

# Des pseudo-sens pour améliorer l'extraction de synonymes à partir de plongements lexicaux

Olivier Ferret

CEA, LIST, Laboratoire Vision et Ingénierie des Contenus, Gif-sur-Yvette, F-91191 France.

olivier.ferret@cea.fr

## RÉSUMÉ

---

Au-delà des modèles destinés à construire des plongements lexicaux à partir de corpus, des méthodes de spécialisation de ces représentations selon différentes orientations ont été proposées. Une part importante d'entre elles repose sur l'utilisation de connaissances externes. Dans cet article, nous proposons *Pseudofit*, une nouvelle méthode de spécialisation de plongements lexicaux focalisée sur la similarité sémantique et opérant sans connaissances externes. *Pseudofit* s'appuie sur la notion de pseudo-sens afin d'obtenir plusieurs représentations pour un même mot et utilise cette pluralité pour rendre plus génériques les plongements initiaux. Nous illustrons l'intérêt de *Pseudofit* pour l'extraction de synonymes et nous explorons dans ce cadre différentes variantes visant à en améliorer les résultats.

## ABSTRACT

---

### **Pseudo-senses for improving the extraction of synonyms from word embeddings**

Beyond the models for building word embeddings, some methods has been proposed for specializing such representations according to a particular perspective. Most of these methods rely on external knowledge. In this article, we propose *Pseudofit*, a new method for specializing word embeddings according to semantic similarity without any external knowledge. *Pseudofit* first exploits the notion of pseudo-sens for building several representations for a word and then uses these representations for making the initial embeddings more generic. We illustrate the interest of *Pseudofit* for acquiring synonyms et we study several variants and extensions of *Pseudofit* according to this perspective.

---

**MOTS-CLÉS :** Sémantique distributionnelle, spécialisation de plongements lexicaux.

**KEYWORDS:** Distributional semantics, specialization of word embeddings.

---

## 1 Introduction

Le développement des modèles neuronaux dans le domaine du Traitement Automatique des Langues s'est accompagné d'une attention spécifique accordée aux représentations qu'ils construisent et manipulent, en particulier celles relatives aux mots, appelées également plongements lexicaux (*word embeddings*). Cette attention s'est traduite par la définition de modèles permettant de construire de telles représentations à partir de larges corpus selon des critères distributionnels, à l'instar des modèles CBOW et Skip-gram (Mikolov *et al.*, 2013) ou de GloVe (Pennington *et al.*, 2014).

La popularisation de ces méthodes s'est rapidement accompagnée de travaux visant à modifier

*a posteriori* les représentations construites afin de leur donner une orientation spécifique. Cette orientation se matérialisant le plus souvent sous la forme de relations lexicales, les plongements lexicaux sont adaptés afin que les similarités qu'ils entretiennent reflètent les relations lexicales considérées. Kiela *et al.* (2015) biaisent ainsi des plongements lexicaux vers la notion de similarité ou de proximité sémantique en s'appuyant soit sur des synonymes, soit sur des associations lexicales libres. Des méthodes telles que Retrofitting (Faruqui *et al.*, 2015), Counter-fitting (Mrkšić *et al.*, 2016) ou PARAGRAM (Wieting *et al.*, 2015) s'inscrivent dans la même perspective.

Les méthodes opérant sans connaissances externes sont le plus souvent intrinsèquement liées à la façon dont les plongements lexicaux sont construits. Levy & Goldberg (2014) orientent ainsi des plongements vers la similarité sémantique plutôt que la proximité en utilisant des dépendances syntaxiques dans les contextes plutôt que des cooccurrents graphiques. Certaines méthodes *a posteriori* existent mais n'affichent pas d'orientation spécifique autre que d'améliorer les plongements de façon générale : la méthode All-but-the-Top (Mu, 2018), axée sur la réduction de la dimensionnalité des plongements, est de celles-ci tandis que (Vulić *et al.*, 2017), en exploitant des relations morphologiques extraites à partir de règles, se classe parmi les approches pauvres en besoins de connaissances.

Dans cet article, nous proposons *Pseudofit*, une méthode d'amélioration de plongements lexicaux n'utilisant pas de connaissances externes et visant la similarité sémantique en général et l'extraction de synonymes en particulier. L'idée est d'exploiter la notion de pseudo-sens issue de la désambiguïsation sémantique pour matérialiser les variations observables dans les représentations distributionnelles des mots et les prendre en compte explicitement pour créer des représentations dépassant ces variations. Nous montrons plus particulièrement l'intérêt de *Pseudofit* dans le cadre d'une évaluation intrinsèque et pour l'extraction de synonymes (cf. section 3.1). Nous testons également différentes variantes de la méthode permettant pour certaines d'entre elles de l'améliorer (cf. section 3.2).

## 2 Méthode

### 2.1 Principes

La représentation distributionnelle d'un mot, au sens des contextes distributionnels, est par essence construite à partir d'un corpus et varie donc d'un corpus à un autre. Cette variabilité existe aussi au sein d'un corpus, même si celui-ci est assez homogène : les représentations distributionnelles des mots construites à partir de chacune des deux moitiés d'un corpus ne sont ainsi pas identiques. Du point de vue d'une modélisation générale du sens des mots et en supposant que les différents sens d'un mot sont répartis de façon homogène entre ces deux moitiés de corpus, il est en revanche naturel de considérer que ces représentations devraient être identiques, ou au moins très proches, et que les différences observées sont pour l'essentiel contingentes. Dans cette optique, une représentation issue de la convergence des représentations construites à partir de chacun des deux sous-corpus devrait avoir un caractère plus générique et donc améliorer ses propriétés du point de vue de la similarité sémantique.

Une mise en œuvre très littérale de cette approche se heurte néanmoins à une difficulté notable : la qualité des représentations distributionnelles est très directement liée à la taille des corpus dont elles sont issues. Une approche conduisant à réduire la taille de ces corpus se traduit donc mécaniquement par une dégradation de qualité que ne compense pas en pratique l'amélioration induite par la convergence des représentations. Pour dépasser ce problème, la méthode *Pseudofit* que nous

proposons s'appuie sur la notion de pseudo-sens. Cette notion est à mettre en relation avec celle de pseudo-mot, introduite en désambiguïsation sémantique par Gale *et al.* (1992) et Schütze (1992). Un pseudo-mot est un mot artificiel constitué du regroupement de deux mots différents, chacun d'entre eux constituant un pseudo-sens de ce mot artificiel. *Pseudofit* adopte la perspective inverse en divisant arbitrairement les occurrences d'un même mot  $m$ , des noms dans le cas présent, en deux sous-ensembles : les occurrences de l'un sont étiquetées  $m_1$  et les occurrences de l'autre,  $m_2$ . Chaque sous-ensemble constitue alors un pseudo-sens de ce mot. Une représentation distributionnelle est construite à la fois pour  $m$ ,  $m_1$  et  $m_2$  selon les mêmes modalités, dans le cas présent au moyen d'un modèle neuronal produisant des plongements lexicaux.

La seconde phase de *Pseudofit* consiste à modifier *a posteriori* la représentation de  $m$  afin de tenir compte de la généralisation issue du rapprochement des représentations de  $m_1$  et  $m_2$ . Considérer simultanément la représentation de  $m$  et celles de  $m_1$  et  $m_2$  permet à la fois de bénéficier de la qualité de la représentation de  $m$ , construite avec tout le corpus, et des différences des représentations de  $m_1$  et  $m_2$ . Cette modification *a posteriori* est quant à elle réalisée en appliquant une méthode de spécialisation de plongements sur la base des relations de similarité sémantique existant par principe entre  $m$ ,  $m_1$  et  $m_2$ .

## 2.2 Construction des plongements lexicaux

La première étape de la méthode *Pseudofit* consiste donc à construire une représentation distributionnelle de chaque mot du corpus considéré ainsi que des deux pseudo-sens de ce mot. Le point de départ de cette construction est la génération pour chaque occurrence d'un mot d'un ensemble de contextes distributionnels. Classiquement, ces contextes sont générés à partir d'une fenêtre graphique centrée sur le mot cible. La particularité dans le cas présent est que ces contextes sont générés à la fois pour le mot cible et pour l'un de ses deux pseudo-sens. Le pseudo-sens concerné change d'une occurrence du mot à l'autre, conduisant ainsi au même nombre d'occurrences pour les deux pseudo-sens (à 1 occurrence près). Le tableau suivant offre une illustration des contextes générés pour les noms d'une phrase avec une fenêtre de trois mots (avant et après le mot cible) :

A policeman<sub>1</sub> was arrested by another policeman<sub>2</sub>.

CIBLE	CONTEXTE	CIBLE	CONTEXTE	CIBLE	CONTEXTE
policeman	a	policeman <sub>1</sub>	a	policeman <sub>2</sub>	another
policeman	be	policeman <sub>1</sub>	be	policeman <sub>2</sub>	by
policeman	arrest (x2)	policeman <sub>1</sub>	arrest	policeman <sub>2</sub>	arrest
policeman	by (x2)	policeman <sub>1</sub>	by		
policeman	another				

Cette phrase, au caractère volontairement artificiel, permet de montrer comment, à partir d'un même corpus, trois représentations différentes du même mot sont construites : l'une, à la première colonne, est construite à partir des contextes de toutes les occurrences du mot cible ; une deuxième, à la troisième colonne, est construite à partir d'une moitié des occurrences du mot cible, représentant son premier pseudo-sens, tandis que la troisième, à la cinquième colonne, s'appuie sur l'autre moitié des occurrences du mot cible, représentant son second pseudo-sens.

Compte tenu de la seconde phase de *Pseudofit*, les représentations distributionnelles construites sont des plongements lexicaux, produits ici grâce au modèle Skip-gram. Plus précisément, il s'agit de la

variante du modèle Skip-gram proposée par Levy & Goldberg (2014) et permettant de considérer des contextes arbitraires, formés de dépendances syntaxiques dans leur cas. Il est à noter que l’approche que nous proposons ne se limite pas au modèle Skip-gram puisque le même type de variante a été défini dans (Li *et al.*, 2017) pour les modèles CBOW et GloVe par exemple.

## 2.3 Convergence des représentations d’un même mot

L’objectif de la seconde phase est de faire converger les trois représentations de chaque mot cible obtenues lors de la première phase – la représentation du mot en tant que tel et celle de ses deux pseudo-sens – en faisant l’hypothèse qu’une telle convergence permet d’en construire une représentation plus générale en gommant des différences qui sont essentiellement contingentes puisque ces trois représentations font référence par principe à une même entité.

Cette convergence est mise en œuvre grâce à l’algorithme PARAGRAM. Celui-ci prend en entrée des plongements lexicaux et un ensemble de relations binaires entre mots caractérisant une forme de similarité sémantique. Ces relations sont supposées fournies a priori et sont généralement issues de réseaux lexicaux construits manuellement. PARAGRAM modifie graduellement les plongements en entrée par descente de gradient stochastique afin de rapprocher les vecteurs des mots impliqués dans une relation donnée. Ce rapprochement s’effectue sous la contrainte de ne pas trop s’éloigner des plongements initiaux. Ce double objectif se traduit plus formellement par la minimisation de la fonction objectif suivante, dans laquelle le rapprochement des vecteurs en fonction des relations données est porté par la première somme tandis que la seconde, modulée par le paramètre  $\lambda$  exprime le terme de conservation des plongements :

$$\underbrace{\sum_{(x_1, x_2) \in \mathcal{L}_i} \max(0, \delta + \mathbf{x}_1 \mathbf{t}_1 - \mathbf{x}_1 \mathbf{x}_2) + \max(0, \delta + \mathbf{x}_2 \mathbf{t}_2 - \mathbf{x}_1 \mathbf{x}_2)}_{\text{rapprochement des plongements en fonction des relations}} + \lambda \underbrace{\sum_{\mathbf{x}_i \in V(\mathcal{L}_i)} \|\mathbf{x}_i^{init} - \mathbf{x}_i\|^2}_{\text{conservation des plongements initiaux}}$$

L’idée directrice de PARAGRAM est de rapprocher les vecteurs des mots impliqués dans une relation donnée (maximisation du terme  $\mathbf{x}_1 \mathbf{x}_2$ ) tout en les éloignant des vecteurs des mots ne faisant pas partie de cette relation (minimisation des termes  $\mathbf{x}_1 \mathbf{t}_1$  et  $\mathbf{x}_2 \mathbf{t}_2$ ). Pour chaque mot  $\mathbf{x}_i$ <sup>1</sup> d’une relation, un mot extérieur  $\mathbf{t}_i$  à la relation est ainsi sélectionné comme point de référence pour cet éloignement. L’ensemble des relations étant divisé en mini-lots  $\mathcal{L}_i$ , la recherche de cette référence externe se limite aux mots du mini-lot ( $V(\mathcal{L}_i)$ ) contenant la relation courante et maximise son pouvoir discriminant en sélectionnant le mot le plus proche, au sens de la mesure *Cosinus*, par rapport au mot considéré de la relation courante. Le paramètre  $\delta$  permet de fixer la marge entre les termes de rapprochement et d’éloignement : l’objectif est que la similarité entre les mots  $\mathbf{x}_1$  et  $\mathbf{x}_2$  soit plus importante que leur similarité avec respectivement les mots  $\mathbf{t}_1$  et  $\mathbf{t}_2$ , ceci avec une marge d’au moins  $\delta$ .

PARAGRAM est appliqué aux plongements issus de la première phase avec l’objectif de rapprocher les représentations des mots et de leurs pseudo-sens. Pour chaque mot  $m$ , trois relations de similarité sont ainsi définies et injectées dans les plongements initiaux grâce à PARAGRAM :  $(m, m_1)$ ,  $(m, m_2)$  et  $(m_1, m_2)$ . Au final, seules les représentations des mots sont utilisées et évaluées, étant de meilleure qualité puisque construites avec deux fois plus de données que celles des pseudo-sens.

1.  $\mathbf{x}_i$  est plus précisément le vecteur représentant le plongement associé à un mot mais pour faciliter la présentation, nous assimilerons mot et vecteur dans ce qui suit.

### 3 Expérimentations

Pour mettre en œuvre et évaluer la méthode *Pseudofit*, nous avons sélectionné aléatoirement au niveau des phrases une sous-partie du corpus Annotated English Gigaword (Napoles *et al.*, 2012) formant un ensemble d’1 milliard de mots. Ce corpus est composé d’articles de journaux en anglais auxquels a été appliqué le Stanford CoreNLP toolkit (Manning *et al.*, 2014). Nous exploitons ce corpus sous une forme lemmatisée. Pour la construction des plongements lexicaux, nous avons utilisé l’outil *word2vecf*, adaptation de *word2vec* issue de (Levy & Goldberg, 2014), avec des paramètres tirés de (Baroni *et al.*, 2014) : fréquence minimale=5, taille vecteurs=300, taille fenêtre=5, 10 exemples négatifs et  $10^{-5}$  pour le sous-échantillonnage des mots les plus fréquents. Pour les paramètres de PARAGRAM, nous avons adopté les valeurs préconisées dans (Vulić *et al.*, 2017), c’est-à-dire  $\delta = 0,6$  et  $\lambda = 10^{-9}$ , avec AdaGrad (Duchi *et al.*, 2011) pour l’optimisation et 50 itérations.

#### 3.1 Évaluation de *Pseudofit*

Notre première évaluation de *Pseudofit* est l’évaluation intrinsèque classiquement réalisée pour les plongements lexicaux. Elle consiste à calculer pour un ensemble de couples de mots la corrélation des rangs de Spearman  $\rho$  entre leurs similarités évaluées par un ensemble de personnes et celles calculées par la mesure *Cosinus* à partir des plongements à évaluer. Dans le cas présent, nous avons retenu des jeux de test de taille suffisante pour être significatifs en nous cantonnant aux noms. Il s’agit de SimLex-999 (Hill *et al.*, 2015), MEN (Bruni *et al.*, 2014) et MTurk-771 (Halawi *et al.*, 2012). Le tableau 1 compare les plongements produits par *Pseudofit* aux plongements initiaux ainsi qu’à ceux produits en remplaçant PARAGRAM par deux autres méthodes de spécialisation de plongements, Retrofitting et Counter-fitting. Pour les trois jeux de test, *Pseudofit* apporte une amélioration significative<sup>2</sup> par rapport à la condition initiale. En revanche, l’utilisation de Retrofitting ou de Counter-fitting pour remplacer PARAGRAM n’obtient pas une telle amélioration, voire conduit à une dégradation.

	SimLex-999	MEN	MTurk 771
INITIAL	0,495	0,783	0,656
Pseudofit	0,512	0,799	0,680
Retrofitting	0,496	0,774	0,650
Counter-fitting	0,495	0,772	0,649

TABLE 1: Évaluation intrinsèque de la méthode *Pseudofit*

La seconde évaluation que nous avons menée, objet principal de notre attention, possède un caractère plus extrinsèque. Il s’agit d’extraire les synonymes d’un mot. Pour ce faire, une mesure de similarité entre la représentation du mot cible et celle de chaque mot candidat est calculée, en l’occurrence la mesure *Cosinus*, et les candidats sont classés par ordre décroissant de leur valeur de similarité. Compte tenu de cet ordonnancement, nous avons adopté les mesures classiques en Recherche d’Information de R-précision (R-préc.), MAP (Mean Average Precision) et précision à différents rangs (P@r) pour nous comparer à notre référence, constituée par les synonymes de WordNet (Miller, 1990). Par ailleurs, nous avons adopté à la fois comme mots cibles et candidats les noms simples présents dans WordNet ayant plus de 10 occurrences dans chaque moitié de notre corpus, ce qui représente 20 813 noms.

2. La significativité statistique des différences a été évaluée par un test bilatéral de Steiger avec un seuil de significativité de 0,01, test implémenté grâce au module R *cocor* (Diedenhofen & Musch, 2015).

méthode	R-préc.	MAP	P@1	P@2	P@5	P@10
INITIAL	13,0	15,2	18,3	13,1	7,7	4,9
Pseudofit	+2,5	+3,3	+3,0	+2,5	+1,8	+1,1
Retrofitting	-0,5	-0,6	-0,6	-0,2 <sup>†</sup>	-0,3	-0,2
Counter-fitting	-0,6	-0,8	-0,6	-0,5	-0,4	-0,3

TABLE 2: Évaluation de *Pseudofit* (résultats différentiels en points / INITIAL, x100)

Le tableau 2 donne le résultat de cette évaluation pour 11 481 noms ayant des synonymes dans WordNet parmi nos 20 813 cibles. Les plongements évalués sont les mêmes que ceux du tableau 1. Là encore, nous pouvons constater que *Pseudofit* se traduit par une amélioration significative des résultats<sup>3</sup> par rapport aux plongements initiaux. Par ailleurs, le remplacement de PARAGRAM par Retrofitting ou Counter-fitting se traduit dans ce cas par une dégradation notable des résultats. On peut faire l’hypothèse que l’ajout par PARAGRAM d’un terme d’éloignement par rapport aux mots non présents dans les relations de similarité contribue à renforcer le rapprochement des termes des relations, qui correspondent dans le cas présent aux différentes représentations d’un même mot que l’on souhaite justement voir converger. Il faut enfin ajouter, ce que ne montre pas le tableau 2 par manque de place, que *Pseudofit* est particulièrement efficace pour les mots de fréquences plus faibles (ici, la moitié inférieure des mots cibles en termes de fréquence) puisque pour ceux-ci, la R-précision augmente de 5,3 points, la MAP, de 6,7 points, la P@1, de 7,0 points et la P@2, de 5,2 points.

### 3.2 Variantes de *Pseudofit*

Nous avons testé plusieurs variantes de la méthode *Pseudofit*. La première, *Pseudofit max*, concerne la stratégie de sélection des  $t_i$  dans le cadre de PARAGRAM. Les résultats du tableau 1 ont été obtenus comme ceux de (Mrkšić *et al.*, 2017) dans une configuration où seule la moitié des  $t_i$  sont choisis en maximisant la similarité avec les  $x_i$ , l’autre moitié étant choisie aléatoirement. Dans *Pseudofit max*, tous les  $t_i$  sont sélectionnés en fonction de leur similarité avec les  $x_i$ . La deuxième variante, *Pseudofit 3 pseudo-sens*, cherche quant à elle à établir si une augmentation du nombre de pseudo-sens par mot peut influencer favorablement les résultats, en commençant par étendre ce nombre à trois. La troisième, *Pseudofit contexte*, teste si l’introduction de ces pseudo-sens au niveau des contextes distributionnels des mots présente un intérêt. Dans cette configuration, des pseudo-sens ont ainsi été distingués pour tous les noms, verbes et adjectifs suffisamment fréquents<sup>4</sup>. Enfin, la dernière variante, *Pseudofit fus-\**, peut être vue elle-même comme une forme de variation de la deuxième. Elle introduit en effet une représentation supplémentaire du mot cible mais cette représentation, au lieu de prendre la forme d’un nouveau pseudo-sens, est constituée au contraire par une agrégation des représentations de ses deux pseudo-sens, ce qui reconstitue une forme de représentation globale de ce mot cible. Nous avons testé trois méthodes d’agrégation : *Pseudofit fus-addition* additionne les vecteurs des pseudo-sens dimension à dimension, *Pseudofit fus-moyenne* les moyenne tandis que *Pseudofit fus-max-pooling* en prend la valeur maximale.

Même si chaque variante se traduit par une amélioration des résultats par rapport à la méthode

3. La significativité statistique des différences a été évaluée grâce à un test de Wilcoxon pour échantillons appariés avec les notations suivantes : rien si  $p \leq 0,01$ ,  $\dagger$  si  $0,01 < p \leq 0,05$  et  $\ddagger$  si  $p > 0,05$ .

4. C’est-à-dire, les noms, verbes et adjectifs présents plus de 21 fois dans le corpus. Le seuil de 21 correspond à une fréquence minimale de 10 dans chaque moitié du corpus.

Variante	R-préc.	MAP	P@1	P@2	P@5	P@10
Pseudofit	15,5	18,5	21,3	15,6	9,5	6,0
Pseudofit max	+0,2 <sup>‡</sup>	+0,3	+0,3 <sup>†</sup>	+0,2 <sup>†</sup>	+0,1	+0,1
Pseudofit 3 pseudo-sens	+0,2 <sup>‡</sup>	+0,2	+0,4 <sup>†</sup>	+0,2 <sup>‡</sup>	+0,0 <sup>‡</sup>	+0,1 <sup>‡</sup>
Pseudofit contexte	+0,4 <sup>†</sup>	+0,3 <sup>‡</sup>	+0,5 <sup>†</sup>	+0,2 <sup>‡</sup>	+0,0 <sup>‡</sup>	+0,0 <sup>‡</sup>
Pseudofit fus-moyenne	+0,2 <sup>†</sup>	+0,3	+0,4	+0,2 <sup>†</sup>	+0,1	+0,1
Pseudofit fus-addition	+0,0 <sup>‡</sup>	+0,0	+0,2 <sup>‡</sup>	+0,1 <sup>‡</sup>	+0,1 <sup>†</sup>	+0,0 <sup>‡</sup>
Pseudofit fus-max-pooling	+0,2 <sup>‡</sup>	+0,3	+0,4	+0,2	+0,2	+0,1
Pseudofit max,fus-max-pooling	+0,4	+0,5	+0,5	+0,4	+0,2	+0,2

TABLE 3: Évaluation de variantes de *Pseudofit* (résultats différentiels en points / *Pseudofit*, x100)

*Pseudofit* de base, le tableau 3 montre que toutes n’ont pas le même intérêt. En termes à la fois du niveau d’amélioration et de significativité, *Pseudofit max* et *Pseudofit fus-max-pooling* sont les variantes les plus intéressantes, ce qui nous a conduit à les associer. Le résultat de cette combinaison, *Pseudofit max,fus-max-pooling*, obtient les meilleurs résultats, avec une différence significative par rapport à *Pseudofit* pour toutes les mesures. Parmi les variantes *Pseudofit fus-\**, *Pseudofit fus-max-pooling* et *Pseudofit fus-moyenne* sont proches et surclassent assez nettement *Pseudofit fus-addition*. Les résultats de *Pseudofit 3 pseudo-sens* montrent quant à eux que l’utilisation d’un plus grand nombre de pseudo-sens se heurte probablement à un problème de qualité de leur représentation résultant d’un nombre d’occurrences plus restreint. Ce même effet de fréquence, cette fois-ci au niveau des contextes, explique vraisemblablement l’impact très limité de l’introduction des pseudo-sens dans les contextes observé pour *Pseudofit contexte*.

## 4 Conclusion et perspectives

Dans cet article, nous avons proposé *Pseudofit*, une nouvelle méthode permettant d’améliorer des plongements lexicaux sans a priori sur leur mode de construction et sans faire appel à des connaissances externes. Les évaluations menées ont permis de montrer que cette méthode est capable d’améliorer de façon très significative des plongements lexicaux en les orientant vers une similarité sémantique propre à favoriser l’extraction de synonymes. Les évaluations menées ont en outre permis de mieux cerner l’influence de certains facteurs de la méthode et d’en proposer des extensions aboutissant à de meilleurs résultats. Dans le travail présenté, les principes sous-jacents à *Pseudofit*, en particulier la convergence de représentations différentes d’un même mot, n’ont été testés qu’au sein d’un même corpus. En conjonction avec les travaux sur la construction de méta-plongements lexicaux (Yin & Schütze, 2016), il serait intéressant de les appliquer à des représentations construites à partir de corpus différents, à l’image de (Mrkšić *et al.*, 2017) pour des langues différentes.

## Remerciements

Ce travail a été partiellement financé par l’Agence Nationale de la Recherche dans le cadre du projet ANR-17-CE23-0001 ADDICTE (Analyse distributionnelle en domaine de spécialité).

# Références

- BARONI M., DINU G. & KRUSZEWSKI G. (2014). Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors. In *52<sup>nd</sup> Annual Meeting of the Association for Computational Linguistics (ACL 2014)*, p. 238–247, Baltimore, Maryland.
- BRUNI E., TRAM N., BARONI M. *et al.* (2014). Multimodal distributional semantics. *Journal of Artificial Intelligence Research*, **49**, 1–47.
- DIEDENHOFEN B. & MUSCH J. (2015). cocor: A Comprehensive Solution for the Statistical Comparison of Correlations. *PLOS ONE*, **10**(4), 1–12.
- DUCHI J., HAZAN E. & SINGER Y. (2011). Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. *Journal of Machine Learning Research*, **12**, 2121–2159.
- FARUQUI M., DODGE J., JAUHAR S. K., DYER C., HOVY E. & SMITH N. A. (2015). Retrofitting Word Vectors to Semantic Lexicons. In *NAACL HLT 2015*, p. 1606–1615, Denver, Colorado.
- GALE W. A., CHURCH K. W. & YAROWSKY D. (1992). Work on statistical methods for word sense disambiguation. In *AAAI Fall Symposium on Probabilistic Approaches to Natural Language*, p. 54–60.
- HALAWI G., DROR G., GABRILOVICH E. & KOREN Y. (2012). Large-scale Learning of Word Relatedness with Constraints. In *18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'12)*, p. 1406–1414: ACM.
- HILL F., REICHART R. & KORHONEN A. (2015). Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, **41**(4), 665–695.
- KIELA D., HILL F. & CLARK S. (2015). Specializing Word Embeddings for Similarity or Relatedness. In *2015 Conference on Empirical Methods in Natural Language Processing (EMNLP 2015)*, p. 2044–2048, Lisbon, Portugal.
- LEVY O. & GOLDBERG Y. (2014). Dependency-Based Word Embeddings. In *52nd Annual Meeting of the Association for Computational Linguistics (ACL 2014)*, p. 302–308, Baltimore, Maryland.
- LI B., LIU T., ZHAO Z., TANG B., DROZD A., ROGERS A. & DU X. (2017). Investigating Different Syntactic Context Types and Context Representations for Learning Word Embeddings. In *2017 Conference on Empirical Methods in Natural Language Processing (EMNLP 2017)*, p. 2421–2431, Copenhagen, Denmark.
- MANNING C. D., SURDEANU M., BAUER J., FINKEL J., BETHARD S. J. & MCCLOSKEY D. (2014). The Stanford CoreNLP Natural Language Processing Toolkit. In *52nd Annual Meeting of the Association for Computational Linguistics (ACL 2014), system demonstrations*, p. 55–60.
- MIKOLOV T., CHEN K., CORRADO G. & DEAN J. (2013). Efficient estimation of word representations in vector space. In *ICLR 2013, workshop track*.
- MILLER G. A. (1990). WordNet: An On-Line Lexical Database. *International Journal of Lexicography*, **3**(4).
- MRKŠIĆ N., Ó SÉAGHDHA D., THOMSON B., GAŠIĆ M., ROJAS-BARAHONA L. M., SU P.-H., VANDYKE D., WEN T.-H. & YOUNG S. (2016). Counter-fitting Word Vectors to Linguistic Constraints. In *2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT 2016)*, p. 142–148, San Diego, California.
- MRKŠIĆ N., VULIĆ I., Ó SÉAGHDHA D., LEVIANT I., REICHART R., MILICA G., KORHONEN A. & YOUNG S. (2017). Semantic Specialization of Distributional Word Vector Spaces using



- Monolingual and Cross-Lingual Constraints. *Transactions of the Association for Computational Linguistics*, **5**, 309–324.
- MU J. (2018). All-but-the-Top: Simple and Effective Postprocessing for Word Representations. In *Sixth International Conference on Learning Representations (ICLR 2018), poster session*, Vancouver, Canada.
- NAPOLES C., GORMLEY M. R. & VAN DURME B. (2012). Annotated Gigaword. In *NAACL Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction (AKBC-WEKEX)*, p. 95–100, Montréal, Canada.
- PENNINGTON J., SOCHER R. & MANNING C. D. (2014). GloVe: Global Vectors for Word Representation. In *2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014)*, p. 1532–1543, Doha, Qatar.
- SCHÜTZE H. (1992). Dimensions of meaning. In *1992 ACM/IEEE conference on Supercomputing*, p. 787–796: IEEE Computer Society Press.
- VULIĆ I., MRKŠIĆ N., REICHART R., Ó SÉAGHDHA D., YOUNG S. & KORHONEN A. (2017). Morph-fitting: Fine-Tuning Word Vector Spaces with Simple Language-Specific Rules. In *55th Annual Meeting of the Association for Computational Linguistics (ACL 2017)*, p. 56–68, Vancouver, Canada: Association for Computational Linguistics.
- WIETING J., BANSAL M., GIMPEL K. & LIVESCU K. (2015). From Paraphrase Database to Compositional Paraphrase Model and Back. *Transactions of the Association for Computational Linguistics*, **3**, 345–358.
- YIN W. & SCHÜTZE H. (2016). Learning Word Meta-Embeddings. In *54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)*, p. 1351–1360, Berlin, Germany.

