

# Analytical Methods for Interpretable Ultradense Word Embeddings

Philipp Dufter, Hinrich Schütze

Center for Information and Language Processing (CIS)

LMU Munich, Germany

[philipp@cis.lmu.de](mailto:philipp@cis.lmu.de)

## Abstract

Word embeddings are useful for a wide variety of tasks, but they lack interpretability. By rotating word spaces, interpretable dimensions can be identified while preserving the information contained in the embeddings without any loss. In this work, we investigate three methods for making word spaces interpretable by rotation: Densifier (Rothe et al., 2016), linear SVMs and DensRay, a new method we propose. In contrast to Densifier, DensRay can be computed in closed form, is hyperparameter-free and thus more robust than Densifier. We evaluate the three methods on lexicon induction and set-based word analogy. In addition we provide qualitative insights as to how interpretable word spaces can be used for removing gender bias from embeddings.

## 1 Introduction

Distributed representations for words have been of interest in natural language processing for many years. Word embeddings have been particularly effective and successful. On the downside, embeddings are generally not interpretable. But interpretability is desirable for several reasons. i) Semantically or syntactically similar words can be extracted: e.g., for lexicon induction. ii) Interpretable dimensions can be used to evaluate word spaces by examining which information is covered by the embeddings. iii) Computational advantage: for a high-quality sentiment classifier only a couple of dimensions of a high-dimensional word space are relevant. iv) By removing interpretable dimensions one can remove unwanted information (e.g., gender bias). v) Most importantly, interpretable embeddings support the goal of interpretable deep learning models.

Orthogonal transformations have been of particular interest in the literature. The reason is twofold: under the assumption that existing word

embeddings are of high-quality one would like to preserve the original embedding structure by using orthogonal transformations (i.e., preserving original distances). Park et al. (2017) provide evidence that rotating existing dense word embeddings achieves the best performance across a range of interpretability tasks.

In this work we modify the objective function of Densifier (Rothe et al., 2016) such that a closed form solution becomes available. We call this method DensRay. Following Amir et al. (2015) we compute simple linear SVMs, which we find to perform surprisingly well. We compare these methods on the task of lexicon induction.

Further, we show how interpretable word spaces can be applied to other tasks: first we use interpretable word spaces for debiasing embeddings. Second we show how they can be used for solving the set-based word analogy task. To this end, we introduce the set-based method IntCos, which is closely related to LRCos introduced by Drozd et al. (2016). We find IntCos to perform comparable to LRCos, but to be preferable for analogies which are hard to solve.

Our contributions are: **i)** We modify Densifier’s objective function and derive an analytical solution for computing interpretable embeddings. **ii)** We show that the analytical solution performs as well as Densifier but is more robust. **iii)** We provide evidence that simple linear SVMs are best suited for the task of lexicon induction. **iv)** We demonstrate how interpretable embedding spaces can be used for debiasing embeddings and solving the set-based word analogy task. The source code of our experiments is available.<sup>1</sup>

<sup>1</sup><https://github.com/pdufter/densray>

## 2 Methods

### 2.1 Notation

We consider a vocabulary  $V := \{v_1, v_2, \dots, v_n\}$  together with an embedding matrix  $E \in \mathbb{R}^{n \times d}$  where  $d$  is the embedding dimension. The  $i$ th row of  $E$  is the vector  $e_i$ .<sup>2</sup> We require an annotation for a specific linguistic feature (e.g., sentiment) and denote this annotation by  $l : V \rightarrow \{-1, 1\}$ . The objective is to find an orthogonal matrix  $Q \in \mathbb{R}^{d \times d}$  such that  $EQ$  is interpretable, i.e., the values of the first  $k$  dimensions correlate well with the linguistic feature. We refer to the first  $k$  dimensions as interpretable ultradense word space. We interpret  $x \in \mathbb{R}^n$  as a column vector and  $x^\top$  as a row vector. Further, we normalize all word embeddings with respect to the euclidean norm.

### 2.2 DensRay

Throughout this section  $k = 1$ . Given a linguistic signal  $l$  (e.g., sentiment), consider  $L_+ := \{(v, w) \in V \times V \mid l(v) = l(w)\}$ , and analogously  $L_-$ . We call  $d_{vw} := e_v - e_w$  a difference vector.

Densifier (Rothe et al., 2016) solves the following optimization problem,

$$\max_q \sum_{(v,w) \in L_-} \alpha_- \|q^\top d_{vw}\|_2 - \sum_{(v,w) \in L_+} \alpha_+ \|q^\top d_{vw}\|_2,$$

subject to  $q^\top q = 1$  and  $q \in \mathbb{R}^d$ . Further  $\alpha_-, \alpha_+ \in [0, 1]$  are hyperparameters. We now modify the objective function: we use the squared euclidean norm instead of the euclidean norm, something that is frequently done in optimization to simplify the gradient. The problem becomes then

$$\max_q \sum_{(v,w) \in L_-} \alpha_- \|q^\top d_{vw}\|_2^2 - \sum_{(v,w) \in L_+} \alpha_+ \|q^\top d_{vw}\|_2^2. \quad (1)$$

Using  $\|x\|_2^2 = x^\top x$  together with associativity of the matrix product we can simplify to

$$\begin{aligned} \max_q q^\top \left( \alpha_- \sum_{(v,w) \in L_-} d_{vw} d_{vw}^\top - \right. & (2) \\ \left. \alpha_+ \sum_{(v,w) \in L_+} d_{vw} d_{vw}^\top \right) q \\ =: \max_q q^\top A q \quad \text{subject to } q^\top q = 1. \end{aligned}$$

<sup>2</sup>We denote the vector corresponding to a word  $w$  by  $e_w$ .

Thus we aim to maximize the Rayleigh quotient of  $A$  and  $q$ . Note that  $A$  is a real symmetric matrix. Then it is well known that the eigenvector belonging to the maximal eigenvalue of  $A$  solves the above problem (cf. Horn et al. (1990, Section 4.2)). We call this analytical solution **DensRay**.

A second dimension that is orthogonal to the first dimension and encodes the linguistic features second strongest is given by the eigenvector corresponding to the second largest eigenvalue. The matrix of  $k$  eigenvectors of  $A$  ordered by the corresponding eigenvalues yields the desired matrix  $Q$  (cf. Horn et al. (1990, Section 4.2)) for  $k > 1$ . Due to  $A$  being a real symmetric matrix,  $Q$  is always orthogonal.

### 2.3 Comparison to Densifier

We have shown that DensRay is a closed form solution to our new formalization of Densifier. This formalization entails differences.

**Case  $k > 1$ .** While both methods – Densifier and DensRay – yield ultradense  $k$  dimensional subspaces. While we show that the spaces are comparable for  $k = 1$  we leave it to future work to examine how the subspaces differ for  $k > 1$ .

**Multiple linguistic signals.** Given multiple linguistic features, Densifier can obtain a single orthogonal transformation simultaneously for all linguistic features with chosen dimensions reserved for different features. DensRay can encode multiple linguistic features in one transformation only by iterative application.

**Optimization.** Densifier is based on solving an optimization problem using stochastic gradient descent with iterative orthogonalization of  $Q$ . DensRay, in contrast, is an analytical solution. Thus we expect DensRay to be more robust, which is confirmed by our experiments.

### 2.4 Geometric Interpretation

Assuming we normalize the vectors  $d_{vw}$  one can interpret Eq. 1 as follows: we search for a unit vector  $q$  such that the square of the cosine similarity with  $d_{vw}$  is large if  $(v, w) \in L_-$  and small if  $(v, w) \in L_+$ . Thus, we identify dimensions that are parallel/orthogonal to difference vectors of words belonging to different/same classes. It seems reasonable to consider the average cosine similarity. Thus if  $n_+, n_-$  is the number of elements in  $L_+, L_-$  one can choose  $\alpha_- = n_+^{-1}$  and  $\alpha_+ = n_-^{-1}$ .

### 3 Lexicon Induction

We show that DensRay and Densifier indeed perform comparably using the task of lexicon induction. We adopt [Rothe et al. \(2016\)](#)’s experimental setup. We also use [Rothe et al. \(2016\)](#)’s code for Densifier. Given a word embedding space and a sentiment/concreteness dictionary (binary or continuous scores where we binarize continuous scores using the median), we identify a one-dimensional interpretable subspace. Subsequently we use the values along this dimension to predict a score for unseen words and report Kendall’s  $\tau$  rank correlation with the gold scores.

To ensure comparability across methods we have redone all experiments in the same setting: we deduplicated lexicons, removed a potential train/test overlap and ignored neutral words in the lexicons. We set  $\alpha_{\neq} = \alpha_{=} = 0.5$  to ensure comparability between Densifier and DensRay.

Additionally we report results created by linear SVM/SVR inspired by their good performance as demonstrated by [Amir et al. \(2015\)](#). While they did not use linear kernels, we require linear kernels to obtain interpretable dimensions. Naturally the normal vector of the hyperplane in SVMs/SVRs reflects an interpretable dimension. An orthogonal transformation can be computed by considering a random orthogonal basis of the null space of the interpretable dimension.

Table 1 shows results. As expected the performance of Densifier and DensRay is comparable (macro mean deviation of 0.001). We explain slight deviations between the results with the slightly different objective functions of DensRay and Densifier. In addition, the re-orthogonalization used in Densifier can result in an unstable training process. Figure 1 assesses the stability by reporting mean and standard deviation for the concreteness task (BWK lexicon). We varied the size of the training lexicon as depicted on the x-axis and sampled 40 subsets of the lexicon with the prescribed size. For the sizes 512 and 2048 Densifier shows an increased standard deviation. This is because there is at least one sample for which the performance significantly drops. Removing the re-orthogonalization in Densifier prevents the drop and restores performance. Recent work ([Zhao and Schütze, 2019](#)) also finds that replacing the orthogonalization with a regularization is reasonable in certain circumstances. Given that DensRay and Densifier yield the same perfor-

mance and DensRay is a stable closed form solution always yielding an orthogonal transformation we conclude that DensRay is preferable.

Surprisingly, simple linear SVMs perform best in the task of lexicon induction. SVR is slightly better when continuous lexica are used for training (line 8). Note that the eigendecomposition used in DensRay yields a basis with dimensions ordered by their correlation with the linguistic feature. An SVM can achieve this only by iterated application.

Task	Emb.	Lex. (Train)	Lex. (Test)	Dens.	DensRay	SVR	SVM
1 sent	CZ	SubLex	SubLex	0.546	0.549	<b>0.585</b>	0.585
2 sent	DE	GermanPC	GermanPC	0.636	0.631	0.674	<b>0.677</b>
3 sent	ES	fullstrength	fullstrength	0.541	0.546	0.571	<b>0.576</b>
4 sent	FR	FEEL	FEEL	0.469	0.471	0.555	<b>0.565</b>
5 sent	EN	WHM	WHM	0.623	0.623	<b>0.627</b>	0.625
6 sent	EN(t)	WHM	SE Trial*	0.624	0.621	0.618	<b>0.637</b>
7 sent	EN(t)	WHM	SE Test*	0.600	0.608	0.619	<b>0.636</b>
8 conc	EN	BWK*	BWK*	0.599	0.602	<b>0.655</b>	0.641
9	Macro Mean			0.580	0.581	0.613	<b>0.618</b>

Table 1: Results on lexicon induction. Numbers are Kendall  $\tau$  rank correlation. For details on the resources see Table 2 and ([Rothe et al., 2016](#)). Bold: best result across methods. \*: continuous lexicon.

Name	Description
CZ, DE, ES	Czech, German, Spanish embeddings by ( <a href="#">Rothe et al., 2016</a> )
FR	French frWac embeddings ( <a href="#">Fauconnier, 2015</a> )
EN	English GoogleNews embeddings ( <a href="#">Mikolov et al., 2013</a> )
EN(t)	English Twitter Embeddings ( <a href="#">Rothe et al., 2016</a> )
Name	Description
SubLex	Czech sentiment lexicon ( <a href="#">Veselovská and Bojar, 2013</a> )
GermanPC	German sentiment lexicon ( <a href="#">Waltinger, 2010</a> )
fullstrength	Spanish sentiment lexicon ( <a href="#">Perez-Rosas et al., 2012</a> )
FEEL	French sentiment lexicon ( <a href="#">Abdaoui et al., 2017</a> )
WHM	English sentiment lexicon; combination of MPQA ( <a href="#">Wilson et al., 2005</a> ), Opinion Lexicon ( <a href="#">Hu and Liu, 2004</a> ) and NRC emotion lexicon ( <a href="#">Mohammad and Turney, 2013</a> )
SE	Semeval 2015 Task 10E shared task data ( <a href="#">Rosenthal et al., 2015</a> )
BWK	English concreteness lexicon ( <a href="#">Bryson et al., 2014</a> )

Table 2: Overview of resources for lexicon induction. The setup is identical to ([Rothe et al., 2016](#)).

### 4 Removing Gender Bias

Word embeddings are well-known for encoding prevalent biases and stereotypes (cf. [Bolukbasi et al. \(2016\)](#)). We demonstrate qualitatively that by identifying an interpretable gender dimension and subsequently removing this dimension, one can remove parts of gender information that potentially could cause biases in downstream processing. Given the original word space  $E$  we consider the interpretable space  $E' := EQ$ , where  $Q$  is computed using DensRay. We denote by  $E_{\cdot, -1} \in \mathbb{R}^{n \times (d-1)}$  the word space with removed first dimension and call it the “complement” space. We expect  $E_{\cdot, -1}$  to be a word space with less gender bias.

To examine this approach qualitatively we use

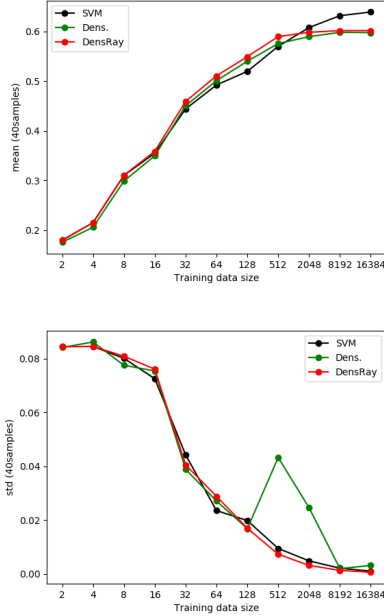


Figure 1: Mean (top) and standard deviation (bottom) of the performance across 40 samples of the training lexicon with varying sizes. Performed on the English concreteness task (line 8 in Table 1). SVR performs similar to SVM and is omitted for clarity.

a list of occupation names<sup>3</sup> by Bolukbasi et al. (2016) and examine the cosine similarities of occupations with the vectors of “man” and “woman”. Figure 2 shows the similarities in the original space  $E$  and debiased space  $E_{,-1}$ . One can see the similarities are closer to the identity (i.e., same distance to “man” and “woman”) in the complement space. To identify occupations with the greatest bias, Table 3 lists occupations for which  $\text{sim}(e_w, e_{\text{man}}) - \text{sim}(e_w, e_{\text{woman}})$  is largest/smallest. One can clearly see a debiasing effect when considering the complement space. Extending this qualitative study to a more rigorous quantitative evaluation is part of future work.

	Original Space		Complement Space			
	man	woman	man	woman		
female bias	actress	0.23	0.46	lawyer	0.16	0.27
	businesswoman	0.32	0.53	ambassador	0.07	0.17
	registered_nurse	0.12	0.33	attorney	0.05	0.15
	housewife	0.34	0.55	legislator	0.26	0.36
	homemaker	0.22	0.40	minister	0.10	0.20
		---				
male bias	hitman	0.41	0.27	captain	0.31	0.24
	gangster	0.34	0.20	marksman	0.29	0.21
	skipper	0.27	0.11	maestro	0.28	0.20
	marksman	0.31	0.14	hitman	0.40	0.32
	maestro	0.30	0.12	skipper	0.25	0.17

Table 3: Top 5 occupations that exhibit the greatest bias (measured by difference in cosine similarity). Numbers indicate cosine similarity between word vectors.

<sup>3</sup> <https://github.com/tolga-b/debiaswe/blob/master/data/professions.json>

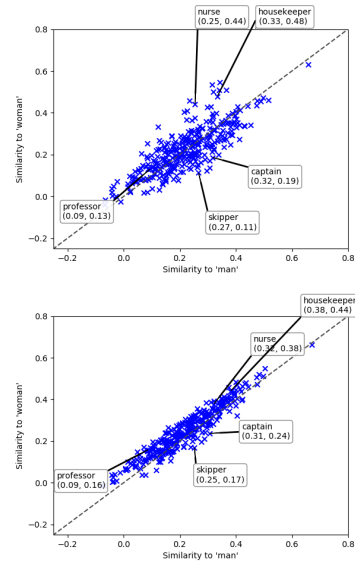


Figure 2: Similarities of occupation vectors with the vectors of man and woman. Top shows the original word space and bottom the word space with removed gender dimension.

## 5 Word Analogy

In this section we use interpretable word spaces for set-based word analogy. Given a list of analogy pairs  $[(a, a'), (b, b'), (c, c'), \dots]$  the task is to predict  $a'$  given  $a$ . Drozd et al. (2016) provide a detailed overview over different methods, and find that their method LRCos performs best.

LRCos assumes two classes: all left elements of a pair (“left class”) and all right elements (“right class”). They train a logistic regression (LR) to differentiate between these two classes. The predicted score of the LR multiplied by the cosine similarity in the word space is their final score. Their prediction for  $a'$  is the word with the highest final score.

We train the classifier on all analogy pairs except for a single pair for which we then obtain the predicted score. In addition we ensure that no word belonging to the test analogy is used during training (splitting the data only on word analogy pairs is not sufficient).

Inspired by LRCos we use interpretable word spaces for approaching word analogy: we train DensRay or an SVM to obtain interpretable embeddings  $E' = EQ$  using the class information as reasoned above. We use a slightly different notation in this section: for a word  $w$  the  $i$ th component of its embedding is given by  $E_{w,i}$ . Therefore we denote as  $E_{,1}$  the first column of  $E'$  (i.e., the

most interpretable dimension). We min-max normalize  $E_{\cdot,1}$  such that words belonging to the right class have a high value (i.e., we flip the sign if necessary). For a query word  $a$  we now want to identify the corresponding  $a'$  by solving

$$\hat{a} = \arg \max_{v \in V} \text{norm}(E_{v,1}) \text{sim}(E_{a,\cdot}, E_{v,\cdot})$$

where  $\text{sim}$  computes the cosine similarity.

Given the result from §4 we extend the above method by computing the cosine similarity in the orthogonal complement, i.e.,  $\text{sim}(E_{a,-1}, E_{v,-1})$ . We call this method **IntCos** (INterpretable, COSine). Depending on the space used for computing the cosine similarity add the word “Original” or “Complement”.

We evaluate this method across two analogy datasets. These are the Google Analogy Dataset (GA) (Mikolov et al., 2013) and BATS (Drozd et al., 2016). As embeddings spaces we use Google News Embeddings (GN) (Mikolov et al., 2013) and FastText subword embeddings (FT) (Bojanowski et al., 2017). We consider the first 80k word embeddings from each space.

Table 4 shows the results. The first observation is that there is no clear winner. IntCos Original performs comparably to LRCos with slight improvements for GN/BATS: here the classes are widespread and exhibit low cosine similarity (IntraR and IntraL), which makes them harder to solve. IntCos Complement maintains performance for GN/BATS and is beneficial for Derivational analogies on GN. For most other analogies it harms performance.

Within IntCos Original it is favorable to use DensRay as it gives slight performance improvements. Especially for harder analogies, where interclass similarity is high and intraclass similarities are low (e.g., in GN/BATS), DensRay outperforms SVMs. In contrast to SVMs, DensRay considers difference vectors *within* classes as well – this seems to be of advantage here.

## 6 Related Work

**Identifying Interpretable Dimensions.** Most relevant to our method is a line of work that uses transformations of existing word spaces to obtain interpretable subspaces. Rothe et al. (2016) compute an orthogonal transformation using shallow neural networks. Park et al. (2017) apply exploratory factor analysis to embedding spaces

	Mean Cosine Sim			Precision				LRCos	
	Inter	IntraL	IntraR	IntCos		original			
				complement	DensR.	SVM	DensR.		SVM
FT/BATS	Inflectional	0.75	0.48	0.51	0.92	0.93	<b>0.97</b>	<b>0.97</b>	<b>0.97</b>
	Derivational	0.63	0.47	0.45	0.74	0.78	<b>0.81</b>	0.80	0.80
	Encyclopedia	0.48	0.43	0.55	0.30	0.43	0.41	0.43	<b>0.45</b>
	Lexicography	0.62	0.37	0.38	0.17	0.20	0.21	0.22	<b>0.26</b>
	Macro Mean	0.62	0.44	0.47	0.53	0.58	0.60	0.60	<b>0.61</b>
Macro Std	0.12	0.06	0.09	0.34	0.33	0.34	0.33	<b>0.32</b>	
GN/BATS	Inflectional	0.63	0.22	0.23	<b>0.88</b>	0.87	<b>0.88</b>	<b>0.88</b>	<b>0.88</b>
	Derivational	0.44	0.21	0.20	<b>0.55</b>	0.50	0.51	0.48	0.44
	Encyclopedia	0.35	0.29	0.42	0.33	<b>0.35</b>	<b>0.35</b>	0.32	0.34
	Lexicography	0.45	0.17	0.18	<b>0.19</b>	0.17	<b>0.19</b>	0.17	0.18
	Macro Mean	0.46	0.22	0.26	<b>0.48</b>	0.47	<b>0.48</b>	0.46	0.45
Macro Std	0.14	0.07	0.12	<b>0.31</b>	<b>0.31</b>	0.32	0.32	0.32	
FT/GA	Micro Mean	0.73	0.48	0.53	0.88	0.91	<b>0.93</b>	0.92	<b>0.93</b>
	Macro Mean	0.71	0.50	0.53	0.87	0.90	<b>0.91</b>	0.90	0.89
	Macro Std	0.11	0.05	0.06	0.11	<b>0.08</b>	0.12	0.17	0.23
GN/GA	Micro Mean	0.62	0.31	0.36	0.85	0.87	<b>0.89</b>	0.87	0.88
	Macro Mean	0.61	0.30	0.35	0.85	0.86	<b>0.88</b>	0.85	0.87
	Macro Std	0.10	0.09	0.10	0.08	<b>0.07</b>	0.09	0.11	0.11

Table 4: Left part shows mean cosine similarity. Inter: mean cosine similarity between pairs. IntraL/R: mean cosine similarity within the left/right class. Right part shows precision for word analogy task.

to obtain interpretable dimensions in an unsupervised manner. Their approach relies on solving complex optimization problems, while we focus on closed form solutions. Senel et al. (2018) use SEMCAT categories in combination with the Bhattacharya distance to identify interpretable directions. Also, oriented PCA (Diamantaras and Kung, 1996) is closely related to our method. However, both methods yield non-orthogonal transformation. Faruqui et al. (2015a) use semantic lexicons to retrofit embedding spaces. Thus they do not fully maintain the structure of the word space, which is in contrast to this work.

**Interpretable Embedding Algorithms.** Another line of work modifies embedding algorithms to yield interpretable dimensions (Koç et al., 2018; Luo et al., 2015; Shin et al., 2018; Zhao et al., 2018). There is also much work that generates sparse embeddings that are claimed to be more interpretable (Murphy et al., 2012; Faruqui et al., 2015b; Fyshe et al., 2015; Subramanian et al., 2018). Instead of learning new embeddings, we aim at making dense embeddings interpretable.

## 7 Conclusion

We investigated analytical methods for obtaining interpretable word embedding spaces. Relevant methods were examined with the tasks of lexicon induction, word analogy and debiasing.

We gratefully **acknowledge** funding through a Zentrum Digitalisierung.Bayern fellowship awarded to the first author. This work was supported by the European Research Council (# 740516).

## References

- Amine Abdaoui, Jérôme Azé, Sandra Bringay, and Pascal Poncelet. 2017. Feel: a french expanded emotion lexicon. *Language Resources and Evaluation*, 51(3).
- Silvio Amir, Ramón Astudillo, Wang Ling, Bruno Martins, Mario J Silva, and Isabel Trancoso. 2015. Inesc-id: A regression model for large scale twitter sentiment lexicon induction. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*.
- Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*.
- 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*.
- Marc Brysbaert, Amy Beth Warriner, and Victor Kuperman. 2014. Concreteness ratings for 40 thousand generally known english word lemmas. *Behavior research methods*, 46(3).
- Konstantinos I Diamantaras and Sun Yuan Kung. 1996. *Principal component neural networks: theory and applications*, volume 5. Wiley New York.
- Aleksandr Drozd, Anna Gladkova, and Satoshi Matsuo. 2016. Word embeddings, analogies, and machine learning: Beyond king-man+ woman= queen. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*.
- Manaal Faruqui, Jesse Dodge, Sujay Kumar Jauhar, Chris Dyer, Eduard Hovy, and Noah A Smith. 2015a. 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*.
- Manaal Faruqui, Yulia Tsvetkov, Dani Yogatama, Chris Dyer, and Noah A Smith. 2015b. Sparse overcomplete word vector representations. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*.
- Jean-Philippe Fauconnier. 2015. [French word embeddings](#).
- Alona Fyshe, Leila Wehbe, Partha P Talukdar, Brian Murphy, and Tom M Mitchell. 2015. A compositional and interpretable semantic space. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.
- Roger A Horn, Roger A Horn, and Charles R Johnson. 1990. *Matrix analysis*. Cambridge university press.
- Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*.
- Aykut Koç, Ihsan Utlü, Lutfi Kerem Senel, and Haldun M Ozaktas. 2018. Imparting interpretability to word embeddings. *arXiv preprint arXiv:1807.07279*.
- Hongyin Luo, Zhiyuan Liu, Huanbo Luan, and Maosong Sun. 2015. Online learning of interpretable word embeddings. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*.
- Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.
- Saif M Mohammad and Peter D Turney. 2013. Crowdsourcing a word–emotion association lexicon. *Computational Intelligence*, 29(3).
- Brian Murphy, Partha Talukdar, and Tom Mitchell. 2012. Learning effective and interpretable semantic models using non-negative sparse embedding. *Proceedings of the 24th International Conference on Computational Linguistics*.
- Sungjoon Park, JinYeong Bak, and Alice Oh. 2017. Rotated word vector representations and their interpretability. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*.
- Veronica Perez-Rosas, Carmen Banea, and Rada Mihalcea. 2012. Learning sentiment lexicons in spanish. In *Proceedings of the seventh international conference on Language Resources and Evaluation*.
- Sara Rosenthal, Preslav Nakov, Svetlana Kiritchenko, Saif Mohammad, Alan Ritter, and Veselin Stoyanov. 2015. Semeval-2015 task 10: Sentiment analysis in twitter. In *Proceedings of the 9th international workshop on semantic evaluation*.
- 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*.
- Lutfi Kerem Senel, Ihsan Utlü, Veysel Yucesoy, Aykut Koc, and Tolga Cukur. 2018. Semantic structure and interpretability of word embeddings. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.
- Jamin Shin, Andrea Madotto, and Pascale Fung. 2018. Interpreting word embeddings with eigenvector analysis. *openreview.net*.

- Anant Subramanian, Danish Pruthi, Harsh Jhamtani, Taylor Berg-Kirkpatrick, and Eduard Hovy. 2018. Spine: Sparse interpretable neural embeddings. In *Thirty-Second AAAI Conference on Artificial Intelligence*.
- Kateřina Veselovská and Ondřej Bojar. 2013. Czech sublex 1.0. Charles University, Faculty of Mathematics and Physics.
- Ulli Waltinger. 2010. Germanpolarityclues: A lexical resource for german sentiment analysis. In *Proceedings of the seventh international conference on Language Resources and Evaluation*.
- Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*.
- Jieyu Zhao, Yichao Zhou, Zeyu Li, Wei Wang, and Kai-Wei Chang. 2018. Learning gender-neutral word embeddings. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*.
- Mengjie Zhao and Hinrich Schütze. 2019. A multilingual bpe embedding space for universal sentiment lexicon induction. In *Proceedings of the 57th Conference of the Association for Computational Linguistics*.