeste.

modèle	AG		dpseg*		pipe.	parallel-w		hier-type		-final	hier-iter	
niveau	W	M	W	M	M	W	M	W	M	M	W	M
BP	**67,3**	78,1	59,9	**91,8**	59,9	69,9	89,3	64,0	47,8	74,4	64,7	74,7
BR	76,6	85,5	**87,9**	63,9	**88,8**	87,4	71,6	83,0	81,7	86,1	77,6	85,1
BF	71,6	**81,6**	71,3	75,3	78,2	70,9	79,5	**72,2**	60,3	79,8	70,6	79,6
WP	**41,6**	55,6	33,3	52,1	46,0	32,8	**57,7**	38,2	19,0	50,8	38,8	51,4
WR	46,7	**60,5**	47,4	37,1	57,8	46,6	46,8	**48,4**	31,9	58,3	45,7	58,2
WF	**44,0**	**57,9**	39,1	43,4	51,2	38,5	51,7	42,7	23,8	54,3	42,0	54,6
LP	45,9	**51,3**	**49,6**	41,4	41,2	49,0	47,7	47,3	24,2	41,1	42,3	43,1
LR	**28,8**	28,0	16,9	**50,4**	16,6	16,7	47,6	22,6	13,1	20,1	25,5	21,8
LF	**35,4**	36,2	25,2	45,5	23,6	25,0	**47,7**	30,5	17,0	27,0	31,8	29,0
WL	4,99	2,58	3,95		2,24	3,95	3,46	4,43	1,68	2,45	4,76	2,48
TL	6,52	3,52	4,53		2,89	4,52	4,32	4,95	2,87	3,07	5,35	3,08
N_{type}	3597	875	1950		646	1958	1600	2732	867	786	3456	812
N_{token}	22,7k	43,8k	28,6k		50,5k	28,6k	32,7k	25,6k	67,4k	46,2k	23,8k	45,5k

TABLE A.1 – Résultats *complets* des modèles de segmentation à un (marqué par *) et deux niveaux (W pour les mots, M pour les morphèmes) sur le corpus *tsez* sans supervision (modèles en cascade et couplés ; moyenne de trois lancers). Les meilleurs scores par métrique et niveau de segmentation sont en **gras**.

modèle	CRF		dpseg		pipe.	parallel-w		hier-type		-final	hier-iter	
niveau	W	M	W	M	M	W	M	W	M	M	W	M
BP	**83,3**	85,9	65,4	**93,3**	83,2	65,3	90,6	69,1	65,6	85,0	69,5	84,0
BR	78,3	82,5	**90,7**	69,3	**95,9**	90,6	74,7	83,6	88,7	92,9	80,7	91,7
BF	**80,7**	84,2	76,0	79,5	**89,1**	75,9	81,9	75,7	75,4	88,8	74,7	87,7
WP	**64,5**	67,8	42,5	61,8	71,9	42,2	63,2	46,6	46,4	**72,5**	46,9	70,6
WR	**60,9**	65,3	57,3	46,7	**82,4**	56,9	52,6	55,4	62,0	79,0	53,8	76,8
WF	**62,6**	66,6	48,8	53,2	**76,8**	48,4	57,4	50,6	53,1	75,6	50,1	73,6
LP	47,6	21,9	**62,7**	49,1	**61,9**	62,4	53,4	53,8	46,5	59,8	50,6	59,3
LR	**61,0**	**62,0**	26,9	57,5	36,7	26,7	54,6	32,7	33,8	38,9	34,4	38,5
LF	**53,5**	32,4	37,6	53,0	46,1	37,3	**54,0**	40,6	39,1	47,1	41,0	46,7
WL	5,94	2,92	4,16	3,72	2,46	4,16	3,38	4,72	2,11	2,58	4,90	2,59
TL	7,83	5,98	5,02	4,58	3,67	5,03	4,40	5,43	3,49	3,70	5,61	3,67
N_{type}	7343	4537	2458	1877	950	2450	1639	3479	1165	1043	3902	1041
N_{token}	19,0k	38,7k	27,2k	30,4k	46,1k	27,2k	33,5k	24,0k	53,7k	43,8k	23,1k	43,7k

TABLE A.2 – Résultats *complets* des modèles CRF, dpseg et ses extensions à deux niveaux (W pour les mots, M pour les morphèmes) sur le corpus *tsez*, supervisés par des annotations denses (**sentence**) de 200 phrases (moyenne de trois lancers). Les meilleurs scores par métrique et niveau de segmentation sont en **gras**.

A.4 Quantité de caractéristiques actives pour la génération de gloses

Le tableau A.4 présente le nombre de caractéristiques sélectionnées (en milliers) parmi toutes les caractéristiques calculées (en millions) par S3. Nous remarquons que grâce à la régularisa-

modèle	dpseg		pipe.	parallel-w		hier-type		-final	hier-iter	
niveau	W	M	M	W	M	W	M	M	W	M
BP	73,2	**95,8**	90,6	**73,4**	94,3	66,0	58,0	87,1	66,6	87,7
BR	84,9	58,5	79,6	84,9	63,1	**91,2**	82,0	84,5	90,5	**85,4**
BF	78,6	72,6	84,7	**78,7**	75,6	76,6	67,9	85,8	76,7	**86,5**
WP	50,3	49,7	66,1	**50,5**	53,1	43,0	29,1	65,8	43,6	**67,7**
WR	57,6	31,3	58,4	57,7	36,4	**57,7**	40,5	64,0	57,7	**66,1**
WF	53,7	38,4	62,0	**53,9**	43,2	49,3	33,8	64,9	49,7	**66,9**
LP	62,0	38,0	49,8	**62,1**	41,5	59,9	43,2	53,7	60,4	**55,2**
LR	37,2	**64,6**	54,1	**37,3**	63,1	26,9	36,2	44,9	27,7	45,9
LF	46,5	47,9	**51,9**	**46,6**	50,0	37,1	39,4	48,9	37,9	50,1
WL	4,91	4,47	3,18	4,92	4,10	4,18	2,02	2,89	4,24	2,88
TL	5,86	5,39	4,38	5,88	5,11	4,82	3,73	3,94	4,88	3,94
N_{type}	3442	2725	1744	3449	2441	2571	1342	1339	2624	1332
N_{token}	23,1k	25,3k	35,6k	23,0k	27,6k	27,1k	56,1k	39,1k	26,7k	39,2k

TABLE A.3 – Résultats *complets* des modèles dpseg et ses extensions à deux niveaux (W pour les mots, M pour les morphèmes) sur le corpus *tsez*, supervisés par des dictionnaires de mots et de morphèmes (**dictionary**) obtenus sur 200 phrases (moyenne de trois lancers). Les meilleurs scores par métrique et niveau de segmentation sont en **gras**.

tion l_1, un poids nul est attribué à la plupart des caractéristiques, ce qui implique ici que moins de 1 % des caractéristiques sont retenues en fin de compte.

langue	ddo	git	lez	ntu	usp
actif (S3)	196k	4k	51k	73k	151k
total	188M	1M	26M	44M	29M

TABLE A.4 – Nombre de caractéristiques sélectionnées parmi toutes les caractéristiques calculées par le système S3 dans chaque langue.