Segmentation en mots faiblement supervisée pour la documentation automatique des langues

Shu Okabe, François Yvon, Laurent Besacier

Segmentation en mots faiblement supervisée pour la documentation automatique des langues

Shu Okabe[1] François Yvon[1] Laurent Besacier[2]
(1) Université Paris-Saclay, CNRS, LISN, Bât. 508, Rue du Belvédère, F-91405 Orsay, France
(2) NAVER LABS Europe, 6-8 chemin de Maupertuis, F-38240 Meylan, France
shu.okabe@limsi.fr, francois.yvon@limsi.fr,
laurent.besacier@naverlabs.com

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

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

1 Introduction

La documentation automatique des langues vise à outiller les linguistes de terrain pour faciliter l'annotation des données linguistiques. Les travaux récents se sont concentrés sur des méthodes non-supervisées. Toutefois, comme le souligne Bird (2020), des ressources auxiliaires, par exemple des listes de mots ou des textes annotés, sont souvent disponibles grâce aux efforts passés et présents de locuteurs natifs et de linguistes. L'idée de cette étude est de mobiliser ces ressources dans les algorithmes de segmentation en mots pour améliorer leurs performances.

La tâche de segmentation en mots consiste à retrouver les frontières des mots dans une séquence continue de caractères. Elle intervient notamment en documentation automatique des langues quand des enregistrements audio sont retranscrits phonétiquement. Une illustration de cette tâche est dans (Godard, 2019), qui étudie différents modèles de segmentation notamment des modèles bayésiens non paramétriques sur une langue peu dotée : le Mboshi (Bantu C25). Dans ce contexte, nous étudions comment des ressources auxiliaires peuvent aider les modèles de segmentation en mots.

2 Méthodes

2.1 Conditions expérimentales

Modèle Le modèle de référence utilisé est la version unigramme de `dpseg` (Goldwater et al., 2009), un modèle simple et bien adapté au traitement de petits corpus. Ce modèle repose sur les processus de Dirichlet pour calculer la probabilité de mots et de séquences de mots. Dans notre implémentation, l'inférence de ce modèle repose sur l'échantillonnage de Gibbs et le recuit simulé. Une variante de ce modèle, basée sur les processus de Pitman-Yor (PYP), a aussi été implémentée (Teh, 2006).

Supervision faible Deux types de ressources ont été considérées pour introduire une supervision faible, qui simulent des conditions réelles de documentation : des phrases partiellement ou complètement segmentées et des listes de mots. La première situation correspond au repérage de pauses dans les enregistrements (annotation partielle) ou à des phrases déjà segmentées (annotation complète). La seconde correspond à la pré-existence de dictionnaires ou au recueil des types observés dans les phrases segmentées. Ces listes sont utilisées pour renforcer la probabilité à priori des types connus.

Langues étudiées Deux langues peu dotées en cours de documentation ont été étudiées : le Mboshi, langue bantoue déjà présentée dans des travaux antérieurs (Godard et al., 2018), ainsi que le Japhug, langue sino-tibétaine parlée dans la partie ouest de la Chine (Jacques, 2021).

2.2 Expériences

Stratégie de supervision faible
La première expérience évalue l'impact d'une supervision faible sur les deux modèles (`dpseg` et sa variante utilisant PYP). Lorsque l'on supervise les frontières de mots, les annotations denses sont plus efficaces que des annotations partielles réparties aléatoirement dans le texte. Dans le cas de listes de mots, l'amélioration du modèle de caractères, associée à une augmentation de la probabilité des mots présents dans le dictionnaire de supervision, permettent d'obtenir les meilleurs résultats.

Le tableau 1 détaille ces résultats pour le Mboshi, avec les modèles `dpseg` et `pypseg`, sans supervision (/), avec une supervision sur les occurrences (phrases annotées) et avec une supervision sur les types (listes de mots). Les données de supervision sont dérivées d'une annotation de 200 phrases du corpus d'entraînement. Les segmentations sont évaluées avec les F-scores à trois niveaux : BF pour l'évaluation des frontières de mots, WF pour l'évaluation au niveau des occurrences et LF pour l'évaluation au niveau des types.

modèle	dpseg			pypseg		
supervision	/	token	type	/	token	type
BF	65.9	68.7	66.4	66.2	68.8	65.8
WF	37.6	42.4	39.4	37.9	42.5	38.7
LF	23.8	31.4	40.0	24.5	31.6	39.9

TABLE 1 – Résultats des segmentations non supervisées et supervisées sur le texte Mboshi

Pour les deux modèles, la supervision faible améliore la segmentation, comme en témoigne la hausse notable des scores pour toutes les métriques, à l'exception du BF dans le cas `pypseg` avec une supervision par liste de mots.

Dans le cas non supervisé, le modèle reposant sur les PYP apparaît meilleur que `dpseg`. Toutefois, dans les deux autres cas, il ne semble pas améliorer de manière significative les performances des modèles faiblement supervisés en comparaison avec leurs versions `dpseg`.

Dans l'ensemble, le modèle `dpseg` supervisé avec un dictionnaire de mots obtient le meilleur résultat.

Apprentissage incrémental
La seconde expérience étudie un scénario d'apprentissage incrémental et simule la situation où un

expert annote progressivement des phrases. Les corrections sont prises en compte pour l'annotation des phrases ultérieures (`regular`). Une variante a aussi été implémentée, dans laquelle le modèle probabiliste de base qui évalue la forme des unités lexicales dans le modèle est régulièrement mis à jour en intégrant les mots au fur et à mesure de leur vérification (`2level`). Cette seconde approche a présenté de meilleurs résultats, avec un effet stable à travers tout le texte.

Le graphique 1 présente l'évolution du taux d'erreur tout au long du texte en Japhug pour le modèle de référence ainsi que les deux modèles faiblement supervisés. Le taux d'erreur a été calculé toutes les 100 phrases, en divisant le nombre d'erreurs par la longueur de ces phrases. Le point de départ de ces trois modèles est la sortie du modèle `dpseg` complètement non supervisé. Régulièrement, le modèle effectue des itérations supplémentaires d'échantillonnage de Gibbs sur l'ensemble du texte restant afin de propager les améliorations obtenues grâce aux phrases corrigées.

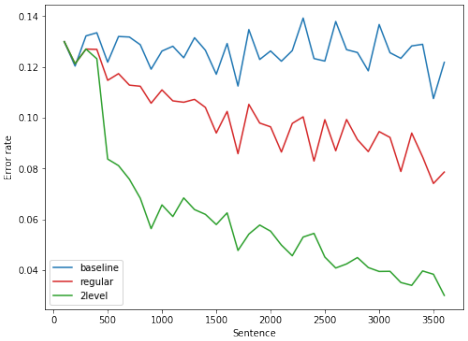

FIGURE 1 – Résultats de l'apprentissage incrémental pour le texte Japhug 3K

Le modèle de référence (en bleu) maintient un taux d'erreur moyen à peu près constant sur tout le texte. À l'inverse, les deux modèles qui bénéficient de l'apprentissage en ligne permettent une chute tendancielle du taux d'erreur ; le modèle `2level` (en vert) se montre sensiblement meilleur, grâce à son modèle lexical qui devient de plus en plus précis.

Comparaison de segmentation en mots ou en morphème
Une dernière expérience porte sur l'étude des segmentations en mots et morphèmes.

référence	mot			morphème		
supervision	/	mot	morph.	/	mot	morph.
BF	72.9	78.8	76.1	80.8	71.0	75.8
WF	45.7	55.8	51.0	54.7	39.2	45.1
LF	20.1	42.7	32.8	41.2	33.5	43.8

TABLE 2 – Comparaison des résultats sur le texte Japhug 3K pour un texte segmenté en mot ou en morphème (référence), avec ou sans supervision sur les types (supervision)

Le tableau 2 présente les résultats comparatifs pour cette expérience qui utilise comme supervision, soit un dictionnaire de mots, soit un dictionnaire de morphèmes (extrait de 200 phrases dans les deux cas). Sans supervision, les modèles comparés ont tendance à produire une segmentation en unités

courtes, proche d'une segmentation en morphèmes ; ajouter une supervision par dictionnaire de mots permet de contre-balancer cette tendance (de manière plus mitigée lors d'une supervision sur les morphèmes).

3 Conclusion

Plusieurs stratégies de supervision faible ont été étudiées pour la segmentation de mots de langues peu dotées. Dans le cadre de la documentation automatique de langues, des ressources auxiliaires, telles des phrases annotées ou des listes de mots, sont souvent disponibles et peuvent être mobilisées pour pallier la faible quantité de données. Les modèles bayésiens non paramétriques parviennent à bénéficier de ces données supplémentaires pour les deux types de ressources. L'apprentissage incrémental semble aussi se prêter à une utilisation réelle, tandis que l'expérience de segmentation à deux niveaux, mots et morphèmes, ouvre des perspectives pour de futurs travaux.

Remerciements

Ce travail est effectué dans le cadre du projet franco-allemand « La documentation automatique des langues à l'horizon 2025 » (*Computational Language Documentation by 2025*, CLD 2025, ANR-19-CE38-0015-04) . Un rapport détaillé de l'ensemble de ces expériences sera mis en ligne sur le site du projet. Les auteurs remercient Guillaume Jacques pour la mise à disposition des textes annotés en Japhug.

Références

Bird, S. (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.

Godard, P. (2019). *Unsupervised word discovery for computational language documentation*. Theses, Université Paris-Saclay.

Godard, P., Adda, G., Adda-Decker, M., Benjumea, J., Besacier, L., Cooper-Leavitt, J., Kouarata, G.-N., Lamel, L., Maynard, H., Mueller, M., Rialland, A., Stueker, S., Yvon, F., and Zanon-Boito, M. (2018). 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).

Goldwater, S., Griffiths, T. L., and Johnson, M. (2009). A Bayesian framework for word segmentation : Exploring the effects of context. *Cognition*, 112(1) :21–54.

Jacques, G. (2021). *A grammar of Japhug*. Language Science Press.

Teh, Y. W. (2006). A hierarchical Bayesian language model based on Pitman-Yor processes. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 985–992, Sydney, Australia. Association for Computational Linguistics.