Refining Pretrained Word Embeddings Using Layer-wise Relevance Propagation

Akira Utsumi
Department of Informatics & Artificial Intelligence eXploration Research Center
The University of Electro-Communications, Chofu, Tokyo 182-8585, Japan
utsumi@uec.ac.jp

Abstract

In this paper, we propose a simple method for refining pretrained word embeddings using layer-wise relevance propagation. Given a target semantic representation one would like word vectors to reflect, our method first trains the mapping between the original word vectors and the target representation using a neural network. Estimated target values are then propagated backward toward word vectors, and a relevance score is computed for each dimension of word vectors. Finally, the relevance score vectors are used to refine the original word vectors so that they are projected into the subspace that reflects the information relevant to the target representation. The evaluation experiment using binary classification of word pairs demonstrates that the refined vectors by our method achieve the higher performance than the original vectors.

1 Introduction

The recent success of neural NLP is partially but largely due to the development of word embedding techniques (Goldberg, 2017). Although a considerable number of studies have been made on training word embeddings from distributional information of language (Mikolov et al., 2013; Pennington et al., 2014; Bojanowski et al., 2017; Nickel and Kiela, 2017), one recent research trend is to refine or fine-tune pretrained word embeddings. One promising approach is the use of other information such as multimodal information (Bruni et al., 2014; Kiela et al., 2014; Kiela and Clark, 2015; Kiela et al., 2015a; Silberer et al., 2017) and language resources (Faruqui et al., 2015; Faruqui and Dyer, 2015; Kiela et al., 2015b; Rothe and Schütze, 2017; Yu and Dredze, 2014). Other refinement methods include task-specific embeddings (Bolukbasi et al., 2016; Yu et al., 2017) and the selective use of multiple embeddings (Bollegala et al., 2017; Kiela et al., 2018).

In this paper, we propose a different approach to refining pretrained word embeddings so that word vectors reflect the information relevant for a specific knowledge. Our method utilizes layer-wise relevance propagation (Bach et al., 2015; Samek et al., 2017), which has been proposed as a general framework for decomposing predictions of modern AI systems, in particular deep learning systems. The basic idea of layer-wise relevance propagation is to quantitatively measure the contribution of each fragment of an input vector (e.g., a single pixel of an image) to the prediction as a relevance score. Using relevance scores, our method projects word vectors into the subspace that better reflects the target knowledge. The assumption underlying our approach is that the information for any given target knowledge is contained in pretrained word embeddings. Our method attempts to make the best use of the information contained in word vectors by estimating the importance in reflecting a target knowledge.

To the best of our knowledge, this paper is the first to employ the technique of layer-wise relevance propagation for refining word embeddings. Our method can be applied to word vectors $\boldsymbol{x}$ trained by any word embedding method. This implies that our method does not compete with other refinement methods, but they are complementary; it can be used for word vectors refined by other methods. In addition, our method can refine word vectors for any target knowledge $\boldsymbol{y}$, from a single binary value to a structured representation, as long as a function $\boldsymbol{y} = f(\boldsymbol{x})$ can be learned.

2 Method for Refining Word Vectors

Our method comprises the following three steps: (1) it trains a prediction function from a pretrained word vector to a target representation; (2) computes a relevance score for each dimension of the

Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 4840–4846
Brussels, Belgium, October 31 - November 4, 2018.

word vector; and (3) projects word vectors into the subspace using the relevance scores. In this section, these three steps are explained in detail.

2.1 Training the Prediction Function

Given pairs of an input word vector $\boldsymbol{x}^{(i)}$ to be refined and a target knowledge representation $\boldsymbol{y}^{(i)}$ for a word $w^{(i)}$, the proposed method trains a function $\boldsymbol{y}^{(i)} = f(\boldsymbol{x}^{(i)})$. In this paper, we use a neural network as a learning method, but other learning methods such as linear transformation and SVM can also be used. Note that a scalar value or a class label can be used as a target representation $\boldsymbol{y}^{(i)}$.

2.2 Computing Relevance Scores

This step derives an explanation of the prediction in terms of input variables, namely the importance of each dimension of a word vector $\boldsymbol{x}^{(i)}$ for the prediction $\hat{\boldsymbol{y}}^{(i)} = f(\boldsymbol{x}^{(i)})$. In layer-wise relevance propagation, the score of the correct prediction $\hat{y}_j^{(i)}$ is redistributed backward using relevance propagation rules. By repeatedly applying propagation rules, it assigns a relevance score $r_k^{(i,j)}$ to each dimension $x_k^{(i)}$ of a word vector $\boldsymbol{x}^{(i)}$. As a result, a relevance score vector $\boldsymbol{r}^{(i,j)}$ is obtained for each word vector $\boldsymbol{x}^{(i)}$ and target dimension $y_j^{(i)}$.

Among a number of propagation rules (Bach et al., 2015), we use the "alpha-beta" rule for multilayer neural networks. The relevance score $R_i^{(l)}$ of the i-th unit $u_i^{(l)}$ in the l-th layer is a function of upper-layer relevances $R_j^{(l+1)}$ defined by:

$$R_{i\leftarrow j}^{(l,l+1)} = R_j^{(l+1)} \cdot \left(\alpha \frac{z_{ij}^{+}}{\sum_i z_{ij}^{+}} + \beta \frac{z_{ij}^{-}}{\sum_i z_{ij}^{-}} \right) \quad (1)$$

$$R_i^{(l)} = \sum_j R_{i\leftarrow j}^{(l,l+1)} \quad (2)$$

$$z_{ij}^{(l,l+1)} = x_i^{(l)} w_{ij}^{(l,l+1)} \quad (3)$$

where $x_i^{(l)}$ is an activation of the unit $u_i^{(l)}$, $w_{ij}^{(l,l+1)}$ is a weight connecting $u_i^{(l)}$ to $u_j^{(l+1)}$, and z_{ij}^{+} and z_{ij}^{-} denote the positive and negative part of $z_{ij}^{(l,l+1)}$. As a result, relevance scores $r_k^{(i,j)}$ of the word vector $\boldsymbol{x}^{(i)}$ and the target dimension $y_j^{(i)}$ are obtained as relevance scores $R_k^{(1)}$ of the input layer. The parameters α and β denote the importance of positive and negative evidence for predicting a target representations and should be chosen such that $\alpha + \beta = 1$. In this paper, we assume that positive and negative evidence equally contributes to the prediction and thus set $\alpha = \beta = 0.5$.

2.3 Projecting Word Vectors into a Subspace

The basic idea of projection is that n-dimensional word vectors are projected into m-dimensional vectors whose relevance scores are more than or equal to a threshold θ_R.

First, for a target dimension j of $\boldsymbol{y}$, relevance score vectors are averaged over words relevant to the target dimension as follows:

$$\overline{\boldsymbol{r}}^{(j)} = g_2\left(\frac{\sum_{w_i \in V_j} g_1(\boldsymbol{r}^{(i,j)})}{|V_j|} \right) \quad (4)$$

$$\{g_1(\boldsymbol{x})\}_i = \begin{cases} x_i & (x_i \geq \theta_{R_1}) \\ 0 & (\text{otherwise}) \end{cases}$$

$$\{g_2(\boldsymbol{x})\}_i = \begin{cases} \frac{x_i}{\max_i x_i} & (\frac{x_i}{\max_i x_i} \geq \theta_{R_2}) \\ 0 & (\text{otherwise}) \end{cases}$$

where V_j is a set of words $w^{(i)}$ such that $\hat{y}_j^{(i)} \geq \theta_T$. The functions g_1 and g_2 are used for downplaying irrelevant dimensions. For example, the target knowledge is the property of *Visually dark* and $V_{visually\_dark}$ is {*chocolate, crow, night*}. By averaging relevance score vectors of these words, we obtain the mean relevance vector $\overline{\boldsymbol{r}}^{(visually\_dark)}$ that represents the importance of word vector dimension in predicting whether a given word has the property of *Visually dark*.

Finally, using the mean relevance vector $\overline{\boldsymbol{r}}^{(j)}$, word vectors $\boldsymbol{x}_i$ is transformed into vectors $\boldsymbol{z}_i^{(j)}$ of a subspace for the target dimension. This is achieved by weighting $\boldsymbol{x}_i$ by component-wise multiplication of $\boldsymbol{x}_i$ and $\overline{\boldsymbol{r}}^{(j)}$ and removing the dimensions of zero relevance. Formally, the projection is defined by the n by m projection matrix $\boldsymbol{T}^{(j)}$ as follows:

$$\boldsymbol{z}_i^{(j)} = \boldsymbol{x}_i \boldsymbol{T}^{(j)} \quad (5)$$

$$T_{ik}^{(j)} = \begin{cases} \overline{r}_i^{(j)} & (\overline{r}_i^{(j)} > 0 \text{ and it is the } k\text{-th} \\ & \text{nonzero dimension of } \overline{\boldsymbol{r}}^{(j)}) \\ 0 & (\text{otherwise}) \end{cases} \quad (6)$$

3 Evaluation Experiment

In order to justify the effectiveness of the proposed method, we conducted an evaluation experiment using binary classification of word pairs.

Corpus: All word vectors were trained on the Corpus of Contemporary American English

Domain	Properties
Vision	Vision, Bright, Dark, Color, Pattern, Large, Small, Motion, Biomotion, Fast, Slow, Shape, Complexity, Face, Body
Somatic	Touch, Temperature, Texture, Weight, Pain
Audition	Audition, Loud, Low, High, Sound, Music, Speech
Gustation	Taste
Olfaction	Smell
Motor	Head, UpperLimb, LowerLimb, Practice
Spatial	Landmark, Path, Scene, Near, Toward, Away, Number
Temporal	Time, Duration, Long, Short
Causal	Caused, Consequential
Social	Social, Human, Communication, Self
Cognition	Cognition
Emotion	Benefit, Harm, Pleasant, Unpleasant, Happy, Sad, Angry, Disgusted, Fearful, Surprised
Drive	Drive, Needs
Attention	Attention, Arousal

Table 1: 65 properties in Binder et al.'s (2016) dataset

(COCA), which includes 0.56G word tokens. Words that occurred less than 30 times in the corpus were ignored, resulting in the vocabulary of 108,230 words. Three context windows of size 3, 5, and 10 were used for training.

Word embedding: We used two representative models, namely skip-gram with negative sampling (SGNS) (Mikolov et al., 2013) and GloVe (Pennington et al., 2014). We trained 100-, 200- and 300-dimensional word vectors from the corpus.

Target knowledge representation: We used Binder et al.'s (2016) brain-based semantic vectors of 535 words as a target representation. [1] This representation comprises 65 properties in Table 1, which are based entirely on functional divisions in the human brain. Each word is represented as a 65-dimensional vector and each dimension corresponds to one of these properties. Each value of the brain-based vectors represents the salience of the corresponding property, which is calculated as a mean salience rating on a 7-point scale ranging from 0 to 6. Because these properties are based on not only perceptual properties but also a variety of other properties such as affective, social, and cognitive ones, this dataset is suitable for evaluation.

Refining word vectors: The prediction function f was trained using a three-layer neural network comprising an input layer for n-dimensional word vectors, one hidden layer with $n/2$ sigmoid units, and a linear output layer. The parameters θ_T, θ_{R_1} and θ_{R_2} for projection were estimated using 10-fold cross-validation and grid search. [2]

[1] http://www.neuro.mcw.edu/semanticrepresentations.html

Bright, Dark, Color, Pattern, Large, Small, Motion, Fast, Slow, Shape, Temperature, Texture, Weight, Loud, Sound, Taste, Smell, Fearful

Table 2: 18 properties in CSLB dataset

Task: We used a binary classification task of judging whether a pair of words is similar or not with respect to each property of Table 1. For example, *night* and *chocolate* should be judged as similar with respect to the property of *Dark*, while *night* and *ice* should be judged as dissimilar with respect to that property. For each property, we chose 10 words with the highest salience and 10 words with the lowest salience from the vocabulary of brain-based vectors, and generated 45 high-salience word pairs and 100 pairs of high-salience and low-salience words. Note that we did not consider low-score word pairs because it does not make sense to ask whether words (e.g., *peace* and *wit*) that do not have a property (e.g., *Dark*) are similar with respect to that property.

To confirm the generality of our method, we also generated another evaluation dataset for untrained words (i.e., words not included in Binder et al.'s vocabulary) using CSLB concept property norms of 638 words (Devereux et al., 2014). [3] After removing words contained in Binder et al.'s vocabulary, we chose properties that were closely related to Binder et al.'s properties and possessed by at least 10 words. As a result, the generated dataset contained 18 properties listed in Table 2, because the property norm mainly includes perceptual and functional properties.

Binary classification was carried out by computing cosine similarity between vectors of paired words and classifying the n highest pairs into similar pairs. Hence, the classification performance was measured by average precision.

4 Results

Table 3 shows mean average precisions across 65 properties for the original word embeddings (Orig) and the refined embeddings by our method (Refn). The asterisk indicates that the mean average precision of the refined vectors is signifi-

[2] The range in grid search was $[3.0, 4.5]$ with a step size of 0.1 for θ_T, $[0.0, 0.02n]$ with a step size of $0.001n$ for θ_{R_1} of n hundred word vector dimension, and $[0.0, 0.7]$ with a step size of 0.05 for θ_{R_2}.

[3] https://cslb.psychol.cam.ac.uk/propnorms

		SGNS		GloVe	
win	dim	Orig	Refn	Orig	Refn
10	300	75.3	**78.6***	67.4	**70.4***
10	200	75.9	**79.3***	67.8	**73.7***
10	100	76.1	**77.0***	68.7	**71.6***
5	300	75.4	**78.8***	67.7	**71.8***
5	200	75.6	**79.4***	68.0	**73.9***
5	100	77.2	**78.3**	68.9	**70.1**
3	300	75.5	**79.3***	67.6	**71.5***
3	200	76.5	**77.9***	68.2	**70.8***
3	100	77.4	**79.0***	68.4	**71.2***

Table 3: Mean average precision for Binder et al.'s (2016) dataset

		SGNS		GloVe	
win	dim	Orig	Refn	Orig	Refn
10	300	57.9	**60.4***	56.8	**58.5**
10	200	58.0	**59.3**	56.3	**56.6**
10	100	**58.8**	58.3	55.9	**56.0**
5	300	58.8	**61.1***	56.4	**59.3***
5	200	58.5	**62.6***	55.6	**56.8**
5	100	58.9	**60.9**	**56.5**	55.5
3	300	58.5	**58.8**	55.2	**55.6**
3	200	58.9	**59.3**	**54.5**	54.1
3	100	**59.3**	58.8	53.5	**54.4**

Table 4: Mean average precision for CSLB property norm dataset

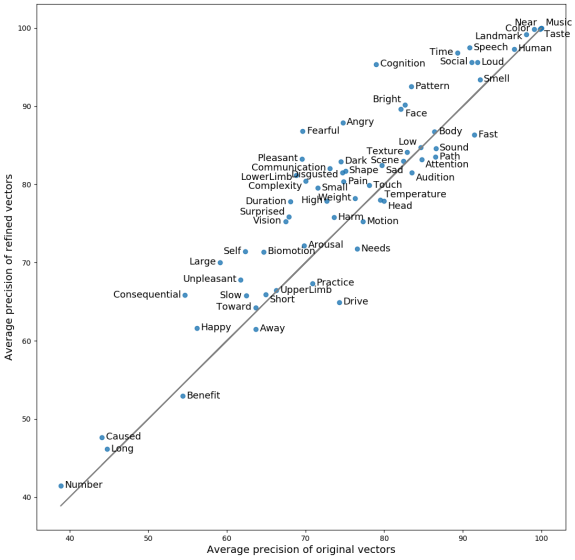


Figure 1: A scatterplot of average precision of the original versus refined vectors for 65 properties in the case of SGNS with win= 5 and dim= 200. The diagonal reference line $y = x$ indicates that the original and refined vectors have equal precision.

cantly higher than that of the original vectors by Wilcoxon signed-rank test ($p < .05$). For all word embeddings, the refined vectors achieved higher mean average precision than the original ones. Furthermore, in almost all cases, the improvement is statistically significant. This result demonstrates that the proposed method is successful in refining word embeddings so that vector similarity better reflects the target knowledge.

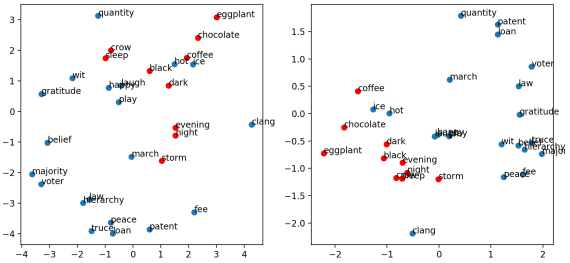
Figure 1 depicts the difference of average precision between the original word vectors and the refined vectors for each target property. Most of the properties are plotted above the diagonal reference line, indicating that these properties are better represented by the refined vectors. Note that properties plotted below the diagonal line, for which refined word vectors yielded lower precision than the original vectors, are sensorimotor or spatiotemporal properties. This result is consistent with Utsumi's (2018) finding that these kinds of knowledge are less likely to be encoded in word vectors.

Table 4 shows the result of binary classification for CSLB property norm dataset. In most cases, the refined vectors of untrained words also yielded better performance than the original vectors. In some cases, however, refinement did not improve the performance. One of the reasons for this failure would be that a small set of vocabulary words in Binder et al.'s (2016) dataset is not enough for the subspace to generalize to untrained words.

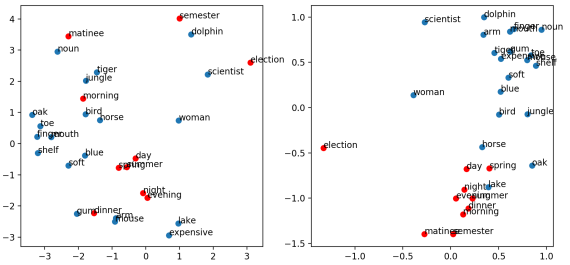
To confirm whether the projected subspace better reflects the target knowledge than the original space, we visualize both spaces using MDS in Figure 2. Although all 535 words are embedded into the two-dimensional space, Figure 2 only shows words used in binary classification task, namely words with the 10 highest salience (denoted by red dots) and 20 lowest salience for a given property. As shown in Figure 2, our method refines the vectors of salient words to be more similar in the subspace, while preserving the other similarity of words.

5 Related Work

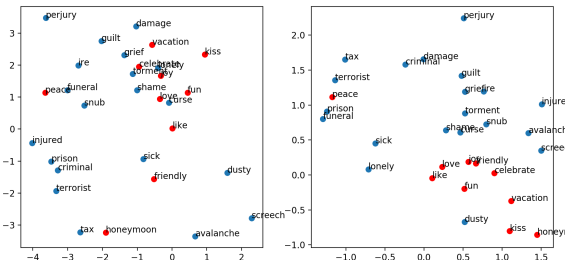
Prior work on word embedding refinement can be classified into general purpose refinement and specific target refinement. Many existing studies have attempted to refine word vectors to improve the performance of general-purpose similarity computation. These studies generally re-



(a) Property *Dark*



(b) Property *Time*



(c) Property *Pleasant*

Figure 2: Two-dimensional MDS visualization of the original space (trained by SGNS with `win`= 5 and `dim`= 200) and the projected subspace ($\theta_T = 3.0$, $\theta_{R_1} = 0.026$ and $\theta_{R_2} = 0.10$). Left: Original space, Right: Projected subspace

fine word vectors by solving an optimization problem whose objective function reflects the similarity obtained by language resources, such as WordNet (Faruqui et al., 2015; Yu and Dredze, 2014; Rothe and Schütze, 2017), Freebase (Rothe and Schütze, 2017), Paraphrase Database (Faruqui et al., 2015; Yu and Dredze, 2014), free association norm (Kiela et al., 2015b), and dictionary (Wang et al., 2015). Our method differs from them in that it is proposed for specific target refinement. In other words, the refined vectors by general purpose refinement method can be further refined to extract a specific knowledge by our method.

Most prior studies for specific purpose refinement propose a method specialized for a specific task such as sentiment analysis (Labutov and Lipson, 2013; Tang et al., 2016; Yu et al., 2017) and lexical entailment (Mrkšić et al., 2016; Vulić and Mrkšić, 2018). On the other hand, our method refines word vectors for a specific knowledge or task, but it is not specialized for a knowledge or task.

Rothe et al. (2016) and Rothe and Schütze (2016) are conceptually similar to our approach; their method refines word vectors for a specific knowledge but it is not specialized for a certain task. The merit of our method is that any types of representation can be used as a target, while their method is limited to binary labels. Furthermore, while their method learns an orthogonal transformation of pretrained word vectors by directly optimizing the objective function, our method can project word vectors to a subspace independent of training method for a prediction function.

6 Conclusion

In this paper, we propose a method for refining pretrained word vectors using layer-wise relevance propagation. We demonstrated that the proposed method can refine word vectors so that they better reflect the target knowledge. One of our motivation is to make embeddings more interpretable and useful. In other studies (Utsumi, 2015, 2018), we have analyzed the internal knowledge encoded in text-based word embeddings, while this study is the first step toward a general method for utilizing the internal knowledge of word embeddings.

In future work, we have to modify the refinement method by relevance propagation to be more effective by exploring the mechanism of how the internal knowledge of word vectors is extracted by multilayer neural networks and examining the effectiveness of other relevance propagation methods. It would also be vital for future work to explore efficient combinations with other refinement methods using language resources.

Acknowledgments

This research was supported by JSPS KAKENHI Grant Numbers JP15H02713 and SCAT Research Grant.

References

Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. 2015. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLoS ONE*, 10(7):e0130140.

Jeffrey R. Binder, Lisa L. Conant, Colin J. Humphries, Leonardo Fernandino, Stephen B. Simons, Mario

Aguilar, and Rutvik H. Desai. 2016. Toward a brain-based componential semantic representation. *Cognitive Neuropsychology*, 33(3–4):130–174.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Danushka Bollegala, Kohei Hayashi, and Ken-ichi Kawarabayashi. 2017. Learning linear transformations between counting-based and prediction-based word embeddings. *PLoS ONE*, 12(9):e0184544.

Tolga Bolukbasi, Kai-Wei Chang, James Y. Zou, Venkatesh Saligrama, and Adam T. Kalai. 2016. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *Advances in Neural Information Processing Systems 29*, pages 4349–4357.

Elia Bruni, Nam K. Tran, and Marco Baroni. 2014. Multimodal distributional semantics. *Jounral of Artificial Intelligence Research*, 49:1–47.

Barry J. Devereux, Lorraine K. Tyler, Jeroen Geertzen, and Billi Randall. 2014. The Centre for Speech, Language and the Brain (CSLB) concept property norms. *Behavior Research Methods*, 46:1119–1127.

Manaal Faruqui, Jesse Dodge, Kumar Sujay Jauhar, Chris Dyer, Eduard Hovy, and A. Noah Smith. 2015. Retrofitting word vectors to semantic lexicons. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1606–1615.

Manaal Faruqui and Chris Dyer. 2015. Non-distributional word vector representation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*, pages 464–469.

Yoab Goldberg. 2017. *Neural Network Methods for Natural Language Processing*. Morgan & Claypool Publishers.

Douwe Kiela, Luana Bulat, and Stephen Clark. 2015a. Grounding semantics in olfactory perception. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pages 231–236.

Douwe Kiela and Stephen Clark. 2015. Multi- and cross-modal semantics beyond vision: Grounding in auditory perception. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2461–2470.

Douwe Kiela, Felix Hill, and Stephen Clark. 2015b. Specializing word embeddings for similarity or relatedness. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2044–2048.

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 the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 835–841.

Douwe Kiela, Changhan Wang, and Kyunghyun Cho. 2018. Context-attentive embeddings for improved sentence representations. *arXiv:1804.07983 [cs.CL]*.

Igor Labutov and Hod Lipson. 2013. Re-embedding words. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 489–493.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *Proceedings of Workshop at the International Conference on Learning Representation (ICLR)*.

Nikola Mrkšić, Diarmuid Ó Séaghdha, BlaiseThomson, Milica Gašić, Lina Rojas-Barahona, Pei-Hao Su, David Vandyke, Tsung-Hsien Wen, and Steve Young. 2016. Counter-fitting word vectors to linguistic constraints. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 142–148.

Maximilian Nickel and Douwe Kiela. 2017. Poincaré embeddings for learning hierarchical representations. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 6338–6347.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1532–1543.

Sascha Rothe, Sebastian Ebert, and Hinrich Schütze. 2016. Ultradense word embeddings by orthogonal transformation. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 767–777.

Sascha Rothe and Hinrich Schütze. 2016. Word embedding calculus in meaningful ultradense subspaces. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 512–517.

Sascha Rothe and Hinrich Schütze. 2017. Autoextend: Combining word embeddings with semantic resources. *Computational Linguistics*, 43(3):593–617.

Wojciech Samek, Thomas Wiegand, and Klaus-Robert Müller. 2017. Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models. *arXiv:1708.08296 [cs.AI]*.

Carina Silberer, Vittorio Ferrari, and Mirella Lapata. 2017. Visually grounded meaning representations. *IEEE Transactions on Pattern Recognition and Machine Intelligence*, 39(11):2284–2297.

Duyu Tang, Furu Wei, Bing Qin, Nan Yang, Ting Liu, and Ming Zhou. 2016. Sentiment embeddings with applications to sentiment analysis. *IEEE Transactions on Knowledge and Data Engineering*, 28:496–509.

Akira Utsumi. 2015. A complex network approach to distributional semantic models. *PLoS ONE*, 10(8):e0136277.

Akira Utsumi. 2018. A neurobiologically motivated analysis of distributional semantic models. In *Proceedings of the 40th Annual Conference of the Cognitive Science Society (CogSci2018)*, pages 1147–1152.

Ivan Vulić and Nikola Mrkšić. 2018. Specialising word vectors for lexical entailment. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1134–1145.

Tong Wang, Abdel-rahman Mohamed, and Graeme Hirst. 2015. Learning lexical embeddings with syntactic and lexicographic knowledge. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 458–463.

Liang-Chih Yu, Jin Wang, K. Robert Lai, and Xuejie Zhang. 2017. Refining word embeddings for sentiment analysis. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 534–539.

Mo Yu and Mark Dredze. 2014. Improving lexical embeddings with semantic knowledge. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 545–550.