Modèle-s bayés-ien-s pour la segment-ation à deux niveau-x faible-ment super-vis-é-e

Shu Okabe, François Yvon

Modèle-s bayés-ien-s pour la segment-ation à deux niveau-x faible-ment super-vis-é-e

Shu Okabe[1] François Yvon[1]
(1) Université Paris-Saclay, CNRS, LISN, Bât. 508, Rue du Belvédère, F-91405 Orsay, France
shu.okabe@limsi.fr, francois.yvon@limsi.fr

RÉSUMÉ

La segmentation automatique en mots et en morphèmes est une étape cruciale dans le processus de documentation des langues. Dans ce travail, nous étudions plusieurs modèles bayésiens pour réaliser une segmentation conjointe des phrases à ces deux niveaux : d'une part, en introduisant un couplage déterministe entre deux modèles spécialisés pour identifier chaque type de frontières, d'autre part, en proposant une modélisation intrinsèquement hiérarchique. Un objectif important de cette étude est de comparer ces modèles dans un scénario où une supervision faible est disponible. Nos expériences portent sur deux langues et permettent de comparer dans des conditions réalistes les mérites de ces diverses modélisations.

ABSTRACT

Bayesian models for weakly supervised two-level segmentation

Automatic segmentation into words and morphemes is a crucial step in the process of language documentation. In this work, we study several Bayesian models for simultaneously segmenting sentences at two levels: first, relying on a deterministic coupling between models specialised in identifying each type of boundary; second, using an intrinsically hierarchical model. An important objective of this study is to compare these two approaches in a setting where weak supervision is available. Our experiments concern two languages and allow us to compare the merits of these various models under realistic conditions.

MOTS-CLÉS : segmentation en mots, segmentation en morphèmes, documentation automatique des langues, modèle bayésien non-paramétrique.

KEYWORDS: word segmentation, morpheme segmentation, computational language documentation, Bayesian non-parametric model.

1 Introduction

La segmentation en mots ou en morphèmes vise à identifier les frontières de ces unités dans une séquence de symboles, correspondant à la représentation orthographique ou phonétique d'une phrase ou d'un mot isolé. Nous nous intéressons ici à une approche conjointe à ces deux problèmes : le calcul d'une segmentation à deux niveaux. Comme l'illustre la figure 1, à partir d'une phrase non segmentée (représentée par une chaîne de caractères), l'objectif est de retrouver les frontières de mots (matérialisées par des espaces) et de morphèmes (indiquées par les tirets).

Phrase non-segmentée	ɯʑokɯatɕɯpɯwɣsɯmto-a
Segmentation en mots	ɯʑo kɯ atɕɯ pɯwɣsɯmtoa
Segmentation en morphèmes	ɯʑo-kɯ-a-tɕɯ-pɯ-wɣ-sɯ-mto-a
Segmentation à deux niveaux	ɯʑo kɯ a-tɕɯ pɯ-wɣ-sɯ-mto-a
Glose	3SG ERG 1SG.POSS-fils AOR-INV-CAUS-voir-1SG
Traduction	Il m'a laissé voir mon fils

FIGURE 1 – Exemple de segmentation en japhug : les mots sont séparés par des espaces (« »), tandis que les morphèmes le sont par des tirets (« - »). Extrait de (Jacques, 2021)

L'exemple de la figure 1 permet également de situer le cadre de ce travail : la segmentation en mots et en morphèmes d'énoncés est une étape clef dans le processus de documentation des langues. Au premier niveau (mot), elle permet de constituer des dictionnaires à partir de retranscriptions phonétiques ou orthographiques d'enregistrements recueillis sur le terrain, alors que la segmentation au second niveau (morphème) accompagne l'annotation morphosyntaxique et la glose.

Les modèles bayésiens non-supervisés de segmentation en mots (voir notamment (Johnson *et al.*, 2007; Goldwater *et al.*, 2009; Godard, 2019) et les références citées dans ces travaux) conduisent à la découverte d'unités de nature incertaine, correspondant soit à des mots, soit à des morphèmes (voir les résultats du tableau 1). Dans ce travail, nous étudions des modèles qui permettent de distinguer explicitement ces deux types d'unités, en nous intéressant plus spécialement à la possibilité de superviser partiellement cette double segmentation, par exemple en utilisant des listes de mots ou de morphèmes préexistantes. Dans un contexte de documentation des langues, il peut s'avérer utile de contrebalancer la petite taille des corpus par des ressources linguistiques, comme des listes de mots qui peuvent aider à apprendre un modèle de segmentation. Les contributions principales de ce travail sont alors (a) la spécification de plusieurs modèles bayésiens non-paramétriques pour cette double segmentation ; (b) la comparaison expérimentale de leurs mérites respectifs pour deux langues et deux stratégies de supervision.

2 Méthodes

2.1 Rappel : segmentation à un niveau

dpseg Notre modèle de base est la version unigramme du modèle de Goldwater *et al.* (2009), dénoté `dpseg` dans la suite. Il évalue la probabilité d'une séquence de mots en se basant sur les processus de Dirichlet, en s'appuyant sur l'équation (1) pour un mot $w = c_1...c_L$ de L caractères :

$$P(w|h^-;\alpha) = \frac{n_w^{(h^-)} + \alpha P_0(w|h^-)}{n^- + \alpha} \tag{1}$$

où h^- dénote le reste du texte (w exclu), $n_w^{(h^-)}$ la fréquence de w et n^- le nombre total de mots dans h^- ; α est le paramètre de concentration. P_0 est défini par l'équation (2) :

$$P_0(w) = p_\#(1-p_\#)^{(L-1)} * \prod_{l=1}^{L} P_c(c_l) \tag{2}$$

avec $p_\#$ la probabilité de finir un mot et P_c la probabilité du caractère c_l, qui suit une distribution uniforme dans le modèle `dpseg`. L'équation (1) est utilisée pour évaluer, à chaque position dans le texte, s'il y a une frontière ou non après le caractère considéré. Dans nos expériences, nous utilisons notre propre ré-implémentation en Python de ce modèle [1]. Dans cette implémentation, l'inférence est basée sur un échantillonnage de Gibbs avec recuit simulé.

Le modèle `dpseg` a été choisi car il s'est révélé meilleur que d'autres modèles tels que SentencePiece (Kudo & Richardson, 2018) ou Morfessor (Smit *et al.*, 2014), dans des situations similaires de documentation automatique des langues. Il est par ailleurs adapté au traitement de petits corpus et permet d'expérimenter des formes de supervision variées (Okabe *et al.*, 2022).

Superviser la segmentation Comme dans (Okabe *et al.*, 2022), nous considérons deux types de ressources pour la supervision faible : d'une part, un petit nombre de phrases entièrement segmentées, d'autre part, des listes d'unités (mots ou morphèmes). Dans nos expériences, les 200 premières phrases du corpus sont utilisées pour constituer ces ressources de la manière suivante : pour la supervision avec les phrases annotées (méthode `sentence`), l'inférence n'échantillonne pas la présence (ou absence) de frontière sur ces phrases et utilise directement la valeur observée ; pour la supervision par un dictionnaire (méthode `dictionary`), une liste de types [2] est extraite et utilisée pour estimer un modèle bigramme de caractères qui remplace le modèle uniforme pour P_0 dans l'équation (2) et augmente ainsi la probabilité des mots ou des morphèmes connus.

2.2 Couplage de modèles segmentant en parallèle

À partir du modèle `dpseg`, un premier modèle à deux niveaux a été implémenté en couplant de manière déterministe deux modèles qui agissent en « parallèle » et de manière synchrone : l'un segmente en mots, tandis que l'autre segmente en morphèmes. Le point crucial dans cette stratégie de segmentation est l'interaction des deux modèles qui peut être implémentée de deux manières : soit i) en imposant que les frontières de mots sont nécessairement des frontières de morphèmes ; soit ii) que les non frontières de morphèmes sont également des non frontières de mots.

Dans le cas i), on échantillonne d'abord les frontières de mots, puis de morphèmes (modèle `parallel-w`). Lorsqu'une frontière de mots est identifiée, on impose également une frontière de morphème (sans échantillonnage). Dans le cas contraire, la position courante est considérée comme une position interne, et l'on échantillonne pour décider le statut de la frontière (ou non frontière) de morphème. Dans le second cas (ii), on échantillonne les frontières de morphèmes puis les frontières de mots (modèle `parallel-m`) : s'il y a une frontière de morphème, le modèle échantillonne pour décider s'il y a également une frontière de mot. Sinon, une non frontière de morphème implique l'absence de frontière de mot. Ces deux stratégies assurent qu'à tout moment les hypothèses de segmentation en mots et morphèmes sont cohérentes entre elles.

2.3 Segmentation hiérarchique

Un modèle hiérarchique (`hierarchical`), inspiré de celui de (Mochihashi *et al.*, 2009) a également été implémenté pour la segmentation à deux niveaux. Ce modèle vise à prendre en compte

1. https://github.com/shuokabe/pyseg.

2. Nous utilisons les termes « occurrence » pour les mots-formes apparaissant dans le texte (en anglais *token*) et « type » pour désigner les mots uniques.

explicitement le caractère structuré de cette double segmentation. Dans cette approche, le modèle de mot est identique à la version de base de `dpseg`, à l'exception de la définition de la distribution de base P_0 dans l'équation (1). Le modèle de caractères est remplacé par un deuxième modèle non-paramétrique (d'où la nature hiérarchique du modèle) pour les morphèmes, basé lui aussi sur `dpseg`. Ce modèle de morphèmes possède également une distribution de base qui est, comme dans le modèle `dpseg`, un modèle unigramme de caractères.

En considérant un mot w (de longueur L) composé de K morphèmes, $w = m_1 m_2 ... m_K$, P_0 peut donc s'écrire, par analogie avec l'équation (2) :

$$P_0(w = m_1 ... m_K | h^-) = p_\#(1 - p_\#)^{(L-1)} * \prod_{k=1}^{K} P_m(m_k | h^-) \tag{3}$$

où $P_m(m_k)$ est la probabilité du morphème m_k selon le modèle de morphèmes.

En échantillonnant à travers chaque position dans le mot pour trouver les frontières de morphèmes, le modèle peut aboutir à la segmentation en morphèmes la plus probable pour w. Cette segmentation sera à son tour considérée pour calculer P_0 (équation (3)).

Par analogie avec le modèle de mot (équation (1)), le modèle de morphèmes s'écrit :

$$P_m(m_k | h^-; \alpha_m) = \frac{n_{m_k}^{(h^-)} + \alpha_m Q_0(m_k)}{n_{m}^{-} + \alpha_m} \tag{4}$$

où α_m est le paramètre de concentration pour les morphèmes, Q_0 est le modèle de base, qui repose sur un modèle unigramme de caractères, identique à celui du modèle `dpseg` standard. Lors de l'entraînement, il est nécessaire d'effectuer un échantillonnage de Gibbs au niveau des mots et au niveau des morphèmes. [3] Les deux types de modèles à deux niveaux exigent un paramètre de concentration par niveau : α pour les mots et α_m pour les morphèmes. Ces deux paramètres sont initialisés avec la même valeur, puis réestimés au fil des itérations.

Quant à la supervision, chaque niveau est supervisé par les ressources correspondantes : d'une part, les annotations de segmentation en mots ou les dictionnaires de mots sont utilisés pour calculer $P(w|h^-, \alpha)$ (équation (1)), d'autre part, les frontières de morphèmes et les listes de morphèmes interviendront au niveau des morphèmes (comme l'équation (4) pour le modèle hiérarchique).

3 Expériences

3.1 Métriques d'évaluation et données expérimentales

Métriques Comme dans (Goldwater *et al.*, 2009), les modèles de segmentation sont évalués à travers trois F-scores principalement : BF au niveau des frontières (***B**oundary **F**-score*), WF au niveau des occurrences dans chaque phrase (***W**ord token **F**-score*) et LF au niveau des types de mots (ou morphèmes) dans le texte entier (***L**exicon **F**-score*). À titre indicatif, les précisions (***P**recision*) et rappels (***R**ecall*) respectifs (BP, WP, LP et BR, WR, LR) sont donnés dans les deux premiers tableaux.

3. Dans notre implémentation, l'initialisation repose sur une segmentation aléatoire à deux niveaux, dont on déduit les statistiques initiales ; la segmentation en morphèmes de chaque mot potentiel est recalculée chaque fois que l'on évalue sa probabilité (y compris pour des mots déjà identifiés par le modèle).

Par ailleurs, des statistiques descriptives sur les corpus sont également présentées : WL et TL pour les longueurs moyennes des occurrences et des types, ainsi que N_{type} et N_{token} pour les nombres de types et d'occurrences, respectivement.

Langues étudiées Nous avons travaillé avec deux langues morphologiquement complexes : d'une part le japhug, langue sino-tibétaine très peu dotée, d'autre part le tsez (Dido), langue caucasienne faisant partie de la famille des langues nakho-daghestaniennes. Les expériences se concentreront particulièrement sur le japhug, langue en cours de documentation, et qui fait l'objet des sections 3.2 et 3.3 ; les résultats concernant le tsez sont présentés à la section 3.4.

Les données en japhug proviennent des phrases d'exemples du livre de grammaire de Jacques (2021) (3 628 phrases). Pour le tsez, nous utilisons les phrases du *Tsez Annotated Corpus* de Abdulaev & Abdulaev (2010) (2 000 phrases), présenté dans Zhao *et al.* (2020) pour une tâche de génération automatique de gloses interlinéaires. Les deux corpus sont segmentés en mots et morphèmes ; leurs statistiques générales seront présentées dans les parties suivantes en tant que référence. Pour rappel, les 200 premières phrases sont utilisées comme ressource de supervision.

Paramètres Pour les expériences, 20 000 itérations de l'échantillonneur de Gibbs ont été effectuées, avec 10 palliers de température pour le recuit simulé. Les méta-paramètres ont pour valeur initiale : $p_{\#} = 0{,}5$ et $\alpha = \alpha_m = 20$, qui sont les valeurs par défaut du modèle `dpseg` de référence. Enfin, le paramètre de concentration α est ré-échantillonné après chaque itération sur le corpus (et donc $\alpha \neq \alpha_m$ en sortie), en suivant Teh (2006) et Mochihashi *et al.* (2009).

3.2 Segmentation(s) non-supervisée(s)

modèle	référence		`dpseg`		`parallel-w`		`parallel-m`		`hierarchical`	
niveau	mot	morph.	mot	morph.	mot	morph.	mot	morph.	mot	morph.
BP			61,10	87,59	61,53	85,01	64,66	87,40	**68,22**	**88,00**
BR			90,20	75,02	**90,93**	**82,52**	84,43	74,85	74,65	75,57
BF			72,85	80,82	**73,39**	**83,75**	73,24	80,64	71,29	81,31
WP			38,98	58,91	39,78	**62,49**	41,01	58,66	**42,17**	60,14
WR			55,18	51,12	**56,37**	**60,80**	51,95	50,89	45,65	52,31
WF			45,69	54,74	**46,65**	**61,63**	45,84	54,50	43,84	55,96
LP			39,95	45,53	**41,03**	**50,98**	39,07	45,14	38,77	46,33
LR			13,38	37,64	13,77	34,31	17,08	37,72	**24,11**	**38,81**
LF			20,05	41,21	20,62	41,02	23,77	41,09	**29,73**	**42,24**
WL	4,73	2,90	3,34		3,34	2,98	3,74	3,34	4,37	3,34
TL	7,30	5,41	4,21		4,23	4,01	4,79	4,21	5,94	4,38
N_{type}	6739	2731	2258		2262	1838	2946	2282	4191	2288
N_{token}	28579	46632	40463		40499	45369	36204	40454	30932	40559

TABLE 1 – Résultats sur le corpus japhug pour `dpseg` et ses versions à deux niveaux non supervisés

Le tableau 1 présente les résultats de quatre modèles de segmentation non supervisés : la *baseline* `dpseg` [4] ainsi que les trois modèles présentés ci-dessus.

Nous pouvons observer tout d'abord que la *baseline* a des performances moins élevées que le modèle `parallel-w` et assez proches du modèle `parallel-m`, sachant que la *baseline* ne permet pas de différencier une frontière de mot d'une frontière de morphème (toutes les unités sont séparées par des espaces). Par ailleurs, le modèle hiérarchique n'apporte qu'une légère amélioration par rapport au modèle de base, insuffisant pour surpasser le modèle `parallel-w` pour la majorité des métriques.

4. Ce modèle fournit une segmentation unique, qui est comparée avec les deux segmentations de référence.

Notons toutefois que dans ces expériences, le modèle hiérarchique est celui qui effectue la plus forte distinction entre les deux types d'unités, dont les longueurs moyennes (WL et TL) sont clairement distinguées. Les autres modèles ont une tendance claire à la sur-segmentation, conduisant à identifier un nombre trop faible de types (au niveau des mots), ce que montre également la métrique LF.

3.3 Segmentations faiblement supervisées

	sentence								dictionary							
modèle	dpseg		parallel-w		parallel-m		hier.		dpseg		parallel-w		parallel-m		hier.	
niveau	mot	morph.	mot	morph.	mot	morph.	mot	morph.	mot	morph.	mot	morph.	mot	morph.	mot	morph.
BP	63,75	86,31	64,33	86,37	66,42	88,80	**70,72**	**89,19**	76,62	**93,16**	**76,75**	90,85	76,14	93,06	70,33	91,28
BR	91,19	83,17	**91,66**	**83,38**	86,12	77,84	77,39	78,03	81,11	63,84	**81,28**	**71,15**	74,66	63,97	75,61	64,50
BF	75,04	84,71	**75,60**	**84,85**	75,00	82,96	73,90	83,24	78,80	75,76	**78,95**	**79,81**	75,39	75,82	72,88	75,59
WP	43,55	**65,37**	44,44	65,23	44,85	63,61	**46,83**	64,20	54,44	54,29	**54,90**	**59,87**	51,22	54,10	45,24	54,28
WR	59,92	**63,18**	**60,93**	63,14	56,46	56,37	50,69	56,79	57,23	38,54	**57,73**	**47,90**	50,35	38,50	48,21	39,59
WF	50,44	**64,25**	**51,39**	64,16	49,99	59,77	48,69	60,27	55,80	45,08	**56,28**	**53,22**	50,78	44,99	46,68	45,79
LP	50,77	**60,36**	**51,17**	54,43	47,35	51,45	44,67	51,03	49,93	36,55	**50,31**	**40,76**	46,25	37,42	43,89	37,75
LR	19,66	35,96	20,15	39,11	22,51	42,95	**29,44**	**43,35**	37,35	54,56	**37,84**	52,07	36,80	**55,29**	27,38	48,33
LF	28,35	45,07	28,92	45,51	30,51	46,82	**35,49**	**46,88**	42,73	43,77	**43,19**	**45,72**	40,99	44,63	33,72	42,39
WL	3,44	3,00	3,45	3,00	3,76	3,27	4,37	3,28	4,50	4,09	4,50	3,63	4,82	4,08	4,44	3,98
TL	4,67	4,11	4,64	4,13	5,05	4,29	5,96	4,42	6,19	5,43	6,16	5,15	6,46	5,38	6,34	5,49
N_{type}	2610	1627	2654	1962	3204	2280	4441	2320	5041	4077	5069	3489	5362	4035	4204	3497
N_{token}	39323	45069	39182	45139	35979	41325	30934	41250	30042	33098	30051	37307	28092	33189	30453	34013

TABLE 2 – Résultats sur le corpus japhug pour `dpseg` et ses versions à deux niveaux supervisés par des annotations denses (`sentence`) ou un par un dictionnaire (`dictionary`). `hier.` représente le modèle `hierarchical`. 200 phrases sont utilisées pour constituer les données de supervision.

Le tableau 2 présente les résultats avec les deux types de supervision. Les colonnes `dpseg` sont obtenues en comparant la segmentation du modèle supervisé ou en mots ou en morphèmes avec le texte de référence segmenté au niveau correspondant ; il n'y a donc aucune garantie que ces deux segmentations soient compatibles entre elles. Le modèle « parallèle » échantillonnant d'abord les mots puis les morphèmes (`parallel-w`) obtient de meilleures performances en général que le modèle inverse (`parallel-m`), qui atteint un F-score supérieur seulement pour la métrique LF au niveau des mots. De plus, si les résultats avec les annotations de phrases sont proches entre la *baseline* et le modèle `parallel-w`, ce n'est pas le cas avec le dictionnaire de supervision, où `parallel-w` améliore ses scores au niveau des morphèmes. Par ailleurs, si le modèle hiérarchique ne semble pas se démarquer, là encore par rapport au modèle « parallèle », il maintient, néanmoins, une plus grande différence de longueur d'unités segmentées (environ un caractère) avec la supervision `sentence`.

Enfin, la supervision faible parvient à améliorer les scores aux deux niveaux pour les trois modèles. Si présenter des phrases segmentées manuellement améliore les F-scores aux deux niveaux, l'influence du dictionnaire est plus contrastée : une hausse significative au niveau des mots, pour une baisse de BF et WF au niveau des morphèmes. Un effet notable de la supervision par dictionnaire est de limiter les phénomènes de sur-segmentation en mots, ce qui s'accompagne ici d'une nette dégradation dans l'identification des frontières de morphèmes.

3.4 Segmentation automatique du corpus tsez

Les tableaux 3 et 4 présentant les résultats pour le corpus tsez semblent confirmer les tendances observées pour le japhug. Dans l'ensemble, le modèle `parallel-w` maintient de meilleures performances

modèle	référence		dpseg		parallel-w		hierarchical	
niveau	mot	morph.	mot	morph.	mot	morph.	mot	morph.
BF			**71,06**	75,38	70,73	**79,35**	70,47	76,21
WF			38,76	43,58	38,46	**51,42**	**40,34**	45,11
LF			25,00	45,74	25,21	**47,66**	**33,28**	46,73
WL	5,61	2,81	3,94		3,97	3,47	4,92	3,87
TL	6,93	5,21	4,53		4,51	4,31	6,17	4,67
N_{type}	5732	1603	1939		1963	1586	3091	1894
N_{token}	20153	40229	28692		28506	32620	23013	29217

TABLE 3 – Résultats sur le corpus tsez pour `dpseg` et ses versions à deux niveaux non supervisées.

supervision	sentence						dictionary					
modèle	dpseg		parallel-w		hierarchical		dpseg		parallel-w		hierarchical	
niveau	mot	morph.	mot	morph.	mot	morph.	mot	morph.	mot	morph.	mot	morph.
BF	**76,11**	79,44	76,05	**82,09**	74,73	79,24	**78,97**	72,43	78,34	**75,67**	74,25	72,59
WF	**49,04**	52,90	48,74	**57,85**	48,91	51,75	**54,17**	38,17	53,34	**43,28**	46,47	38,38
LF	37,66	52,95	37,63	**54,55**	**42,92**	53,06	**46,66**	47,97	46,47	**50,73**	41,47	47,19
WL	4,18	3,73	4,16	3,37	5,03	3,74	4,90	4,48	4,90	4,10	5,12	4,41
TL	5,03	4,56	5,02	4,39	6,19	4,70	5,84	5,39	5,87	5,10	6,46	5,41
N_{type}	2468	1902	2463	1649	3495	1880	3437	2741	3439	2442	3416	2550
N_{token}	27079	30329	27180	33533	22499	30271	23096	25263	23083	27621	22085	25666

TABLE 4 – Résultats sur le corpus tsez pour `dpseg` et ses versions à deux niveaux supervisés par des annotations denses (`sentence`) ou un dictionnaire (`dictionary`). 200 phrases sont utilisées pour constituer les données de supervision.

que le modèle hiérarchique. On notera ici que les scores LF sont meilleurs qu'en japhug, phénomène explicable par la taille moindre des données. Par ailleurs, la supervision permet de segmenter en unités de longueur plus proche de la référence : les TL s'allongent pour se rapprocher de 6,93 au niveau des mots et 5,21 au niveau des morphèmes, en particulier avec une supervision par liste de mots.

3.5 Analyse

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

FIGURE 2 – Exemple de phrase en japhug segmentée par les différents modèles, avec et sans supervision : « Atterrissez sur la queue de mon cheval ».

L'exemple de la figure 2 présente une phrase en japhug segmentée par les différents modèles présentés. Tout d'abord, sans supervision, `dpseg` joint deux unités enjambant une frontière de mot (« mbrowjme »), tout comme `parallel-w` avec « ɯkɤ ». Ce type d'erreurs, non seulement dégrade les trois niveaux de F-score, mais également crée des segments n'ayant pas de sens. De plus, les frontières de morphèmes dans la référence ne sont pas identifiées comme telles (pas de frontière pour « ɯ-jme » et une frontière de mot dans « a-mbro »).

Avec supervision, le modèle `parallel-w` parvient à rectifier son erreur initiale (« ɯkɤ » est segmenté) et à trouver des frontières de morphèmes. Les données de supervision comportaient en effet les mots « a-mbro » et « kɤ-zo », ce qui semble avoir aidé le modèle. L'erreur restante (« ɯjme ») s'explique par le fait que dans le corpus, toutes les occurrences du morphème « jme » sont toujours précédées de « ɯ- ». Le modèle n'identifie donc pas, ni ne recrée « jme » en tant qu'unité mais conserve « ɯjme ». L'effet négatif de ces co-occurrences constitue une limite inhérente au modèle `dpseg` unigramme qui est déjà observée dans (Goldwater *et al.*, 2009).

4 Conclusion

En étendant le modèle bayésien non-paramétrique de segmentation `dpseg`, nous avons pu développer deux types de modèles de segmentation en mots et morphèmes : l'un segmentant en « parallèle », l'autre de manière hiérarchique. Sur des corpus de deux langues morphologiquement complexes, nous avons constaté une meilleure performance du premier type de modèle lorsqu'il échantillonne d'abord les mots, puis les morphèmes. Par ailleurs, de manière générale, nous avons observé une amélioration légère des résultats grâce à la supervision faible (phrases annotées ou listes de mots et morphèmes), montrant que la modélisation à deux niveaux permet de mieux distinguer les types d'unités segmentées.

Remerciements

Ce travail est effectué dans le cadre du projet franco-allemand ANR-DFG « La documentation automatique des langues à l'horizon 2025 » (*Computational Language Documentation by 2025*, CLD 2025, ANR-19-CE38-0015-04). Les auteurs remercient Guillaume Jacques pour la mise à disposition des textes annotés en japhug et Antonios Anastasopoulos pour le corpus tsez.

Références

ABDULAEV A. K. & ABDULAEV I. K. (2010). *Cezjas fol'klor : (gíurus mecrek°iorno butirno) = Dido (Tsez) folklore = Didojskij (cezskij) fol'klor*. Leipzig : Lotos.

GODARD P. (2019). *Unsupervised word discovery for computational language documentation*. Theses, Université Paris-Saclay. HAL : tel-02286425.

GOLDWATER S., GRIFFITHS T. L. & JOHNSON M. (2009). A Bayesian framework for word segmentation : Exploring the effects of context. *Cognition*, **112**(1), 21–54. DOI : https ://doi.org/10.1016/j.cognition.2009.03.008.

JACQUES G. (2021). *A grammar of Japhug*. Volume 1 de Comprehensive Grammar Library. Berlin : Language Science Press. DOI : 10.5281/zenodo.4548232.

JOHNSON M., GRIFFITHS T. L. & GOLDWATER S. (2007). Adaptor Grammars : a Framework for Specifying Compositional Nonparametric Bayesian Models. In B. SCHÖLKOPF, J. PLATT & T. HOFFMAN, Éds., *Advances in Neural Information Processing Systems 19*, p. 641–648, Cambridge, MA : MIT Press.

KUDO T. & RICHARDSON J. (2018). SentencePiece : A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing : System Demonstrations*, p. 66–71, Brussels, Belgium : Association for Computational Linguistics. DOI : 10.18653/v1/D18-2012.

MOCHIHASHI D., YAMADA T. & UEDA N. (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*, p. 100–108, Suntec, Singapore : Association for Computational Linguistics.

OKABE S., BESACIER L. & YVON F. (2022). Weakly Supervised Word Segmentation for Computational Language Documentation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, Dublin, Ireland : Association for Computational Linguistics.

SMIT P., VIRPIOJA S., GRÖNROOS S.-A. & KURIMO M. (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*, p. 21–24, Gothenburg, Sweden : Association for Computational Linguistics. DOI : 10.3115/v1/E14-2006.

TEH Y. W. (2006). *A Bayesian Interpretation of Interpolated Kneser-Ney*. Rapport interne TRA2/06, School of Computing, National University of Singapore.

ZHAO X., OZAKI S., ANASTASOPOULOS A., NEUBIG G. & LEVIN L. (2020). Automatic interlinear glossing for under-resourced languages leveraging translations. In *Proceedings of the 28th International Conference on Computational Linguistics*, p. 5397–5408, Barcelona, Spain (Online) : International Committee on Computational Linguistics. DOI : 10.18653/v1/2020.coling-main.471.