Improving Semantic Composition with Offset Inference

Anonymous ACL submission

Abstract

Distributional semantic models based on counting co-occurrences suffer from sparsity due to unobserved but plausible co-occurrences in any text collection. This problem is amplified for models like Anchored Packed Dependency Trees (APTs), that take the type of a co-occurrence between two lexemes into account. We therefore introduce a novel form of distributional inference that leverages the rich type structure in APTs to overcome the sparsity issue for semantic composition.

1 Introduction

Anchored Packed Dependency Trees (APTs) is a recently proposed approach to distributional semantics that takes distributional composition to be a process of lexeme contextualisation (Weir et al., 2016). A lexeme's meaning, characterised as knowledge concerning co-occurrences involving that lexeme, is represented with a higher-order dependency-typed structure (the APT) where paths associated with higher-order dependencies connect vertices associated with weighted lexeme multisets. The central innovation in the compositional theory is that the APT's type structure enables the precise alignment of the semantic representation of each of the lexemes being composed. Like other count-based distributional spaces, however, it is prone to considerable data sparsity, caused by not observing all plausible co-occurrences in the given data. Recently, Kober et al. (2016) introduced a simple unsupervised algorithm to infer missing co-occurrence information by leveraging the distributional neighbourhood and ease the sparsity effect in count-based models. In this paper, we generalise distributional inference (DI) in APTs and consider a novel variant that exploits a key aspect, in particular the ability of APTs to align the semantics of different lexemes by creating "offset representations", allowing us to infer co-occurrences from neighbours of these representations. We show that our algorithm is able to more effectively exploit the rich structure inherent in APTs and close the performance gap between neural network and count-based approaches, while retaining the interpretability of our model[1].

2 Offset Representations

The basis of how composition is modelled in the APT framework is the way that the co-occurrences are structured. In characterising the distributional semantics of some lexeme w, rather than just recording a co-occurrence between w and w' within some context window, we follow Padó and Lapata (2007) and record the dependency path from w to w'. This type information imposes a structure on the semantic space which makes it possible to precisely align the semantic representations of each of the lexemes being composed in some phrase. For example many nouns will have distributional features starting with the type `amod`, which cannot be observed for adjectives or verbs. Thus, when composing the adjective *white* with the noun *clothes*, the feature spaces of the two lexemes need to be aligned first. This can be achieved by offsetting one of the constituents, which we will explain in more detail in this section.

We will make use of the following notation throughout this work. A typed distributional feature consists of a path and a lexeme such as in `amod`:*white*. Inverse paths are denoted by a horizontal bar above the dependency relation such as in $\overline{\texttt{dobj}}$:*prefer* and higher-

[1] We release our code and data at https://gitlab.com/user/repo

order paths are separated by a dot such as in $\overline{\texttt{amod}}.\overline{\texttt{compound}}$:*dress*.

Offset representations are the central component in the composition process in the APT framework. Figure 1 shows the APT representations for the adjective *white* (left) and the APT for the noun *clothes* (right), as might have been observed in a text collection. Each node holds a multiset of lexemes and the anchor of an APT reflects the current perspective of a lexeme at the given node. An offset representation can be created by shifting the anchor along a given path. For example the lexeme *white* is at the same node as other adjectives such as *black* and *clean*, whereas nouns such as *shoes* or *noise* are typically reached via the $\overline{\texttt{amod}}$ edge.

Offsetting in APTs only involves a change in the anchor, the underlying structure remains unchanged. By offsetting the lexeme *white* by `amod` the anchor is shifted along the $\overline{\texttt{amod}}$ edge, which results in creating a noun view for the adjective *white*. We denote the offset view of a lexeme for a given path by superscripting the offset path, for example the `amod` offset of the adjective *white* is denoted as *white*$^{\texttt{amod}}$. The offsetting procedure changes the starting points of the paths as visible in Figure 1 between the anchors for *white* and *white*$^{\texttt{amod}}$, since paths always begin at the anchor. The red dashed line in Figure 1 reflects that anchor shift. The lexeme *white*$^{\texttt{amod}}$ represents a prototypical "white thing", that is, a noun that has been modified by the adjective *white*. We note that all edges in the APT space are bi-directional as exemplified in the coloured `amod` and $\overline{\texttt{amod}}$ edges in the APT for *white*, however for brevity we only show uni-directional edges in Figure 1.

By considering the APT representations for the lexemes *white* and *clothes* in Figure 1, it becomes apparent that lexemes with different parts of speech are located in different areas of the semantic space. If we want to compose the adjective-noun phrase *white clothes*, we need to offset one of the two constituents to align the feature spaces in order to leverage their distributional commonalities. This can be achieved by either creating a noun offset view of *white*, by shifting the anchor along the $\overline{\texttt{amod}}$ edge, or by creating an adjective offset representation of *clothes* by shifting its anchor along `amod`. In this work we follow Weir et al. (2016) and always offset the dependent in a given relation. Table 1 shows a subset of the features of Figure 1 as would be represented in a vectorised APT. Vectorising the whole APT lexicon results in a very high-dimensional and sparse typed distributional space. The features for *white*$^{\texttt{amod}}$ (middle column) highlight the change in feature space caused by offsetting the adjective *white*. The features of the offset view *white*$^{\texttt{amod}}$, are now aligned with the noun *clothes* such that the two can be composed. Composition can be performed by either selecting the *union* or *intersection* of the aligned features.

white	***white*$^{\texttt{amod}}$**	***clothes***
:*clean*	`amod`:*clean*	`amod`:*wet*
$\overline{\texttt{amod}}$:*shoes*	:*shoes*	:*dress*
$\overline{\texttt{amod}}.\overline{\texttt{dobj}}$:*wear*	$\overline{\texttt{dobj}}$:*wear*	$\overline{\texttt{dobj}}$:*wear*
$\overline{\texttt{amod}}.\overline{\texttt{nsubj}}$.*earn*	$\overline{\texttt{nsubj}}$:*earn*	$\overline{\texttt{nsubj}}$:*admit*

Table 1: Sample of vectorised features for the APTs shown in Figure 1. Offsetting *white* by `amod` creates an offset view, *white*$^{\texttt{amod}}$, representing a noun, and has the consequence of aligning the feature space with *clothes*.

2.1 Qualitative Analysis of Offset Representations

Any offset view of a lexeme is behaviourally identical to a "normal" lexeme. It has an associated part of speech, a distributional representation which locates it in semantic space, and we can find neighbours for it in the same way that we find neighbours for any other lexeme. In this way, a single APT data structure is able to provide many different views of any given lexeme. These views reflect the different ways in which the lexeme is used. For example *law*$^{\overline{\texttt{nsubj}}}$ is the $\overline{\texttt{nsubj}}$ offset representation of the noun *law*. This lexeme is a verb and represents an action carried out by the *law*. This contrasts with *law*$^{\overline{\texttt{dobj}}}$, which is the $\overline{\texttt{dobj}}$ offset representation of the noun *law*. It is also a verb, however represents actions done to the *law*. Table 2 lists the 10 nearest neighbours for a number of lexemes, offset by `amod`, $\overline{\texttt{dobj}}$ and $\overline{\texttt{nsubj}}$ respectively.

For example, the neighbourhood of the lexeme *ancient* in Table 2 shows that the offset view for *ancient*$^{\texttt{amod}}$ is a prototypical representation of an "ancient thing", with neighbours easily associated with the property *ancient*. Furthermore, Table 2 illustrates that nearest neighbours of offset views are often other offset representations. This means that for example actions carried out by a *mother* tend to be similar to actions carried out by a *father* or *parent*.

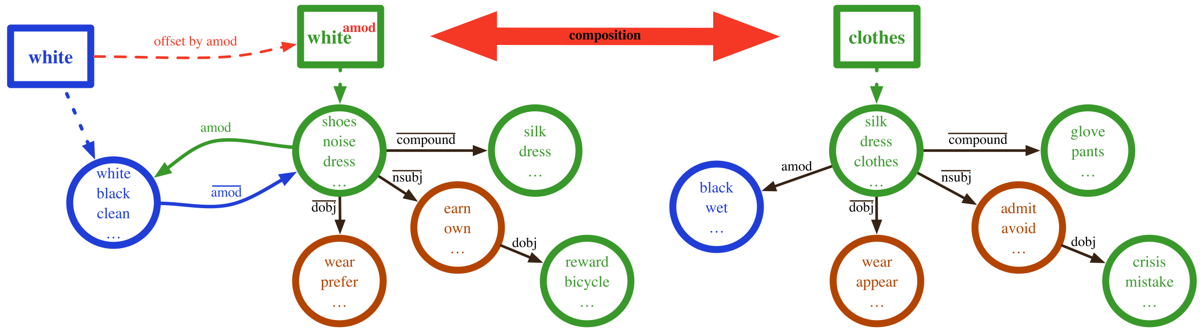

Figure 1: Structured distributional APT space. Different colours reflect different parts of speech. Boxes denote the current anchor of the APT, circles represent nodes in the APT space, holding lexemes, and edges represent their relationship within the space.

Offset Representation	Nearest Neighbours
ancient$^{\texttt{amod}}$	civilzation, mythology, tradition, ruin, monument, trackway, tomb, antiquity, folklore, deity
red$^{\texttt{amod}}$	blue$^{\texttt{amod}}$, black$^{\texttt{amod}}$, green$^{\texttt{amod}}$, dark$^{\texttt{amod}}$, onion, pepper, red, tomato, carrot, garlic
government$^{\overline{\texttt{dobj}}}$	overthrow, party$^{\overline{\texttt{dobj}}}$, authority$^{\overline{\texttt{dobj}}}$, leader$^{\overline{\texttt{dobj}}}$, capital$^{\overline{\texttt{dobj}}}$, force$^{\overline{\texttt{dobj}}}$, state$^{\overline{\texttt{dobj}}}$, official$^{\overline{\texttt{dobj}}}$, minister$^{\overline{\texttt{dobj}}}$, oust
law$^{\overline{\texttt{dobj}}}$	violate, rule$^{\overline{\texttt{dobj}}}$, enact, repeal, principle$^{\overline{\texttt{dobj}}}$, unmake, enforce, policy$^{\overline{\texttt{dobj}}}$, obey, flout
mother$^{\overline{\texttt{nsubj}}}$	wife$^{\overline{\texttt{nsubj}}}$, father$^{\overline{\texttt{nsubj}}}$, parent$^{\overline{\texttt{nsubj}}}$, woman$^{\overline{\texttt{nsubj}}}$, re-married, remarry, girl$^{\overline{\texttt{nsubj}}}$,breastfeed, family$^{\overline{\texttt{nsubj}}}$, disown
law$^{\overline{\texttt{nsubj}}}$	rule$^{\overline{\texttt{nsubj}}}$, principle$^{\overline{\texttt{nsubj}}}$, policy$^{\overline{\texttt{nsubj}}}$, criminalize, case$^{\overline{\texttt{nsubj}}}$, contract$^{\overline{\texttt{nsubj}}}$, prohibit, proscribe, enjoin, charge$^{\overline{\texttt{nsubj}}}$

Table 2: List of the 10 nearest neighbours of $\texttt{amod}$, $\overline{\texttt{dobj}}$ and $\overline{\texttt{nsubj}}$ offset representations.

2.2 Offset Inference

Our approach generalises the unsupervised algorithm proposed by Kober et al. (2016), henceforth "standard DI", as a method for inferring missing knowledge into an APT representation. The central novelty in our algorithm is that we infer neighbours of offset representations, such that *white*$^{\texttt{amod}}$ is enriched with other nouns similar to it.

A sketch of how offset inference for a lexeme w works is shown in Algorithm 1. Our algorithm requires a distributional model M, a vectorised encoding of the APT for the lexeme w on which to perform distributional inference, a dependency path p, describing the offset for w, and the number of neighbours α. The offset representation of w' is then enriched with the information from its distributional neighbours and scaled by α. We note that if the offset path p is the empty path, we would recover the algorithm presented by Kober et al. (2016). Our algorithm is unsupervised, and agnostic to the input distributional model and the neighbour retrieval function.

Algorithm 1 Offset Inference

```
procedure OFFSET_INFERENCE(M, w, p, α)
    w′ ← offset(w, p)
    w″ ← w′ × α
    for all n in neighbours(M, w′, α) do
        w″ ← w″ + n
    end for
    return w″
end procedure
```

3 Experiments

For our experiments we re-implemented the standard DI method of Kober et al. (2016) for a direct comparison. We built an order 2 APT space on the basis of the concatenation of ukWaC, Wackypedia and the BNC (Baroni et al., 2009), pre-parsed with the Malt parser (Nivre et al., 2006). We PPMI transformed the raw co-occurrence counts, using a negative SPPMI shift of $\log 5$ (Levy and Goldberg, 2014b). We evaluate our offset inference algorithm on two popular short phrase composition benchmarks by Mitchell and Lapata (2008) and Mitchell and Lapata (2010), henceforth ML08 and ML10 respectively. The ML08 dataset consists of 120 distinct verb-object (VO) pairs and the ML10 dataset contains 108 adjective-noun (AN), 108 noun-noun (NN) and 108 verb-object pairs. The goal is to compare a model's similarity estimates to human provided judgements. For both tasks, each phrase pair has been rated by multiple human annotators on a scale between 1 and 7, where 7 indicates maximum similarity. Comparison with human judgements is achieved by calculating Spearman's ρ between the model's similarity estimates and the scores of each human annotator individually. We performed composition by *intersection* and tuned the number of neighbours by a grid search over $\{0, 10, 30, 50, 100, 500, 1000\}$ on the ML10 development set, selecting 10 neighbours for NNs, 100 for ANs and 50 for VOs for both DI algorithms. We calculate statistical significance using the method of Steiger (1980).

Table 3 shows that both forms of distributional inference significantly outperform a baseline without DI. On average, offset inference outperforms the method of Kober et al. (2016) by a statistically significant margin on both datasets.

	ML10				ML08
APT configuration	**AN**	**NN**	**VO**	**Avg**	**VO**
None	0.35	0.50	0.39	0.41	0.22
Standard DI	$0.48^{\ddagger}$	0.51	$0.43^{\ddagger}$	$0.47^{\ddagger}$	$0.29^{\ddagger}$
Offset Inference	$\mathbf{0.49}^{\ddagger}$	**0.52**	$\mathbf{0.44}^{\ddagger}$	$\mathbf{0.48}^{*\ddagger}$	$\mathbf{0.31}^{\dagger\ddagger}$

Table 3: Comparison of DI algorithms. ‡ denotes statistical significance at $p < 0.01$ in comparison to the method without DI, * denotes statistical significance at $p < 0.01$ in comparison to standard DI and † denotes statistical significance at $p < 0.05$ in comparison to standard DI.

Table 4 shows that offset inference substantially outperforms comparable sparse models by Dinu et al. (2013) on ML08, achieving a new state-of-the-art, and matches the performance of the state-of-the-art neural network model of Hashimoto et al. (2014) on ML10, while being fully interpretable. Hence, our algorithm is able to successfully bridge the gap in performance between non-interpretable neural network models and interpretable count-based models such as APTs.

Model	ML10 - Average	ML08
Our work	**0.48**	**0.31**
Blacoe and Lapata (2012)	0.44	-
Hashimoto et al. (2014)	**0.48**	-
Weir et al. (2016)	0.43	0.26
Dinu et al. (2013)	-	0.23 – 0.26
Erk and Padó (2008)	-	0.27

Table 4: Comparison with existing methods.

4 Related Work

Composition with distributional semantic models has become a popular research area in recent years. Simple, yet competitive methods, are based on pointwise vector addition or multiplication (Mitchell and Lapata, 2008, 2010). However, these approaches neglect the structure of the text defining composition as a commutative operation.

A number of approaches proposed in the literature attempt to overcome this shortcoming by introducing weighted additive variants (Guevara, 2010, 2011; Zanzotto et al., 2010). Another popular strand of work models semantic composition on the basis of ideas arising in formal semantics. Composition in such models is usually implemented as operations on higher-order tensors (Baroni and Zamparelli, 2010; Baroni et al., 2014; Coecke et al., 2011; Grefenstette et al., 2011; Grefenstette and Sadrzadeh, 2011; Grefenstette et al., 2013; Kartsaklis and Sadrzadeh, 2014; Paperno et al., 2014; Tian et al., 2016; Van de Cruys et al., 2013). Another widespread approach to semantic composition is to use neural networks as composition functions, frequently over a syntactic tree of an input sentence, in an end-to-end system (Bowman et al., 2016; Hashimoto et al., 2014; Hill et al., 2016; Mou et al., 2015; Socher et al., 2012, 2014; Wieting et al., 2015; Yu and Dredze, 2015).

The above approaches are applied to untyped distributional vector space models where untyped models contrast with typed models (Baroni and Lenci, 2010) in terms of whether structural information is encoded in the representation as in the models of Erk and Padó (2008); Levy and Goldberg (2014a); Padó and Lapata (2007); Thater et al. (2010, 2011).

The perhaps most popular approach in the literature to evaluating compositional distributional semantic models is to compare human word and phrase similarity judgements with similarity estimates of composed meaning representations, under the assumption that better distributional representations will perform better at these tasks (Blacoe and Lapata, 2012; Dinu et al., 2013; Erk and Padó, 2008; Hashimoto et al., 2014; Hermann and Blunsom, 2013; Kiela et al., 2014; Turney, 2012).

5 Conclusion

In this paper we have introduced a novel form of distributional inference that generalises the method introduced by Kober et al. (2016). We have shown its effectiveness for semantic composition on two benchmark phrase similarity tasks where we achieved state-of-the-art performance while retaining the interpretability of our model. In future work we intend to scale up to longer phrases and full sentences, and apply offset inference to multi-word sub-components of a phrase.

References

Marco Baroni, Raffaella Bernardi, and Roberto Zamparelli. 2014. Frege in space: A program for compositional distributional semantics. *Linguistic Issues in Language Technology* 9(6):5–110.

Marco Baroni, Silvia Bernardini, Adriano Ferraresi, and Eros Zanchetta. 2009. The wacky wide web: a collection of very large linguistically processed web-crawled corpora. *Language Resources and Evaluation* 43(3):209–226. https://doi.org/10.1007/s10579-009-9081-4.

Marco Baroni and Alessandro Lenci. 2010. Distributional memory: A general framework for corpus-based semantics. *Computational Linguistics* 36(4):673–721.

Marco Baroni and Roberto Zamparelli. 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*. Association for Computational Linguistics, Cambridge, MA, pages 1183–1193.

William Blacoe and Mirella Lapata. 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*. Association for Computational Linguistics, Jeju Island, Korea, pages 546–556.

Samuel Bowman, Jon Gauthier, Abhinav Rastogi, Raghav Gupta, Christopher Manning, and Christopher Potts. 2016. A fast unified model for parsing and sentence understanding. In *Proceedings of ACL*. Association for Computational Lingustics, pages 1466–1477.

Bob Coecke, Mehrnoosh Sadrzadeh, and Stephen Clark. 2011. Mathematical foundations for a compositional distributed model of meaning. *Linguistic Analysis* 36(1-4):345–384.

Georgiana Dinu, Nghia The Pham, and Marco Baroni. 2013. General estimation and evaluation of compositional distributional semantic models. In *Proceedings of the Workshop on Continuous Vector Space Models and their Compositionality*. Association for Computational Linguistics, Sofia, Bulgaria, pages 50–58.

Katrin Erk and Sebastian Padó. 2008. A structured vector space model for word meaning in context. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Honolulu, Hawaii, pages 897–906.

Edward Grefenstette, Georgiana Dinu, Yao-Zhong Zhang, Mehrnoosh Sadrzadeh, and Marco Baroni. 2013. Multi-step regression learning for compositional distributional semantics. *Proceedings of the Tenth International Conference on Computational Semantics* .

Edward Grefenstette and Mehrnoosh Sadrzadeh. 2011. Experimental support for a categorical compositional distributional model of meaning. In *Proceedings of EMNLP*. Association for Computational Linguistics, pages 1394–1404. http://www.aclweb.org/anthology/D11-1129.

Edward Grefenstette, Mehrnoosh Sadrzadeh, Stephen Clark, Bob Coecke, and Stephen Pulman. 2011. Concrete sentence spaces for compositional distributional models of meaning. *Proceedings of the 9th International Conference on Computational Semantics (IWCS 2011)* pages 125–134.

Emiliano Guevara. 2010. A regression model of adjective-noun compositionality in distributional semantics. In *Proceedings of the ACL GEMS Workshop*. pages 33–37.

Emiliano Guevara. 2011. Computing semantic compositionality in distributional semantics. In *Proceedings of the Ninth International Conference on Computational Semantics*. Association for Computational Linguistics, Stroudsburg, PA, USA, IWCS '11, pages 135–144.

Kazuma Hashimoto, Pontus Stenetorp, Makoto Miwa, and Yoshimasa Tsuruoka. 2014. Jointly learning word representations and composition functions using predicate-argument structures. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, Doha, Qatar, pages 1544–1555. http://www.aclweb.org/anthology/D14-1163.

Karl Moritz Hermann and Phil Blunsom. 2013. The role of syntax in vector space models of compositional semantics. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Sofia, Bulgaria, pages 894–904.

Felix Hill, KyungHyun Cho, Anna Korhonen, and Yoshua Bengio. 2016. Learning to understand phrases by embedding the dictionary. *Transactions of the Association for Computational Linguistics* 4:17–30. http://www.aclweb.org/anthology/Q/Q16/Q16-1002.pdf.

Dimitri Kartsaklis and Mehrnoosh Sadrzadeh. 2014. A study of entanglement in a categorical framework of natural language. In *Proceedings of the 11th Workshop on Quantum Physics and Logic (QPL)*.

Douwe Kiela, Felix Hill, Anna Korhonen, and Stephen Clark. 2014. Improving multi-modal representations using image dispersion: Why less is sometimes more. In *Proceedings of ACL 2014*.

Thomas Kober, Julie Weeds, Jeremy Reffin, and David Weir. 2016. Improving sparse word representations with distributional inference for semantic composition. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Austin, Texas, pages 1691–1702. https://aclweb.org/anthology/D16-1175.

Omer Levy and Yoav Goldberg. 2014a. Dependency-based word embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Association for Computational Linguistics, Baltimore, Maryland, pages 302–308.

Omer Levy and Yoav Goldberg. 2014b. Neural word embedding as implicit matrix factorization. In Z. Ghahramani, M. Welling, C. Cortes, N.D. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, Curran Associates, Inc., pages 2177–2185.

Jeff Mitchell and Mirella Lapata. 2008. Vector-based models of semantic composition. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technology Conference*. Association for Computational Linguistics, Columbus, Ohio, pages 236–244.

Jeff Mitchell and Mirella Lapata. 2010. Composition in distributional models of semantics. *Cognitive Science* 34(8):1388–1429. https://doi.org/10.1111/j.1551-6709.2010.01106.x.

Lili Mou, Hao Peng, Ge Li, Yan Xu, Lu Zhang, and Zhi Jin. 2015. Discriminative neural sentence modeling by tree-based convolution. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Lisbon, Portugal, pages 2315–2325.

Joakim Nivre, Johan Hall, and Jens Nilsson. 2006. Maltparser: A data-driven parser-generator for dependency parsing. Technical report, Växjö University.

Sebastian Padó and Mirella Lapata. 2007. Dependency-based construction of semantic space models. *Computational Linguistics* 33(2):161–199. https://doi.org/10.1162/coli.2007.33.2.161.

Denis Paperno, Nghia The Pham, and Marco Baroni. 2014. A practical and linguistically-motivated approach to compositional distributional semantics. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Baltimore, Maryland, pages 90–99.

Richard Socher, Brody Huval, Christopher D. Manning, and Andrew Y. Ng. 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*. Association for Computational Linguistics, Jeju Island, Korea, pages 1201–1211.

Richard Socher, Andrej Karpathy, Quoc Le, Christopher Manning, and Andrew Ng. 2014. Grounded compositional semantics for finding and describing images with sentences. *Transactions of the Association for Computational Linguistics* 2:207–218.

James H Steiger. 1980. Tests for comparing elements of a correlation matrix. *Psychological Bulletin* 87(2):245.

Stefan Thater, Hagen Fürstenau, and Manfred Pinkal. 2010. Contextualizing semantic representations using syntactically enriched vector models. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Uppsala, Sweden, pages 948–957.

Stefan Thater, Hagen Fürstenau, and Manfred Pinkal. 2011. Word meaning in context: A simple and effective vector model. In *Proceedings of 5th International Joint Conference on Natural Language Processing*. Asian Federation of Natural Language Processing, Chiang Mai, Thailand, pages 1134–1143.

Ran Tian, Naoaki Okazaki, and Kentaro Inui. 2016. Learning semantically and additively compositional distributional representations. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Berlin, Germany, pages 1277–1287. http://www.aclweb.org/anthology/P16-1121.

Peter D. Turney. 2012. Domain and function: A dual-space model of semantic relations and compositions. *Journal of Artificial Intelligence Research* 44(1):533–585.

Tim Van de Cruys, Thierry Poibeau, and Anna Korhonen. 2013. A tensor-based factorization model of semantic compositionality. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, Atlanta, Georgia, pages 1142–1151.

David Weir, Julie Weeds, Jeremy Reffin, and Thomas Kober. 2016. Aligning packed dependency trees: a theory of composition for distributional semantics. *Computational Linguistics, special issue on Formal Distributional Semantics* 42(4):727–761.

John Wieting, Mohit Bansal, Kevin Gimpel, and Karen Livescu. 2015. From paraphrase database to compositional paraphrase model and back. *TACL* 3:345–358. http://aclweb.org/anthology/Q/Q15/Q15-1025.pdf.

Mo Yu and Mark Dredze. 2015. Learning composition models for phrase embeddings. *Transactions of the Association for Computational Linguistics* 3:227–242. http://aclweb.org/anthology/Q/Q15/Q15-1017.pdf.

Fabio Massimo Zanzotto, Ioannis Korkontzelos, Francesca Fallucchi, and Suresh Manandhar. 2010. Estimating linear models for compositional distributional semantics. In *Proceedings of Coling*. pages 1263–1271. http://www.aclweb.org/anthology/C10-1142.