Une évaluation approfondie de différentes méthodes de compositionalité sémantique

Antoine Bride Tim Van de Cruys Nicolas Asher
IRIT, Université Paul Sabatier, 118 route de Narbonne, F-31062 TOULOUSE CEDEX 9
[nom]@irit.fr

Résumé. Au cours des deux dernières décennies, de nombreux algorithmes ont été développés pour capturer la sémantique des mots simples en regardant leur répartition dans un grand corpus, et en comparant ces distributions dans un modèle d'espace vectoriel. En revanche, il n'est pas trivial de combiner les objets algébriques de la sémantique distributionnelle pour arriver à une dérivation d'un contenu pour des expressions complexes, composées de plusieurs mots. Notre contribution a deux buts. Le premier est d'établir une large base de comparaison pour les méthodes de composition pour le cas adjectif_nom. Cette base nous permet d'évaluer en profondeur la performance des différentes méthodes de composition. Notre second but est la proposition d'une nouvelle méthode de composition, qui est une généralisation de la méthode de Baroni & Zamparelli (2010). La performance de notre nouvelle méthode est également évaluée sur notre nouveau ensemble de test.

Abstract. In the course of the last two decades, numerous algorithms have sprouted up that successfully capture the semantics of single words by looking at their distribution in text, and comparing these distributions in a vector space model. However, it is not straightforward to construct meaning representations beyond the level of individual words – i.e. the combination of words into larger units – using distributional methods. Our contribution is twofold. First of all, we carry out a large scale evaluation, comparing different composition methods within the distributional framework for the case of adjective-noun composition, making use of a newly developed dataset. Secondly, we propose a novel method for adjective-noun composition, which is a generalization of the approach by Baroni & Zamparelli (2010). The performance of our novel method is equally evaluated on our new dataset.

Mots-clés : sémantique lexicale, sémantique distributionnelle, compositionalité.

Keywords: lexical semantics, distributional semantics, compositionality.

1 Introduction

Au cours des deux dernières décennies, il y a eu un intérêt croissant dans les méthodes dites « distributionnelles » pour la sémantique lexicale (Landauer & Dumais, 1997; Lin, 1998; Turney & Pantel, 2010). Ces méthodes sont nommées ainsi car elles se fondent sur l'hypothèse distributionnelle (Harris, 1954), qui stipule que les mots qui apparaissent dans les mêmes contextes ont tendance à être sémantiquement similaire. Dans l'esprit de cet adage, maintenant bien connu, de nombreux algorithmes ont été développés pour tenter de capturer la sémantique des mots simples en regardant leur répartition dans un grand corpus, et en comparant ces distributions dans un modèle d'espace vectoriel.

En comparaison avec les études manuelles de la sémantique formelle lexicale, cette approche apporte une couverture bien plus vaste et une analyse d'une grande masse de données empiriques. En revanche, il n'est pas trivial de combiner les objets algébriques de la sémantique distributionnelle pour arriver à une dérivation d'un contenu pour des expressions complexes, composées de plusieurs mots. *A contrario*, l'opération de l'application et des représentations qui utilisent le formalisme du λ-calcul dans la sémantique formelle nous donne des méthodes de composition générales et sophistiquées qui peuvent traiter non seulement la composition de sens dans les cas simples mais aussi des phénomènes complexes comme la coercition ou la composition avec des formules finement typées (Asher, 2011; Luo, 2010; Bassac *et al.*, 2010). Malgré des efforts pour trouver une méthode générale de composition et diverses approches proposées pour la composition des structures syntaxiques spécifiques (par exemple adjectifs et syntagmes nominaux, ou verbes transitifs et objets (Mitchell & Lapata, 2008; Coecke *et al.*, 2010; Baroni & Zamparelli, 2010)), le problème de composition demeure un défi pour

l'approche distributionnelle. De plus, la validation des méthodes de composition proposées s'est souvent faite à petite échelle (Mitchell & Lapata, 2008). Bien que ces études sur les jugements de similarité soient prometteuses et significatives, il serait intéressant d'avoir des études ayant une plus large couverture de validation. Elles nous permettraient de mieux comparer les différentes méthodes de composition proposées.

Notre contribution a deux buts. Le premier est d'établir une large base de comparaison pour les méthodes de composition pour le cas adjectif_nom. Pour cela nous avons créé un vaste ensemble de test utilisant des paires contenant une expression composée (adjectif_nom) et un nom qui doit être proche sinon identique sémantiquement de l'expression composée. Ces paires ont été extraites semi-automatiquement du Wiktionnaire français. Cette base de paires similaires nous permet d'évaluer en profondeur la performance des différentes méthodes de composition. Nous avons testé trois méthodes de composition déjà existantes, à savoir l'approche additive et multiplicative (Mitchell & Lapata, 2008), ainsi que l'approche par fonctions lexicales (Baroni & Zamparelli, 2010).

Les deux premières méthodes sont complètement générales et s'appliquent à des vecteurs que l'on peut automatiquement calculer pour les adjectifs et noms. En revanche, l'approche de Baroni et Zamparelli nécessite d'apprendre une fonction particulière associée à chaque adjectif. Notre second but est de généraliser l'approche fonctionnelle afin d'éliminer le besoin de conserver une fonction par adjectif. Pour cela nous utilisons une fonction généralisée apprise à l'aide des fonctions d'adjectifs de l'approche de Baroni et Zamparelli. Cette fonction généralisée se combine alors avec le vecteur d'un adjectif et celui d'un nom de manière entièrement générale. La performance de notre nouvelle méthode de l'approche fonctionnelle généralisée est également évaluée sur notre ensemble de test.

Nous avons organisé notre contribution de façon suivante. Nous détaillons d'abord les différents modèles de composition que nous évaluons dans notre étude, avec un rappel sur les différentes méthodes de composition existantes et puis une description de notre généralisation de l'approche fonctionnelle. Puis nous décrivons notre méthode d'évaluation et les résultats. Après une section sur les travaux connexes aux nôtres, nous concluons et nous précisons quelques pistes de travaux futurs.

2 Modèles de composition

Nous expliquons, dans cette section, quels modèles de composition ont été testés et à quoi ceux-ci correspondent. Après une bref rappel des modèles de composition simples, nous détaillons notamment la méthode des fonctions lexicales de Baroni & Zamparelli (2010) ainsi que la généralisation que nous en avons faite.

Voici les notations utilisées dans la suite. Lorsque nous décrivons un objet théorique, sans soucis de sa représentation physique par l'ordinateur, nous utilisons la police de base. Quand nous discutons de vecteurs, ceux-ci sont écrits en **gras**. Les matrices sont représentées en **MAJUSCULES GRASSES**. Enfin, nous écrivons les tenseurs [1] d'ordre 3 avec une majuscule calligraphiée, par exemple $\mathscr{A}$. De plus, comme nous ne manipulons pas de tenseur d'ordre supérieur à 4, nous appelons simplement les tenseurs d'ordre 3 « tenseurs » [2]. Pour conclure, le coefficient d'indice i d'un vecteur $\mathbf{v}$ est noté $\mathbf{v}_i$; la notation des coefficients des matrices et tenseurs se fait de manière analogue.

Dans la suite de cet article les adjectifs seront représentés par la lettre « a » et les noms par la lettre « n ».

2.1 Modèles de composition simples

Trois modèles de composition que nous avons utilisés sont simples à décrire : les méthodes triviale, additive et multiplicative. L'approche triviale, notée C_t et que nous utilisons comme base de comparaison, ignore l'adjectif :

$$C_t(a,n) = \mathbf{n}$$

Le modèle additif, noté C_a, consiste à réaliser une combinaison linéaire entre les vecteurs $\mathbf{a}$ et $\mathbf{n}$ à l'aide de coefficients indépendants de ceux-ci :

$$C_a(a,n) = \alpha\,\mathbf{n} + \beta\,\mathbf{a}$$

1. Un tenseur est la généralisation d'une matrice à plusieurs indices. Pour une introduction sur les tenseurs, regardez Kolda & Bader (2009).
2. les tenseurs d'ordre 1 étant les vecteurs et les tenseurs d'ordre 2 les matrices.

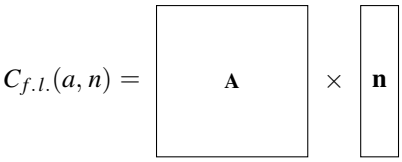


FIGURE 1: Composition dans l'approche par fonctions lexicales

Enfin, tandis que les deux rois faisaient chanter des *Te Deum*, chacun dans son camp, [Candide] prit le parti d'aller raisonner ailleurs des effets et des causes. Il passa par-dessus des tas de morts et de mourants, et gagna d'abord un **village** voisin ; il était en cendres : c'était un **village** abare que les Bulgares avaient brûlé, selon les lois du droit public. [...]
Candide s'enfuit au plus vite dans un autre **village** : il appartenait à des Bulgares, et les héros abares l'avaient traité de même...

FIGURE 2: extrait de *Candide* de Voltaire, Chapitre 3

Sur un ensemble de développement, nous avons testé le modèle pour différentes valeurs de α et β telles que $\alpha+\beta=1$ [3] et conservé les valeurs donnant les meilleurs résultats : $\alpha=0.4$ et $\beta=0.6$.

Le modèle multiplicatif, noté C_m, consiste à multiplier les vecteurs **a** et **n** terme à terme :

$$C_m(a,n) = \mathbf{n} \otimes \mathbf{a}$$
$$\text{où} \quad (\mathbf{n} \otimes \mathbf{a})_i = \mathbf{n}_i \times \mathbf{a}_i$$

L'approche par fonctions lexicales de Baroni & Zamparelli (2010) étant plus complexe, nous la décrivons dans la section suivante. Nous expliquons ensuite pourquoi et comment nous avons tenté de généraliser cette approche.

2.2 Fonctions Lexicales

Le modèle de composition par fonctions lexicales, noté $C_{f.l.}$, consiste à représenter les adjectifs par des matrices. Ainsi la combinaison d'un adjectif et d'un nom est le produit de la matrice **A** et du vecteur **n** comme le montre la figure 1.

L'approche distributionelle ne permet cependant pas de générer naturellement des matrices. Baroni et Zamparelli proposent donc d'apprendre la matrice d'un adjectif à partir d'exemples de vecteurs nom_adjectif obtenus directement à partir du corpus. De tels vecteurs nom_adjectif sont obtenus de la même manières que des vecteurs représentant un seul mot : quand la combinaison de l'adjectif et du nom occurre, on observe son contexte. Prenons l'exemple du paragraphe en figure 2. Le mot « village » apparaît trois fois. La première occurrence peut contribuer à créer le vecteur **village_voisin**, la deuxième à créer **village_abare**, et la dernière à créer **village_autre**.

Une fois que l'on a créé suffisamment de vecteurs nom_adjectif pour un adjectif donné, on calcule la matrice **A**. Pour cela, on réalise une régression partielle des moindres carrés, sur les combinaisons nom_adjectif. Formellement, en notant $\mathbf{n_a}$ les vecteurs nom_adjectif, il s'agit de trouver **A** minimisant :

$$\sum_{\mathbf{n}} \| \mathbf{A} \times \mathbf{n} - \mathbf{n_a} \|_2 \qquad \text{où} \quad \| \mathbf{v} \|_2 = \sqrt{\sum_i \mathbf{v}_i^2}$$

Pour reprendre l'exemple précédent, on minimiserait, notamment, $\|\mathbf{VOISIN} \times \mathbf{village} - \mathbf{village\_voisin}\|_2$ pour obtenir la matrice **VOISIN**.

Il est important de noter qu'une telle approche nécessite un corpus plus important que les autres approches. En effet, comme il ne s'agit plus seulement d'observer des exemples d'utilisation d'adjectifs ou de noms isolés mais des exemples d'utilisation de la combinaison d'un adjectif et d'un nom, les occurrences sont intrinsèquement plus rares. Dans le paragraphe en figure 2, chacune des apparitions du mot « village » peut contribuer à la création du vecteur **village** mais aucune ne peut contribuer à la création du vecteur **village_félon**.

3. Les vecteurs étant normalisés (*cf.* 3.2), cette condition ne réduit pas la généralité de notre test.

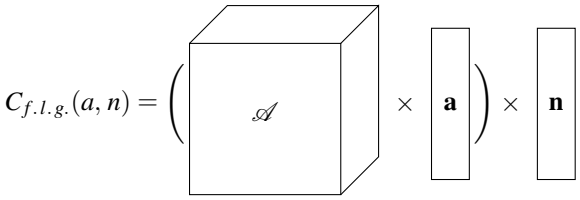


FIGURE 3: Composition dans l'approche par fonction lexicale généralisée

Baroni & Zamparelli (2010) expliquent comment limiter les problèmes liés au manque d'exemples. De plus, les expériences présentées jusqu'à maintenant montrent que les corpus actuels permettent une implémentation efficace de l'approche par fonctions lexicales pour les adjectifs les plus courants. En effet, celle-ci a obtenu les meilleurs résultats sur un certain nombre d'expériences.

Néanmoins, l'approche de Baroni et Zamparelli reste limitée pour traiter des adjectifs relativement rares. Par exemple, l'adjectif « félon » apparaît 217 fois dans le corpus FRWaC (Baroni *et al.*, 2009). C'est assez pour générer un vecteur **félon**, mais très peu pour espérer générer un nombre suffisant de vecteurs **nom_félon** et donc générer la matrice **FÉLON**.

De plus, devoir apprendre une matrice pour chaque adjectif pose un problème théorique. En effet, cette approche suppose, comme l'approche de Montague, que l'effet d'un adjectif sur un nom est idiosyncratique à l'adjectif (Kamp, 1975). Mais le désavantage de ceci est que les données montrent que la plupart des adjectifs dans les langues du monde sont subjectifs et se comportent selon des principes générales de composition (Partee, 2010; Asher, 2011). La manière dont les adjectifs sont utilisés dans la langue française laisse supposer qu'il existe une façon générale de combiner adjectifs et noms. Lorsque l'on connaît la signification d'un adjectif, l'association à un nom est rarement problématique. Ceci, indépendamment de la présence ou de l'absence d'exemples d'association.

2.3 Généralisation

Pour résoudre ces problèmes, nous proposons de généraliser les fonctions lexicales que sont les matrices d'adjectifs par une fonction lexicale unique : le tenseur de composition adjectivale $\mathscr{A}$. Dans notre approche, notée $C_{f.l.g.}$, la combinaison d'un adjectif et d'un nom est le produit du tenseur $\mathscr{A}$ avec le vecteur adjectif puis le vecteur nom, *c.f.* figure 3.

On peut noter que le produit du tenseur $\mathscr{A}$ et du vecteur **a** est une matrice dépendante de l'adjectif et multipliée au vecteur **n**. Cette matrice correspond à la matrice **A** de l'approche par fonctions lexicales de Baroni et Zamparelli. Ainsi, comme le montre la figure 4, nous obtenons le tenseur $\mathscr{A}$ à l'aide d'exemples de matrices obtenues par la méthode de Baroni et Zamparelli, et de vecteurs obtenus naturellement dans l'approche distributionelle. Plus précisément nous effectuons une régression partielle des moindres carrés sur les matrices générées par les équations. Formellement, il s'agit de trouver $\mathscr{A}$ minimisant :

$$\sum_a \| \mathscr{A} \times \mathbf{a} - \mathbf{A} \|_2 \qquad \text{où } \| \mathbf{M} \|_2 = \sqrt{\sum_{i,j} \mathbf{M}_{i,j}^2}$$

Cette équation ressemble beaucoup à l'équation non généralisée. En effet, dans les deux cas, l'objectif formel est de trouver une application linéaire [4] minimisant des équations dans un espace vectoriel de dimension finie.

Cependant, la généralisation fait une hypothèse bien plus forte. En effet, l'image d'une application linéaire est toujours de dimension inférieure à son espace de départ. Or, par construction, si les vecteurs **adjectif** existent dans un espace de dimension N, alors les matrices **ADJECTIF** existent dans un espace de dimension $N \times N$. Ainsi, s'il existe un tenseur $\mathscr{A}$, cela signifie que le sous-espace engendré par les matrices **ADJECTIF** est de taille inférieur à N réduisant considérablement leur degré de liberté maximal initial (dimension $N \times N$). L'approche de Baroni et Zamparelli n'a pas cette hypothèse puisque les vecteurs **nom** et **nom_adjectif** coexistent dans un même espace (et les espaces engendrés par ceux-ci ont donc la même dimension maximale).

Moins formellement, chercher une telle application linéaire présuppose qu'il n'y pas plus d'« information » dans le sous-espace d'arrivée que de le sous-espace de départ. Cela créé une différence notable entre les deux méthodes. En effet, dans la méthode de Baroni et Zamparelli, les vecteurs **nom** et **nom_adjectif** existent dans le même espace. L'hypothèse sus-cité

4. représenté par une matrice ou un tenseur.

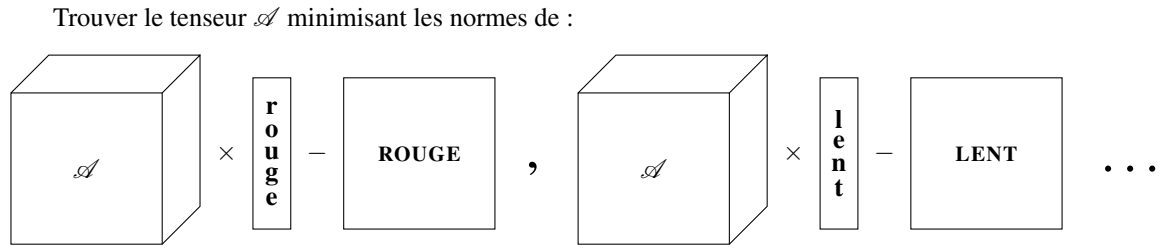


FIGURE 4: Apprentissage du tenseur $\mathscr{A}$

consiste donc à supposer que l'on a pas besoin de plus d'informations pour décrire **voisin_village** que pour décrire **village** ; uniquement d'informations différentes. Cette hypothèse est partagée par beaucoup de méthodes de composition dont l'objectif est de pouvoir réaliser des compositions en cascade [5]. *A contrario*, dans la méthode généralisée, l'application linéaire recherchée a un espace de départ [6] bien plus petit que son espace d'arrivée [7]. La généralisation fait donc une hypothèse beaucoup plus forte : elle suppose que les matrices **A** créées par la méthode de Baroni et Zamparelli ne sont pas plus informatives que les vecteurs **a** que l'on peut extraire directement d'un *corpus*. Ces matrices ne seraient, d'une certaine manière, qu'une réécriture des adjectifs pertinente pour la composition.

Ceci étant, de manière similaire à l'approche de Baroni et Zamparelli, notre approche nécessite d'apprendre un nombre significatif de matrices **A**. Cela n'est pas un problème, car le FRWaC fournit suffisamment d'adjectifs sur lesquels l'approche de Baroni et Zamparelli fonctionne parfaitement. Par exemple, le 2000$^{\text{ième}}$ adjectif le plus courant dans le FRWaC (« fasciste ») y occurre plus de 4000 fois.

Pour reprendre l'exemple de l'adjectif « félon », notre approche exige uniquement de connaître le vecteur **félon**, évitant le problème de manque de données lié à la construction de la matrice FÉLON.

Une fois le tenseur $\mathscr{A}$ obtenu, il nous fallait vérifier expérimentalement sa pertinence. En effet, nous n'avions pas garantie que le tenseur optimisant les équations décrites dans la figure 4 soit intéressant sémantiquement.

3 Évaluation

3.1 Description de la tâche

Pour évaluer les différents modèles de composition, nous avons construit une tâche de similarité inspirée des travaux de Zanzotto *et al.* (2010) et utilisée pour la tâche *evaluating phrasal semantics* de SEMEVAL-2013 (Korkontzelos *et al.*, 2013). La tâche propose de juger la similarité entre une combinaison adjectif_nom et un seul nom. Ceci est important, étant donné que les modèles de composition doivent être capables de traiter des combinaisons adjectifs_nom de taille arbitraire. La tâche est donc la suivante :

Soient **comb** = Combinaison(adjectif, nom1) et **nom2**
Évaluer Similarité(**comb**, **nom2**)

La « Combinaison » est réalisé par les différents modèles de composition. La « Similarité » doit être une fonction binaire ; les valeurs de retour étant « similaire » et « non_similaire ». Cependant, l'approche distributionnelle ne fournit naturellement que des valeurs de similarité continues (*e.g.* cosinus entre deux vecteurs). Nous avons donc pris des exemples positifs et des exemples négatifs de notre ensemble de test afin de savoir quelles valeurs de cosinus correspondent à « similaire » et quelles valeurs de cosinus correspondent à « non_similaire ». Plus précisément, nous avons, pour chaque approche, réalisé une régression logistique sur 50 exemples positifs et 50 exemples négatifs (dorénavant séparés de notre ensemble de test) afin d'apprendre le seuil de cosinus à partir duquel une paire est similaire.

Nous avons créé notre ensemble de test d'une manière semi-automatique, en utilisant des dictionnaires. Prenons par exemple la définition de *champagne* dans le Wiktionnaire français [8], figure 5. D'une telle définition, il est assez simple

5. Par exemple, obtenir le sens de « grosse voiture rouge » en composant « grosse » et « voiture » puis « rouge » et « grosse voiture ».
6. l'espace des vecteurs **mot**.
7. l'espace des matrices **ADJECTIF**.
8. `http://fr.wiktionary.org/wiki/champagne`, accédé à 20 février 2014.

d'extraire la paire (mousseux_vin, champagne). En traversant un grand dictionnaire, il est ainsi possible d'extraire des paires (`adjectif_nom`, `nom`) positives (similaires).

champagne /ʃɑ̃.paɲ/ masculin

1. Vin mousseux produit en Champagne et protégé par une appellation d'origine contrôlée.
2. *(Histoire des techniques)* Cercle de fer pour soutenir l'étoffe à teindre dans la cuve de teinture.

FIGURE 5: Définition de *champagne*, extrait du Wiktionnaire français

Nous avons donc téléchargé toutes les entrées du Wiktionnaire français, et nous les avons tagées avec le tageur MElt (Denis *et al.*, 2010). Ensuite, nous avons sélectionné les définitions qui débutent avec une combinaison adjectif-nom. Enfin, nous avons supprimé les instances utilisant des mots qui apparaissent trop peu fréquemment dans notre corpus FRWaC [9].

Les instances ainsi extraites d'une manière automatique étaient alors contrôlées manuellement. Toutes les paires jugées incorrectes étaient rejetées. Nous avons ainsi obtenu 714 exemples positifs.

Nous avons ensuite créé un premier fichier d'exemples négatifs en sélectionnant deux noms (`nom1`, `nom2`) et un adjectif `adjectif` aléatoirement. Les couples (`adjectif_nom1`, `nom2`) ainsi créés étaient ensuite vérifiés manuellement. Nous avons ainsi obtenu 899 exemples négatifs.

L'inconvénient d'un tel procédé est qu'il propose souvent des combinaisons `adjectif_nom1` insensées. Ceci simplifie la tâche de séparer exemples positifs et négatifs. Nous avons donc créé un second fichier d'exemples négatifs en sélectionnant des combinaisons `adjectif_nom1` depuis le Wiktionnaire et des noms `nom2` aléatoires. Nous avons ensuite vérifié manuellement que les couples (`adjectif_nom1`, `nom2`) ainsi créés était bien des exemples négatifs. Nous avons ainsi obtenu 494 exemples négatifs.

La table 1 montre 5 exemples positifs et 5 exemples négatifs de chaque sorte. Dans cette table, les noms et adjectifs sont sous forme de lemme. On peut noter que les exemples négatifs générés complètement aléatoirement contiennent des combinaisons `adjectif_nom` ayant un sens clair (penchant_autoritaire) et n'en n'ayant pas (chasse_fossile). Les exemples négatives créés à base du Wiktionnaire, en revanche, contiennent uniquement des combinaisons qui ont un sens clair. [10]

exemples positifs	exemples négatifs aléatoires	exemples négatifs Wiktionnaire
(mot_court, abréviation)	(importance_fortuit, gamme)	(jugement_favorable, discorde)
(ouvrage_littéraire, essai)	(penchant_autoritaire, ile)	(circonscription_administratif, fumier)
(compagnie_honorifique, ordre)	(auspice_aviaire, ponton)	(mention_honorable, renne)
(costume_féminin, ensemble)	(banquette_celeste, discipline)	(attitude_hautain, racine)
(partie_unitaire, élément)	(chasse_fossile, propulsion)	(examen_attentif, condamnation)

TABLE 1: Une partie des ensembles de test.

3.2 Espace sémantique

Une fois le test choisi et les fichiers de test réalisés nous avons créé l'espace sémantique. Pour cela, nous avons utilisé le corpus FRWaC (Baroni *et al.*, 2009) – un corpus de 1,6 milliard de mots extrait du web – tagé avec le tageur MElt (Denis *et al.*, 2010) et parsé à l'aide du parseur MaltParser (Nivre *et al.*, 2006), formé sur une version de dépendances du *French treebank* (Candito *et al.*, 2010). Nous avons d'abord récupéré les lemmes des mots, adjectifs, et noms du corpus. Nous avons uniquement conservé les lemmes écrits en toutes lettres [11] puis selectionné les 10000 lemmes les plus fréquents pour chaque catégorie (mots, adjectifs, noms). Enfin, nous avons généré l'espace en utilisant les adjectifs et les noms

9. *i.e.* moins de 200 fois pour les adjectifs et moins de 1500 fois pour les noms.
10. Nous fournissons les fichiers correspondant sur simple demande par e-mail.
11. Cette étape élimine principalement les dates, les nombres en chiffre, et la ponctuation. Nous estimons que ceux-ci ont un intérêt limité en approche distributionnelle.

comme vecteurs, et les mots comme dimensions en utilisant la méthode des *bags of words*. Nous avons alors nettoyé l'espace ainsi créés en normalisant les vecteurs et en appliquant le *positive point-wise mutual information* (*ppmi*, (Church & Hanks, 1990)) à l'espace.

Nous avons alors comparé les méthodes sur trois versions de l'espace : l'espace entier, l'espace réduit à 300 dimensions par la méthode de décomposition en valeurs singulières (*svd*, (Golub & Van Loan, 1996)), et l'espace réduit à 300 dimensions par la méthode de factorisation en matrices positives (*nmf*, (Lee & Seung, 2000)). Nous avons fait cela pour pouvoir tester chaque méthode dans des conditions optimales. En effet :

– Un espace non réduit contient plus d'informations. Ainsi les méthodes compatibles (additive et multiplicative) peuvent obtenir de meilleurs résultats. Cependant, utiliser la méthode des fonctions lexicales sur l'espace non réduit demanderait d'apprendre des matrices de taille 10000×10000. Ceci poserait des problèmes de temps de calcul et de parcimonie des données comme on a vu ci-dessus. De même pour les fonctions étendues.
– Un espace réduit avec la méthode *svd* permet expérimentalement d'obtenir de bon résultats pour les fonctions lexicales. Cependant, la présence de valeurs négatives dans le vecteurs de l'espace réduit drastiquement l'efficacité de l'approche multiplicative.
– Un espace réduit avec la méthode *nmf* ne pénalise pas les approches multiplicatives.

3.3 Résultats

Les espaces sémantiques ayant été créés, nous avons d'abord testé les différentes approches sur le jeu de test utilisant des exemples négatifs complètement aléatoires (deuxième colonne de la table 1). Nous présentons les résultats dans la table 2a. Plusieurs commentaires peuvent être faits. Nous commentons d'abord les méthodes individuellement, puis nous les comparons.

	triviale	multiplicative	additive	fonctions lexicales	f. l. généralisées
non-réduit	0.83	0.86	0.88	N/A	N/A
svd	0.79	0.55	0.84	**0.93**	0.61
nmf	0.78	0.83	0.79	0.90	0.88

(a) Les exemples négatifs sont créés entièrement aléatoirement.

	triviale	multiplicative	additive	fonctions lexicales	f. l. généralisées
non-réduit	0.78	0.79	**0.83**	N/A	N/A
svd	0.77	0.55	0.82	**0.84**	0.46
nmf	0.75	0.73	0.79	0.78	0.78

(b) Les exemples négatifs sont créés à l'aide de combinaisons adjectif-nom existantes.

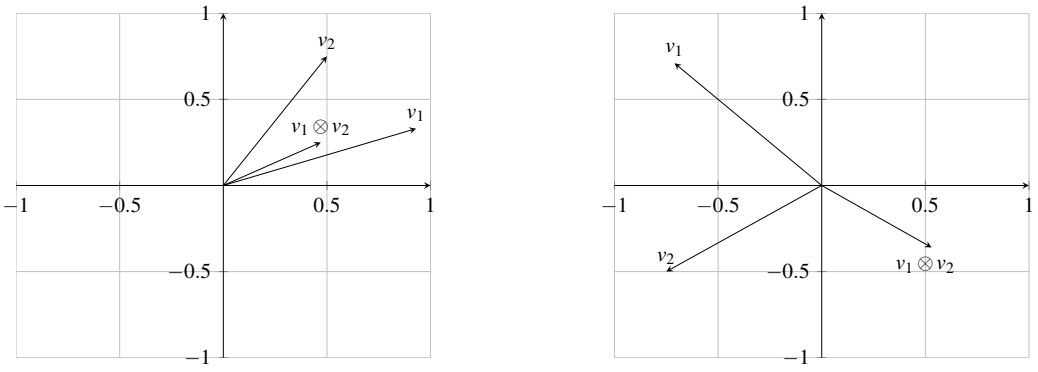
TABLE 2: Pourcentage de couples $(\text{adjectif\_nom1}, \text{nom2})$ bien classés selon l'approche et l'espace.

D'abord, l'approche triviale, consistant à comparer les deux noms et ignorer l'adjectif, affiche un taux de réussite relativement élevé ($\sim 80\%$). Ceci est dû au fait que la plupart des adjectifs ne changent pas la nature du nom auquel ils sont accolés. Une voiture rouge, lente, grosse, ou ancienne reste fondamentalement une voiture. Une voiture miniature n'est plus nécessairement une voiture mais de tels exemples sont rares.

Ensuite, la méthode multiplicative a de mauvaises performances sur l'espace réduit à l'aide de *svd*. Cela confirme l'incompatibilité de cette méthode avec les valeurs négatives générées par *svd*. La figure 6 permet de visualiser la raison à cela. On peut y voir que multiplier terme à terme deux vecteurs ayant des valeurs négatives résulte en un troisième vecteur très éloigné des deux autres. Cela va à l'encontre de l'idée selon laquelle la combinaison d'un nom et d'un adjectif à un sens proche du nom d'origine.

De plus, nous constatons que le modèle multiplicatif sur l'espace non-réduit n'atteint pas des résultats sensiblement meilleurs que le modèle trivial. La différence entre le modèle multiplicatif (0.86) et le modèle trivial (0.83) n'est pas statistiquement significative ($\chi^2 = 2.69$, $p > 0.05$). [12] Le modèle additif, en revanche, atteint un résultat en mode non-réduit (0.88) qui est significativement meilleur que la méthode triviale ($\chi^2 = 24.83$, $p < 0.01$) et le modèle multiplicatif

12. Pour tous nos tests de signification, nous utilisons le test de McNemar (Dietterich, 1998).



(a) vecteurs à valeurs positives

(b) vecteurs à valeurs négatives

FIGURE 6: l'effet de valeurs négatives sur l'approche multiplicative

($\chi^2 = 21.33$, $p < 0.01$). Les résultats du modèle additif pour les espaces *svd* et *nmf* sont également significatifs ($\chi^2 = 11.82$, $p < 0.01$ et $\chi^2 = 18.91$, $p < 0.01$, respectivement) mais il sont inférieurs au résultat de l'espace non-réduit. On constate que le modèle multiplicatif atteint un résultat de 0.83 dans l'espace *nmf* qui est significativement meilleur que le modèle multiplicatif ($\chi^2 = 31.34$, $p < 0.01$), mais toujours inférieur au résultat de l'espace non-réduit.

Ensuite, nous constatons que l'approche par fonctions lexicales de Baroni et Zamparelli dans l'espace *svd* obtient des résultats qui sont significativement meilleurs que toute autre approche avec tout autre espace ($\chi^2 = 33.49$, $p < 0.01$ pour la différence avec le modèle additif non-réduit). Nous constatons également que la fonction lexicale généralisée dans l'espace *nmf* obtient des résultats qui sont comparables avec l'approche de Baroni et Zamparelli dans ce même espace ($\chi^2 = 3.95$, $p > 0.01$) et équivalents aux meilleurs résultats des autres méthodes (notamment le modèle additif non-réduit). Cependant, la fonction lexicale généralisée – comme le modèle multiplicatif – a de faibles performances sur l'espace *svd* (0.61). Cela semble signifier que, dans cet espace, les matrices de la méthode des fonctions lexicales ne sont pas générables à l'aide d'un tenseur unique et des vecteurs correspondants.

Nous avons ensuite répété nos tests avec des exemples négatifs utilisant les combinaisons adjectif_nom extraites du Wiktionnaire français (troisième colonne de la table 1). Nous présentons les résultats dans la table 2b. Nous constatons que les résultats de nos premiers tests sont largement confirmés. Le modèle additif en espace non-réduit atteint un score qui est significativement meilleur que la méthode triviale (0.83 vs. 0.78, $\chi^2 = 10.69$, $p < 0.01$), bien que le modèle multiplicatif ne donne pas un score supérieur à la méthode triviale. Nous notons, toutefois, que l'approche par fonctions et l'approche additive obtiennent désormais des résultats globalement équivalents dans leurs meilleurs conditions respectives — 0.83 pour le modèle additif non-réduit vs. 0.84 pour le modèle fonctionel *svd*, une différence non-significative ($\chi^2 = 0.20$, $p > 0.05$). Cela semble indiquer que les fonctions lexicales sont particulièrement efficaces pour séparer les combinaisons insensées, mais qu'ils obtiennent un score inférieur quand ils doivent juger la similarité de compositions réelles.

4 Travaux connexes

Un certain nombre de chercheurs a déjà étudié et évalué divers modèles de composition au sein d'un cadre distributionnel. Une des premières tentatives pour évaluer de manière systématique des modèles simples de composition a été faite par Mitchell & Lapata (2008). Ils explorent un certain nombre de modèles différents pour la composition de vecteurs, dont les plus importants sont le modèle additif et le modèle multiplicatif. Ils évaluent leurs modèles sur une tâche de similitude de phrases nom-verbe. Pour évaluer leur modèle, ils ont demandé à des annotateurs humains de juger la similarité entre deux paires compositionnelles (par exemple en attribuant un certain score). La tâche du modèle de composition est alors de reproduire les jugements humains. Les résultats montrent que le modèle multiplicatif ainsi qu'une combinaison pondérée du modèle additif et du modèle multiplicatif donnent les meilleurs résultats. Les auteurs ont refait leur étude dans Mitchell & Lapata (2010) avec un ensemble de test plus large (les paires d'adjectifs et noms étaient également inclues), et ils ont confirmé leur résultats initiaux. Bien qu'une telle tâche de similitude a ses mérites, l'attribution d'un score de similitude est

plutôt difficile pour des juges humains [13]. Une décision binaire, comme dans notre tâche, est beaucoup moins floue. Nous soutenons que l'approche adoptée dans notre contribution donne une image plus claire et plus stable de la performance des différents modèles de composition.

Baroni & Zamparelli (2010) évaluent leur modèle de fonctions lexicales dans un contexte quelque peu différent. Ils évaluent leur modèle en regardant la capacité de reconstruire les vecteurs **nom_adjectif** qui n'ont pas été vus pendant la phase d'entraînement. Leur résultats montrent que leur modèle de fonctions lexicales atteint les meilleurs résultats pour reconstruire les vecteurs de co-occurrence originaux, suivi de près par le modèle additif. Notez que nous observons la même tendance dans notre évaluation.

Grefenstette *et al.* (2013) proposent aussi une généralisation du modèle de fonctions lexicales par des tenseurs. Leur généralisation a pour objectif différent, à savoir modéliser les verbes transitifs à l'aide de tenseurs. Cependant, nous utilisons un approche très similaire pour l'obtention des tenseurs. En effet, ils utilisent la méthode de Baroni et Zamparelli pour apprendre des matrices correspondant à une combinaison VERBE_COMPLÉMENT que l'on peut multiplier à un vecteur **sujet**, pour obtenir le vecteur **sujet_verbe_complément**. Par exemple MANGER_VIANDE multiplié au vecteur **chien** permet d'obtenir **chien_manger_viande**. Ils apprennent alors un tenseur correspondant à chaque verbe de la même manière que nous apprenons le tenseur $\mathscr{A}$.

Coecke *et al.* (2010) présentent un cadre théorique abstrait dans lequel un vecteur de phrase est une fonction du produit de Kronecker de ses vecteurs de mots, ce qui permet une plus grande interaction entre les différentes traits de mots. Un certain nombre d'instanciations du modèle de Coecke *et al.* (2010) – où l'idée clé est que les mots relationnels (par exemple les verbes) ont une structure riche (multidimensionnelle) qui agit comme un filtre sur leurs arguments – sont testés expérimentalement dans les articles de Grefenstette & Sadrzadeh (2011a) et Grefenstette & Sadrzadeh (2011b). Les auteurs évaluent leurs modèles en utilisant une tâche de similitude semblable à celle de Mitchell & Lapata. Cependant, ils utilisent des constructions compositionnelles plus étendues : plutôt que d'utiliser des compositions de deux mots (par exemple *verbe* et *objet*), ils utilisent des phrases simples transitives (*sujet verbe objet*). Ils montrent que leurs instanciations du modèle catégoriel obtiennent des meilleurs résultats que les modèles additifs et multiplicatifs sur leur tâche de similitude transitive.

Socher *et al.* (2012) présentent un modèle compositionnel basé sur les réseaux de neurones récursifs. Chaque nœud dans un arbre syntaxique est attribué à la fois un vecteur et une matrice ; le vecteur capture la signification réelle du constituant, tandis que la matrice modélise la manière dont il change le sens des mots et expressions voisins. L'évaluation s'est faite extrinsèquement, en utilisant le modèle dans une tâche de prédiction du sentiment. Ils montrent que l'approche basée sur les réseaux de neurones obtient de meilleurs résultats que les modèles additifs, multiplicatifs, et par fonctions lexicales. Cependant, d'autres chercheurs ont rapporté des résultats différents. Blacoe & Lapata (2012) évaluent les modèles additifs et multiplicatifs ainsi que l'approche de Socher *et al.* (2012) sur deux tâches différentes : la tâche de similitude de Mitchell & Lapata (2010) et une tâche de détection de paraphrases. Ils trouvent que les modèles additifs et multiplicatifs atteignent des meilleurs scores que le modèle de Socher *et al.* (2012).

Étroitement liée aux travaux sur la compositionnalité est la recherche sur le calcul de sens du mot en contexte. Erk & Padó (2008, 2009) font usage de restrictions sélectionnelles pour exprimer le sens d'un mot dans son contexte ; le sens d'un mot en présence d'un argument est calculé en multipliant le vecteur du mot avec un vecteur qui capture les restrictions sélectionnelles inverses de l'argument. Thater *et al.* (2009, 2010) étendent l'approche fondée sur les restrictions sélectionnelles en incorporant des co-occurrences du deuxième ordre dans leur modèle. Dinu & Lapata (2010) proposent un cadre probabiliste qui modélise la signification des mots comme une distribution de probabilité sur des facteurs latents. Cela permet de modéliser le sens contextualisé comme un changement dans la distribution du mot originel. Dinu et Lapata utilisent la factorisation de matrice positive (NMF) pour induire les facteurs latents.

En général, les modèles latents se sont avérées utiles pour la modélisation du sens des mots. L'un des modèles latents de la sémantique les plus connus est l'analyse de sémantique latente (LSA, Landauer & Dumais (1997)), qui utilise la décomposition en valeurs singulières afin d'induire automatiquement des facteurs latents de matrices terme-document. Un autre modèle de sens latent bien connu, qui adopte une approche générative, est l'allocation Dirichlet latente (LDA, Blei *et al.* (2003)).

Les tenseurs ont été utilisés auparavant pour la modélisation du langage naturel. Giesbrecht (2010) décrit un modèle de factorisation de tenseurs pour la construction d'un modèle distributionnel qui est sensible à l'ordre des mots. Et Van de Cruys (2010) utilise un modèle de factorisation de tenseurs afin de construire un modèle de restrictions sélectionnelles

13. En témoignent le faible taux d'accord d'inter-annotateur – Mitchell & Lapata (2010) rapportent une corrélation entre juges humains assez faible de 0.52 pour les combinaisons adjectif_nom.

multidimensionnelles de verbes, sujets et objets.

5 Conclusion

Dans notre contribution, nous avons testé différentes méthodes principales de compositionalité en approche distributionelle. À notre connaissance, nous sommes les premiers à réaliser de tels tests sur la langue française. Nous avons, de plus, créé un nouveau ensemble de test pour l'évaluation de la compositionalité dans un cadre distributionnel pour la langue française, librement disponible pour d'autres chercheurs.

Nos tests confirment que la méthode des fonctions lexicales de Baroni et Zamparelli a de bonnes performances en comparaison des autres approches. Nos tests semblent nuancer ceci par le fait que ces performances ne sont sensiblement meilleures que lorsque les exemples négatifs sont entièrement aléatoire.

De plus, nous avons proposé une généralisation de la méthode des fonctions lexicales. D'après nos tests, cette généralisation ne peut pas se faire dans les conditions optimales pour la méthode des fonctions lexicales. Ainsi bien que notre généralisation fonctionne correctement, les conditions dans lesquelles elle est utilisée font qu'elle a des résultats équivalent aux méthodes additive et multiplicative de Mitchell et Lapata, mais légèrement inférieurs à ceux de l'approche de Baroni et Zamparelli.

Dans le futur, il serait intéressant de tester différentes valeurs de réduction dimensionnelle afin d'optimiser notre fonction lexicale généralisée. De plus, il est possible que de meilleurs résultats puissent être obtenus en proposant plusieurs fonctions généralisées plutôt qu'une. On peut tenter, par exemple, de séparer les adjectifs intersectifs [14] des adjectifs non-intersectifs [15].

Il serait également intéressant de réaliser un ensemble de test pour une tâche avec laquelle la méthode des fonctions lexicales n'est pas entièrement compatible, comme la combinaison de noms. En effet, pour obtenir le sens de « laboratoire d'analyses médicales », il faut appliquer « analyses médicales » à « laboratoire ». Or la méthode des fonctions lexicales ne propose pas de manière satisfaisante d'obtenir la matrice ANALYSE_MÉDICALE. En effet, obtenir une telle matrice par apprentissage à partir d'exemples d'utilisation d'« analyse médicale » est en contradiction avec le principe de compositionalité.

Remerciements

Nous tenons à remercier toute l'équipe du projet composes [16] pour leur boite à outils DisSeCT (Dinu *et al.* (2013)) qui nous a sûrement épargné plusieurs mois de développement.

Références

ASHER N. (2011). *Lexical Meaning in Context : A Web of Words*. Cambridge University Press.

BARONI M., BERNARDINI S., FERRARESI A. & ZANCHETTA E. (2009). The wacky wide web : A collection of very large linguistically processed web-crawled corpora. *Language Resources and Evaluation*, **43**(3), 209–226.

BARONI M. & ZAMPARELLI R. (2010). Nouns are vectors, adjectives are matrices : Representing adjective-noun constructions in semantic space. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, p. 1183–1193, Cambridge, MA : Association for Computational Linguistics.

BASSAC C., MERY B. & RETORÉ C. (2010). Towards a Type-theoretical account of lexical semantics. *Journal of Logic, Language and Information*, **19**(2), 229–245.

BLACOE W. & LAPATA M. (2012). A comparison of vector-based representations for semantic composition. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, p. 546–556, Jeju Island, Korea : Association for Computational Linguistics.

14. « Rouge » par exemple. Une voiture rouge est une voiture.
15. « Faux » par exemple. Une fausse voiture n'est pas une voiture.
16. `http://clic.cimec.unitn.it/composes/`

BLEI D. M., NG A. Y. & JORDAN M. I. (2003). Latent dirichlet allocation. *The Journal of Machine Learning Research*, **3**, 993–1022.

CANDITO M., CRABBÉ B., DENIS P. *et al.* (2010). Statistical french dependency parsing : treebank conversion and first results. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC 2010)*, p. 1840–1847.

CHURCH K. W. & HANKS P. (1990). Word association norms, mutual information & lexicography. *Computational Linguistics*, **16**(1), 22–29.

COECKE B., SADRZADEH M. & CLARK S. (2010). Mathematical foundations for a compositional distributed model of meaning. *Lambek Festschrift, Linguistic Analysis, vol. 36*, **36**.

DENIS P., SAGOT B. *et al.* (2010). Exploitation d'une ressource lexicale pour la construction d'un étiqueteur morpho-syntaxique état-de-l'art du français. In *Traitement Automatique des Langues Naturelles : TALN 2010*.

DIETTERICH T. G. (1998). Approximate statistical tests for comparing supervised classification learning algorithms. *Neural computation*, **10**(7), 1895–1923.

DINU G. & LAPATA M. (2010). Measuring distributional similarity in context. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, p. 1162–1172, Cambridge, MA.

DINU G., PHAM N. & M. B. (2013). Dissect : Distributional semantics composition toolkit. In *Proceedings of the System Demonstrations of ACL*, p. 31–36, East Stroudsburg PA : Association for Computational Linguistics.

ERK K. & PADÓ S. (2008). A structured vector space model for word meaning in context. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, p. 897–906, Waikiki, Hawaii, USA.

ERK K. & PADÓ S. (2009). Paraphrase assessment in structured vector space : Exploring parameters and datasets. In *Proceedings of the Workshop on Geometrical Models of Natural Language Semantics*, p. 57–65, Athens, Greece.

GIESBRECHT E. (2010). Towards a matrix-based distributional model of meaning. In *Proceedings of the NAACL HLT 2010 Student Research Workshop*, p. 23–28 : Association for Computational Linguistics.

GOLUB G. H. & VAN LOAN C. F. (1996). *Matrix Computations (3rd Ed.)*. Baltimore, MD, USA : Johns Hopkins University Press.

GREFENSTETTE E., DINU G., ZHANG Y.-Z., SADRZADEH M. & M. B. (2013). Multi-step regression learning for compositional distributional semantics. In *Proceedings of the 10th International Conference on Computational Semantics (IWCS)*, p. 131–142, East Stroudsburg PA : Association for Computational Linguistics.

GREFENSTETTE E. & SADRZADEH M. (2011a). Experimental support for a categorical compositional distributional model of meaning. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, p. 1394–1404, Edinburgh, Scotland, UK. : Association for Computational Linguistics.

GREFENSTETTE E. & SADRZADEH M. (2011b). Experimenting with transitive verbs in a discocat. In *Proceedings of the GEMS 2011 Workshop on GEometrical Models of Natural Language Semantics*, p. 62–66, Edinburgh, UK : Association for Computational Linguistics.

HARRIS Z. S. (1954). Distributional structure. *Word*, **10**(23), 146–162.

KAMP H. (1975). Two theories about adjectives. *Formal semantics of natural language*, p. 123–155.

KOLDA T. G. & BADER B. W. (2009). Tensor decompositions and applications. *SIAM Review*, **51**(3), 455–500.

KORKONTZELOS I., ZESCH T., ZANZOTTO F. M. & BIEMANN C. (2013). Semeval-2013 task 5 : Evaluating phrasal semantics. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2 : Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, p. 39–47, Atlanta, Georgia, USA : Association for Computational Linguistics.

LANDAUER T. & DUMAIS S. (1997). A solution to Plato's problem : The Latent Semantic Analysis theory of the acquisition, induction, and representation of knowledge. *Psychology Review*, **104**, 211–240.

LEE D. D. & SEUNG H. S. (2000). Algorithms for non-negative matrix factorization. In *Advances in Neural Information Processing Systems 13*, p. 556–562.

LIN D. (1998). Automatic retrieval and clustering of similar words. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics (COLING-ACL98), Volume 2*, p. 768–774, Montreal, Quebec, Canada.

LUO Z. (2010). Type-theoretical semantics with coercive subtyping. *SALT20, Vancouver*.

MITCHELL J. & LAPATA M. (2008). Vector-based models of semantic composition. *proceedings of ACL-08 : HLT*, p. 236–244.

MITCHELL J. & LAPATA M. (2010). Composition in distributional models of semantics. *Cognitive Science*, **34**(8), 1388–1429.

NIVRE J., HALL J. & NILSSON J. (2006). Maltparser : A data-driven parser-generator for dependency parsing. In *Proceedings of LREC-2006*, p. 2216–2219, Genoa, Italy.

PARTEE B. H. (2010). Privative adjectives : subsective plus coercion. *BÄUERLE, R. et ZIM-MERMANN, TE, éditeurs : Presuppositions and Discourse : Essays Offered to Hans Kamp*, p. 273–285.

SOCHER R., HUVAL B., MANNING C. D. & NG A. Y. (2012). Semantic compositionality through recursive matrix-vector spaces. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, p. 1201–1211, Jeju Island, Korea : Association for Computational Linguistics.

THATER S., DINU G. & PINKAL M. (2009). Ranking paraphrases in context. In *Proceedings of the 2009 Workshop on Applied Textual Inference*, p. 44–47, Suntec, Singapore.

THATER S., FÜRSTENAU H. & PINKAL M. (2010). Contextualizing semantic representations using syntactically enriched vector models. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, p. 948–957, Uppsala, Sweden.

TURNEY P. & PANTEL P. (2010). From frequency to meaning : Vector space models of semantics. *Journal of artificial intelligence research*, **37**(1), 141–188.

VAN DE CRUYS T. (2010). A non-negative tensor factorization model for selectional preference induction. *Natural Language Engineering*, **16**(4), 417–437.

ZANZOTTO F. M., KORKONTZELOS I., FALLUCCHI F. & MANANDHAR S. (2010). Estimating linear models for compositional distributional semantics. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, p. 1263–1271, Beijing, China : Coling 2010 Organizing Committee.