Répliquer et étendre pour l'alsacien « Étiquetage en parties du discours de langues peu dotées par spécialisation des plongements lexicaux »

Alice Millour[1] Karën Fort[1, 2] Pierre Magistry[3]
(1) Sorbonne Université / STIH, 28, rue Serpente 75006 Paris, France,
(2) Université de Lorraine, CNRS, Inria, LORIA, 54000 Nancy, France
(3) Aix-Marseille Université, ENP-China, Irasia, 13100 Aix-en-Provence, France
alice.millour@sorbonne-universite.fr, karen.fort@sorbonne-universite.fr,
pierre.magistry@univ-amu.fr

RÉSUMÉ

Nous présentons ici les résultats d'un travail de réplication et d'extension pour l'alsacien d'une expérience concernant l'étiquetage en parties du discours de langues peu dotées par spécialisation des plongements lexicaux (Magistry *et al.*, 2018). Ce travail a été réalisé en étroite collaboration avec les auteurs de l'article d'origine. Cette interaction riche nous a permis de mettre au jour les éléments manquants dans la présentation de l'expérience, de les compléter, et d'étendre la recherche à la robustesse à la variation.

ABSTRACT

Replicating and extending for Alsatian : "POS tagging for low-resource languages by adapting word embeddings"

We present here the results of our efforts in replicating and extending for Alsatian an experiment concerning the POS tagging of low-resourced languages by adapting word embeddings (Magistry *et al.*, 2018). This work was performed in close collaboration with the authors of the original article. This rich interaction allowed us to identify the missing elements in the presentation of the experiment, to add them and to extend the experiment to robustness to variation.

MOTS-CLÉS : réplicabilité, étiquetage en parties du discours, langues peu dotées, variation.

KEYWORDS: replicability, POS-tagging, low-resourced languages, variation.

1 Motivations

Les avancées obtenues ces dernières années en traitement automatique des langues (TAL) grâce à l'apprentissage neuronal sont largement dépendantes de la disponibilité de très gros corpus dans les langues considérées. Or, pour de très nombreuses langues (la majorité, dites « peu dotées »), de tels corpus sont inexistants. Un article présenté à TALN 2018 (Magistry *et al.*, 2018) propose une solution partielle à ce problème pour l'étiquetage automatique en parties du discours, par spécialisation des plongements lexicaux. Les auteurs y annoncent des résultats supérieurs à l'état de l'art, notamment pour l'alsacien (0,91 d'exactitude).

Les corpus utilisés et le tagger entraîné ont été développés dans le cadre du projet RESTAURE porté

par D. Bernhard (LiLPa, Strasbourg). Notons que le développement parallèle des ressources et des outils a pu entraîner un certain nombre de difficultés liées à la publication asynchrone de chacun des éléments. Ceux-ci n'ont pas été publiés avec un suivi de versions strict alors que les corpus présentent des évolutions critiques, notamment la modification du jeu d'étiquettes au fur et à mesure.

Travaillant sur le sujet, nous avons souhaité reproduire cette expérience dans le but de tester la robustesse du modèle à la variation, un phénomène très répandu dans les langues peu dotées, en particulier les langues non standardisées.

Pour ne pas nous inscrire dans une logique concurrentielle, mais plutôt dans une dynamique de passage de flambeau permettant de construire des solutions véritablement ré-utilisables, nous avons choisi de bâtir notre recherche en collaboration avec les auteurs de l'article original. Nous détaillons ici le processus de réplication et les résultats que nous avons obtenus en étendant l'expérience à la robustesse à la variation.

2 Reproduire ou répliquer ?

La terminologie utilisée mérite qu'on s'y attarde, tant elle rend compte de la complexité de l'acte, apparemment simple, de rejouer une expérience décrite dans un article de recherche. Nous reprenons ici de manière succincte les questions mises au jour et détaillées dans (Cohen *et al.*, 2018). La réplicabilité est une propriété d'une expérience, celle d'être rejouée ou répétée [1], alors que la reproductibilité est une propriété des **résultats** de l'expérience menée : on peut obtenir les mêmes conclusions ou les mêmes valeurs [2]. Nous nous intéressons ici en priorité à répliquer l'expérience (pour mieux la comprendre), pour ensuite tenter d'en reproduire le résultat (pour être sûr de partir sur les mêmes bases) pour, enfin, étendre l'expérience.

De tels efforts sont de plus en plus valorisés dans le domaine du TAL. Après deux ateliers à LREC 2016 et 2018 (Branco *et al.*, 2016, 2018), une *shared task*, REPROLANG [3], a été organisée dans le cadre de LREC 2020. La possibilité de rejouer une expérience est même devenue un critère de sélection pour COLING 2018. Une étude menée parmi les chercheurs du domaine a montré que le sujet est perçu comme un problème important par la majorité des répondants (Mieskes *et al.*, 2019) et que, lorsque ceux-ci ont essayé de reproduire une expérience (et y sont parvenus), les résultats obtenus se sont très souvent révélés significativement différents de ceux publiés. Cela ne signifie pas pour autant que les auteurs originaux sont de mauvaise foi. Simplement, le manque de documentation des expérimentations empêche souvent de se replacer dans les conditions expérimentales de l'expérience initiale [4].

Parmi les éléments trop souvent mal documentés, les pré-traitements (dont la tokénisation) et les versions des logiciels et des ressources langagières utilisées sont des classiques (Fokkens *et al.*, 2013). L'expérience reproduite ici ne fait pas exception, malgré les efforts de ses auteurs.

1. "Replicability or repeatability is a property of an experiment : the ability to repeat –or not– the experiment described in a study." (p. 3)

2. "Reproducibility is a property of the outcomes of an experiment : arriving –or not– at the same conclusions, findings, or values." (p. 3).

3. Voir : https://lrec2020.lrec-conf.org/en/reprolang2020/.

4. Nous ne formulons pas ici d'hypothèse quant aux raisons méthodologiques ou pratiques de ce manque de documentation dans le domaine du TAL. Il nous a été signalé par un des relecteurs que dans d'autres domaines souffrant tout autant de la précarité de leurs chercheurs, la reproductibilité systématique des expériences est assurée, notamment grâce à l'utilisation de "carnets de recherche" pour leur documentation.

3 Faire tourner le code

Plutôt que de réimplémenter la solution proposée, nous avons essayé de retrouver les conditions initiales dans lesquelles l'expérience avait été menée, tant au niveau du logiciel que des ressources langagières. Dans cette section, nous présentons donc la méthodologie que nous avons souhaitée reproduire ainsi que les éléments relatifs (i) à la disponibilité du code source et (ii) à la mise en place des configurations logicielles nécessaires pour faire tourner ce code.

3.1 Méthodologie

La méthodologie proposée permet de spécialiser les plongements lexicaux à la tâche d'annotation en parties du discours en combinant l'analyse au niveau caractère et l'utilisation des propriétés morphosyntaxiques pour un mot cible et son contexte. Le système MIMICK (Pinter *et al.*, 2017), qui se base sur la graphie des mots pour calculer les vecteurs, est utilisé pour établir les plongements des mots hors vocabulaire, nombreux dans le cas de langues peu dotées et non standardisées. Cette méthodologie peut être découpée en trois étapes permettant d'obtenir des résultats intermédiaires : i) entraînement sur un corpus brut permettant de produire un fichier de plongements lexicaux dits morphosyntaxiques, ii) entraînement du modèle de *tagger* basé sur un Bi-LSTM en utilisant les plongements lexicaux et iii) évaluation du *tagger*. Il est à noter que les deux éléments intermédiaires (fichiers de plongements et modèle du Bi-LSTM entraînés pour l'alsacien) ne sont pas distribués.

3.2 Accès au code source

Le code tel qu'utilisé dans l'expérience originale n'a pas pu être retrouvé. Le co-auteur en charge des expériences ayant terminé le postdoctorat qu'il réalisait à l'époque de la publication de l'article, il n'a aujourd'hui plus accès aux machines sur lesquelles celui-ci était stocké. Nous avons néanmoins eu accès à deux versions ultérieures du code source, correspondant à deux implémentations de la méthodologie décrite dans l'article.

Le premier code source auquel nous avons eu accès, CS_1 [5] a été mis à disposition par le premier auteur de l'article. Le dépot GitHub transmis contient une réécriture partielle du code original. Cette réécriture ayant été abandonnée avant son terme, l'ensemble des étapes réalisées dans l'expérience initiale n'y sont pas représentées.

Un second dépôt, CS_2 [6], contenant le code complet en python a été identifié dans un second temps. Il s'agit de la version simplifiée et documentée du code original, réalisée et distribuée par une postdoctorante ne faisant pas partie des auteurs initiaux de l'article.

Ces deux dépôts GitHub n'étant pas renseignés dans l'article, ni associés aux noms des auteurs, ils ne pouvaient pas être identifiés sans prise de contact avec ceux-ci. Or, ces derniers sont encore précaires et leurs affiliations changent régulièrement, nous avons donc eu de la chance d'une part de parvenir à rentrer en contact avec l'un d'entre eux malgré la désactivation de sa boîte mail, et d'autre part de pouvoir accéder à la deuxième version du code.

5. Accessible ici : https://github.com/a-tsioh/MSETagger.
6. Accessible ici : https://github.com/eknyazeva/MSETagger_py.

3.3 Accès à des modèles pré-entraînés

Bien que ce soit une pratique devenue courante en TAL, aucun des codes sources diffusés ne s'accompagne de modèles pré-entraînés. Ce système comporte trois étapes produisant des modèles : les plongements lexicaux initiaux, le modèle MIMICK qui permet de les compléter et les poids du Bi-LSTM de l'étiqueteur final. Les auteurs initiaux ont fait le choix de diffuser le code permettant de reconstruire ces modèles, mais aucun des résultats intermédiaires. Chacune de ces trois étapes recourt à de l'apprentissage profond qui suppose une initialisation aléatoire de grandes matrices de poids. La stabilité de ces modèles n'est pas garantie. elle a même d'autant plus de chances d'être problématique lorsque les corpus d'entraînement sont relativement petits, comme c'est le cas ici. [7]

3.4 Configuration logicielle

Les deux dépôts GitHub sont accompagnés de `README` contenant la majorité des informations de configuration nécessaires à l'exécution du code, notamment une liste de dépendances quasi complète. Les versions de certaines librairies python sont absentes de la documentation, mais les versions compatibles entre elles des différentes librairies ont pu être déduites à tâtons.

De la même manière, l'architecture de Bi-LSTM sur laquelle s'appuie le travail des auteurs est l'implémentation YASET (Tourille *et al.*, 2017). La version de YASET utilisée n'était précisée que dans l'un des dépôts. Dans les deux cas, lorsque les hyper-paramètres n'étaient pas précisés dans l'article, ils étaient donnés dans un fichier de configuration distribué avec le code source.

Ces difficultés, associées à la méconnaissance initiale des technologies employées par les auteurs (par exemple le langage `scala`, et le moteur de production `sbt`), constituent des freins importants à la réplication de l'expérience. La mise en place de la configuration logicielle s'est donc faite en étroite collaboration avec le premier auteur de l'article d'origine.

4 Données utilisées

4.1 Généralités sur l'alsacien

L'alsacien est un terme englobant qui regroupe les langues germaniques, principalement alémaniques, parlées en Alsace et une partie de la Moselle. L'alsacien fait partie, avec les parlers alémaniques d'Allemagne et de Suisse, des langues regroupées sous le code ISO-639-3 `gsw`. Il présente des variantes à la fois dialectales et orthographiques en raison de l'absence de standard consensuel. Nous nous intéressons à la gestion en TAL de ces variantes.

Concernant les ressources brutes disponibles, la Wikipédia alémanique (code wikipédia `als`) contient des pages écrites dans 35 langues alémaniques identifiées [8]. Les pages en alsacien sont celles catégorisées `Artikel uf Elsassisch` (1 893 pages) et `Artikel uf Elsässisch` (1 page). La majorité de ces pages concernent des lieux et sont très semblables entre elles.

7. Un article plus long, décrivant le système plus en détails et détaillant ce problème était en cours de rédaction suite à l'article de TALN 2018, mais il n'a pas pu être terminé avant la fin du projet ANR.

8. Voir les catégories commençant par "`Artikel uf`", https://als.wikipedia.org/wiki/Spezial:Kategorie consultées en février 2020

4.2 Corpus bruts

Les corpus bruts utilisés pour entraîner les plongements lexicaux sont les corpus C_{Brut}_56k et C_{Brut}_200k. C_{Brut}_56k, communiqué sur demande par l'un des auteurs l'ayant lui même obtenu de D. Bernhard, est constitué d'un ensemble de 103 pages Wikipédia totalisant 56 965 tokens. Ce corpus, libre de droit, peut être reconstruit à partir de la liste des pages fournies avec le corpus. C_{Brut}_200k a été obtenu ultérieurement auprès de D. Bernhard. C'est un ensemble de documents contenant des pages de la Wikipédia alémanique rédigées en alsacien, ainsi que des documents dont les licences ne sont pas claires. La proportion de ce corpus qui est effectivement libre de droit n'a pas été déterminée.

4.3 Corpus annoté

Le corpus annoté de l'alsacien utilisé pour entraîner et évaluer le *tagger*, $C_{Annoté}$, est distribué sous licence CC BY-SA [9]. C'est un ensemble constitué de (i) pages de la Wikipédia alémanique écrites en alsacien (ii) chroniques publiées par le conseil général du département du Haut-Rhin, (iii) une recette et (iv) un extrait de pièce de théâtre, totalisant 12 644 tokens annotés (Bernhard *et al.*, 2018).

5 Résultats obtenus

Les résultats que nous obtenons diffèrent des résultats publiés précédemment. Une partie de la variation observée peut s'expliquer par la difficulté à reconstituer des corpus similaires (notamment la division en jeu d'entraînement et jeu test). Il semble aussi qu'une grande part de cette variation est à attribuer à l'instabilité des plongements lexicaux entraînés sur de petits corpus (voir Section 3.3). Ceci pose la question de l'importance de la diffusion de modèles pré-entraînés. Une telle pratique favorise la reproductibilité des résultats mais dans le même temps, elle masque des propriétés importantes de la chaîne de traitement complète.

5.1 Premières expériences, réalisées avec la réécriture partielle du code (CS_1)

La première tentative de reproduction des résultats a été réalisée à partir du code CS_1 en utilisant C_{Brut}_56k pour entraîner les plongements, 80 % de $C_{Annoté}$ pour entraîner le modèle, et 20 % $C_{Annoté}$ pour l'évaluer.

Cette expérience nous a permis d'attester que nos conditions logicielles étaient les mêmes que celles de l'auteur initial (à ce jour) : nous avons en effet mené cette expérience en parallèle et obtenu le même résultat (une exactitude du *tagger* de 0,78). Il n'y a donc pas d'élément de configuration implicite n'ayant pas été communiqué par l'auteur. En revanche, la taille du corpus C_{Brut}_56k transmis par l'auteur ne correspondant pas aux données présentées dans l'article initial, nous avons poussé nos recherches pour finalement obtenir l'accès au corpus C_{Brut}_200k.

Cette expérience a également permis de mettre au jour que soit le corpus $C_{Annoté}$ disponible à ce jour en ligne n'est pas dans l'état dans lequel les expériences initiales ont été menées, soit la réécriture du code utilisé à l'époque est incomplète et ne gère plus certains cas particuliers propres à l'alsacien et

9. Voir https://zenodo.org/record/2536041.

pris en charge avant la réécriture. Nous n'avons en effet pas pu retrouver plusieurs éléments utilisés à l'époque, tels qu'un filtre sur les corpus bruts permettant d'éliminer les entrées de dictionnaire, et une opération visant à uniformiser les jeux d'étiquettes.

Concernant le jeu d'étiquettes et la tokénisation, l'article initial ne mentionne pas les choix qui ont été faits à ce sujet. Les corpus C_{Brut} et $C_{Annoté}$ sont aujourd'hui disponibles tokénisés de deux manières différentes, et le tokéniseur distribué pour l'alsacien [10] ne gère pas les cas divergents, en l'occurrence le découpage – ou non – des contractions de prépositions (ADP) et déterminants (DET), par exemple : « *zum*/ADP+DET », découpé en « *zu*/ADP *dem*/DET ».

Nous avons réalisé une seconde expérience en utilisant C_{Brut}_200k pour entraîner les plongements, et en utilisant les mêmes corpus que précédemment après uniformisation du jeu d'étiquettes. Cette nouvelle configuration nous a permis d'obtenir un *tagger* d'une exactitude de 0,81. Un score de 0,87 a été obtenu plus tard par l'auteur de l'article après activation d'une option non spécifiée dans la documentation.

Ces diverses expériences ont donc montré que la réplicabilité du travail en question ne pouvait se faire sans que l'auteur ne complète le code mis à disposition. Par ailleurs, certaines ressources langagières, non librement disponibles, n'ont pu être retrouvées que par relations inter-personnelles. Enfin, certains traitements (en particulier la tokénisation) n'étaient pas suffisamment documentés et n'ont pas pu être reconstitués, ce qui, comme nous l'avons précisé en section 2, est un oubli classique.

5.2 Un pas plus loin : tester la robustesse à la variation avec le code CS_2

Nous avons fixé le paramètre `patience` à 75 pour toutes les expériences réalisées avec le code C_2, la réécriture en python simplifiée et documentée du code original. La première expérience réalisée, en utilisant C_{Brut}_200k pour l'entraînement des plongements, et les corpus aux jeux d'étiquettes normalisés décrits ci-dessus, nous a permis d'obtenir une exactitude de 0,89 (valeur moyenne sur un 5-fold avec un écart-type de 0,005). Ce code implémente selon nous de manière fiable la méthodologie présentée par les auteurs de l'article d'origine. Nous l'avons donc utilisé pour mener des expériences additionnelles, afin d'en tester la robustesse à la variation.

Pour ce faire, nous avons séparé le corpus annoté en deux sous-ensembles, en nous basant sur une caractéristique linguistique identifiée : la prédominance de la terminaison des noms et adjectifs en "-e" dans les variantes du nord, et en "-a" dans les variantes du sud (Brunner, 2001). Chaque sous-ensemble n'est pas uniforme et contient lui même plusieurs variantes, par exemple la variante strasbourgeoise parmi les variantes du nord.

Nous avons fait l'hypothèse qu'un article Wikipédia ne contenait qu'une seule variante. Néanmoins, lorsque le calcul des fréquences relatives des terminaisons propres à chaque variantes ne nous permettait pas de décider, nous avons examiné le fichier à la main. Dans tous les cas, nous avons pu identifier le biais à l'origine de l'équilibre des terminaisons, comme la fréquence élevée d'un élément de vocabulaire (par exemple le déterminant "***de***"), ou la présence de mots en français (par exemple, "Stad**e** d**e** l'Ill"). Nous avons ainsi pu attribuer à chaque article la variante lui correspondant. Les fréquences relatives des terminaisons en "-e" et en "-a" sont en moyenne d'un facteur 30. Nous avons ainsi pu déterminer que 40 % du corpus annoté contenait des variantes du sud ($C_{Annoté-Nord}$, 4 998 tokens) et 60 % des variantes du nord ($C_{Annoté-Sud}$, 7 646 tokens).

10. Distribué par l'équipe du projet RESTAURE, voir https://zenodo.org/record/2454993.

Les résultats présentés dans le tableau 1 ont été obtenus en découpant les deux corpus obtenus en 3 sous-corpus : corpus d'entraînement ($C80_{Annoté-X}$, 80 %), de développement (10 %), et d'évaluation ($C10_{Annoté-X}$).

	$C10_{Annoté-Nord}$	$C10_{Annoté-Sud}$
$C80_{Annoté-Nord}$	0,75	0,74
$C80_{Annoté-Sud}$	0,74	0,79

TABLE 1 – Résultats de l'entraînement sur des corpus plus uniformes quant aux variantes présentes dans les corpus d'entraînement et d'évaluation

En première analyse, et bien que les corpus d'évaluation soient de taille réduite, il semble que la méthodologie proposée soit sensible aux variantes présentes dans les corpus : les performances les meilleures sont obtenues lorsque le corpus d'entraînement et d'évaluation contiennent les mêmes variantes. Notamment, les performances du *tagger* entraîné sur le corpus $C80_{Annoté-Sud}$ diminuent de 4 points sur le corpus d'évaluation $C10_{Annoté-Nord}$. Il serait intéressant de prolonger cette étude en mesurant à taille de corpus égales pour les deux sous-variantes, l'impact de la présence - ou non - de celles-ci dans le corpus utilisé pour entraîner les plongements.

6 Conclusions et perspectives

Nous avons présenté ici une expérience de réplication collaborative d'un article publié à TALN 2018. Le logiciel conçu à l'époque n'est plus disponible en tant que tel et nous n'avons pas réussi à le reconstruire à partir des pièces accessibles au premier auteur de l'article. Nous avons appris entre temps qu'il a été repris par une autre personne (précaire également), a fait l'objet d'améliorations et est désormais disponible sur un autre dépôt GitHub [11]. Nous avons répliqué l'expérience sur cette nouvelle base (qui n'est pas non plus celle de l'article initial), mais ne sommes pas parvenus à en reproduire les résultats (0,87 avec CS_1, 0,89 avec CS_2, *vs* 0,91 dans l'article initial).

Comme évoqué en section 3.3, la question de la distribution des résultats intermédiaires (fichiers de vecteurs, modèles de *tagger*) se pose dans le cas général. Cependant, dans le contexte de langues peu dotées, la distribution de modèles instables ne paraît pas indiquée. Dans ce contexte, l'effort doit donc être dirigé en priorité vers l'accessibilité aux ressources. En effet, l'accès à ces dernières pose de nombreux problèmes : le corpus C_{Brut} à partir duquel sont entraînés les plongements dans l'expérience initiale n'est en effet pas librement disponible.

Cela pose la question de la définition d'un résultat « état de l'art ». Selon nous celle-ci devrait être précisée pour prendre en compte les cas où la ressource d'origine ne peut pas être ré-utilisée librement.

Il paraît évident que la reproductibilité d'une expérience ne peut être assurée sans une documentation précise des conditions de l'expérience au sens large mais aussi du protocole d'évaluation. Il nous semble que la pratiques à mettre en œuvre dépendent largement de l'expérience et de la méthode d'évaluation initiale, c'est pourquoi nous nous gardons de dresser ici une liste de bonnes pratiques à partir d'une seule tentative de reproduction d'expérience.

Enfin, une partie des obstacles que nous avons rencontrés est liée au cadre dans lequel la recherche est réalisée. Parmi les difficultés rencontrées figurent en effet l'instabilité des affiliations des chercheurs,

11. Voir : https://github.com/eknyazeva/MSETagger_py.

l'urgence dans laquelle les recherches sont réalisées, l'abandon des démarches de pérennisation du code, etc. Comme il a été proposé par plusieurs relecteurs, ils serait intéressant de mener une étude comparative des pratiques de documentation dans différentes disciplines souffrant tout autant de la précarité de leurs chercheurs.

Remerciements

Nous remercions A-L. Ligozat et S. Rosset (LIMSI-CNRS) ainsi que D. Bernhard (LiLPa, Strasbourg) pour leur disponibilité, leurs conseils et l'aide qu'elles nous ont apporté. Nous remercions également les relecteurs de l'atelier ETeRNAL, qui nous ont permis, grâce à leurs remarques constructives, d'améliorer notre article.

Références

BERNHARD D., LIGOZAT A.-L., MARTIN F., BRAS M., MAGISTRY P., VERGEZ-COURET M., STEIBLE L., ERHART P., HATHOUT N., HUCK D., REY C., REYNÉS P., ROSSET S., SIBILLE J. & LAVERGNE T. (2018). Corpora with Part-of-Speech Annotations for Three Regional Languages of France : Alsatian, Occitan and Picard. In *11th edition of the Language Resources and Evaluation Conference*, Miyazaki, Japon. HAL : hal-01704806.

BRANCO A., CALZOLARI N. & CHOUKRI K., Éds. (2016). *Proceedings of the Workshop on Research Results Reproducibility and Resources Citation in Science and Technology of Language*.

BRANCO A., CALZOLARY N. & CHOUKRI K., Éds. (2018). *Proceedings of the 4REAL 2018 - Workshop on Replicability and Reproducibility of Research Results in Science and Technology of Language*, Paris, France. European Language Resources Association.

BRUNNER J.-J. (2001). *L'alsacien sans peine*. Assimil.

COHEN K. B., XIA J., ZWEIGENBAUM P., CALLAHAN T., HARGRAVES O., GOSS F., IDE N., NÉVÉOL A., GROUIN C. & HUNTER L. E. (2018). Three Dimensions of Reproducibility in Natural Language Processing. In N. C. C. CHAIR), K. CHOUKRI, C. CIERI, T. DECLERCK, S. GOGGI, K. HASIDA, H. ISAHARA, B. MAEGAARD, J. MARIANI, H. MAZO, A. MORENO, J. ODIJK, S. PIPERIDIS & T. TOKUNAGA, Éds., *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

FOKKENS A., VAN ERP M., POSTMA M., PEDERSEN T., VOSSEN P. & FREIRE N. (2013). Offspring from Reproduction Problems : What Replication Failure Teaches Us. p. 1691–1701.

MAGISTRY P., LIGOZAT A.-L. & ROSSET S. (2018). Étiquetage en parties du discours de langues peu dotées par spécialisation des plongements lexicaux. In *Conférence sur le Traitement Automatique des Langues Naturelles*, Rennes, France. HAL : hal-01793092.

MIESKES M., FORT K., NÉVÉOL A., GROUIN C. & COHEN K. B. (2019). NLP Community Perspectives on Replicability. In *Recent Advances in Natural Language Processing*, Varna, Bulgarie. HAL : hal-02282794.

PINTER Y., GUTHRIE R. & EISENSTEIN J. (2017). Mimicking word embeddings using subword RNNs. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, p. 102–112, Copenhague, Danemark : Association for Computational Linguistics. DOI : 10.18653/v1/D17-1010.

TOURILLE J., FERRET O., NÉVÉOL A. & TANNIER X. (2017). Neural architecture for temporal relation extraction : A bi-LSTM approach for detecting narrative containers. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2 : Short Papers)*, p. 224–230, Vancouver, Canada : Association for Computational Linguistics. DOI : 10.18653/v1/P17-2035.