Word Embedding Calculus in Meaningful Ultradense Subspaces

Sascha Rothe and **Hinrich Schütze**
Center for Information and Language Processing
LMU Munich, Germany
sascha@cis.lmu.de

Abstract

We decompose a standard embedding space into interpretable orthogonal subspaces and a "remainder" subspace. We consider four interpretable subspaces in this paper: polarity, concreteness, frequency and part-of-speech (POS) subspaces. We introduce a new calculus for subspaces that supports operations like "$-1 \times$ *hate* $=$ *love*" and "give me a neutral word for *greasy*" (i.e., *oleaginous*). This calculus extends analogy computations like "*king*$-$*man*$+$*woman* $=$ *queen*". For the tasks of Antonym Classification and POS Tagging our method outperforms the state of the art. We create test sets for Morphological Analogies and for the new task of Polarity Spectrum Creation.

1 Introduction

Word embeddings are usually trained on an objective that ensures that words occurring in similar contexts have similar embeddings. This makes them useful for many tasks, but has drawbacks for others; e.g., antonyms are often interchangeable in context and thus have similar word embeddings even though they denote opposites. If we think of word embeddings as members of a (commutative or Abelian) group, then antonyms should be *inverses* of (as opposed to *similar* to) each other. In this paper, we use DENSIFIER (Rothe et al., 2016) to decompose a standard embedding space into interpretable orthogonal subspaces, including a one-dimensional polarity subspace as well as concreteness, frequency and POS subspaces. We introduce a new calculus for subspaces in which antonyms are inverses, e.g., "$-1 \times$ *hate* $=$ *love*". The formula shows what happens in the polarity subspace; the orthogonal complement (all the remaining subspaces) is kept fixed. We show below that we can predict an entire polarity spectrum based on the subspace, e.g., the four-word spectrum *hate*, *dislike*, *like*, *love*. Similar to polarity, we explore other interpretable subspaces and do operations such as: given a concrete word like *friend* find the abstract word *friendship* (concreteness); given the frequent word *friend* find a less frequent synonym like *comrade* (frequency); and given the noun *friend* find the verb *befriend* (POS).

2 Word Embedding Transformation

We now give an overview of DENSIFIER; see Rothe et al. (2016) for details. Let $Q \in \mathbb{R}^{d \times d}$ be an orthogonal matrix that transforms the original word embedding space into a space in which certain types of information are represented by a small number of dimensions. The orthogonality can be seen as a hard regularization of the transformation. We choose this because we do not want to add or remove any information from the original embeddings space. This ensures that the transformed word embeddings behave differently only when looking at subspaces, but behave identically when looking at the entire space. By choosing an orthogonal and thus linear transformation we also assume that the information is already encoded linearly in the original word embedding. This is a frequent assumption, as we generally use the vector addition for word embeddings.

Concretely, we learn Q such that the dimensions $D^p \subset \{1, \ldots, d\}$ of the resulting space correspond to a word's polarity information and the $\{1, \ldots, d\} \setminus D^p$ remaining dimensions correspond to non-polarity information. Analogously, the sets of dimensions D^c, D^f and D^m correspond to a word's concreteness, frequency and POS (or morphological) information, respectively. In this paper, we assume that these properties do not corre-

Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, pages 512–517,
Berlin, Germany, August 7-12, 2016. ©2016 Association for Computational Linguistics

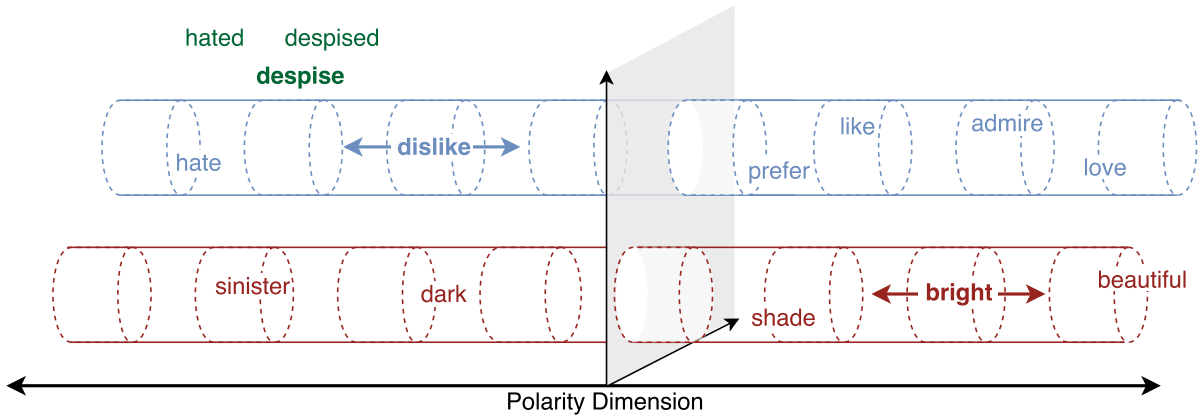

Figure 1: Illustration of the transformed embeddings. The horizontal axis is the polarity subspace. All non-polarity information, including concreteness, frequency and POS, is projected into a two dimensional subspace for visualization (gray plane). A query word (bold) specifies a line parallel to the horizontal axis. We then construct a cylinder around this line. Words in this cylinder are considered to be part of the word spectrum.

late and therefore the ultradense subspaces do not overlap. E.g., $D^p \cap D^c = \emptyset$. This might not be true for other settings, e.g., sentiment and semantic information. As we are using only four properties there is also a subspace which is in the orthogonal complement of all trained subspaces. This subspace includes the not classified information, e.g., genre information in our case (e.g., "clunker" is a colloquial word for "automobile").

If $e_v \in \mathbb{R}^d$ is the original embedding of word v, the transformed representation is $u_v = Qe_v$. We use $*$ as a placeholder for polarity (p), concreteness (c), frequency (f) and POS/morphology (m) and call $d^* = |D^*|$ the dimensionality of the ultradense subspace of property $*$. For each ultradense subspace, we create $P^* \in \mathbb{R}^{d^* \times d}$, an identity matrix for the dimensions in D^*. Thus, the ultradense (UD) representation $u_v^* \in \mathbb{R}^{d^*}$ of word v is defined as:

$$u_v^* := P^* Q e_v \tag{1}$$

For notational simplicity, u_v^* will either refer to a vector in $\mathbb{R}^{d^*}$ or to a vector in $\mathbb{R}^d$ where all dimensions $\notin D^*$ are set to zero.

For training, the orthogonal transformation Q we assume we have a lexicon resource. Let $L^*_{\not\sim}$ be a set of word index pairs (v, w) with different labels, e.g., positive/negative, concrete/abstract or noun/verb. We want to maximize the distance for pairs in this group. Thus, our objective is:

$$\underset{Q}{\operatorname{argmin}} \sum_{*\in\{p,c,f,m\}} \sum_{(v,w)\in L^*_{\not\sim}} -\|P^*Q(e_v - e_w)\| \tag{2}$$

subject to Q being an orthogonal matrix. Another goal is to minimize the distance of two words with identical labels. Let $L^*_{\sim}$ be a set of word index pairs (v, w) with identical labels. In contrast to Eq. 2, we now want to minimize each distance. Thus, the objective is given by:

$$\underset{Q}{\operatorname{argmin}} \sum_{*\in\{p,c,f,m\}} \sum_{(v,w)\in L^*_{\sim}} \|P^*Q(e_v - e_w)\| \tag{3}$$

subject to Q being an orthogonal matrix. For training Eq. 2 is weighted with α^* and Eq. 3 with $1 - \alpha^*$. We do a batch gradient descent where each batch contains the same number of positive and negative examples. This means the number of examples in the lexica – which give rise to more negative than positive examples – does not influence the training.

3 Setup and Method

Eqs. 2/3 can be combined to train an orthogonal transformation matrix. We use pretrained 300-dimensional English word embeddings (Mikolov et al., 2013) (W2V). To train the transformation matrix, we use a combination of MPQA (Wilson et al., 2005), Opinion Lexicon (Hu and Liu, 2004) and NRC Emotion lexicons (Mohammad and Turney, 2013) for polarity; BWK, a lexicon of 40,000 English words (Brysbaert et al., 2014), for concreteness; the order in the word embedding file for frequency; and the training set of the FLORS tagger (Schnabel and Schütze, 2014) for POS. The application of the transformation ma-

trix to the word embeddings gives us four subspaces for polarity, concreteness, frequency and POS. These subspaces and their orthogonal complements are the basis for an embedding calculus that supports certain operations. Here, we investigate four such operations. The first operation computes the antonym of word v:

$$\text{antonym}(v) = \text{nn}(u_v - 2u_v^p) \quad (4)$$

where nn : $\mathbb{R}^d \rightarrow V$ returns the word whose embedding is the nearest neighbor to the input. Thus, our hypothesis is that antonyms are usually very similar in semantics except that they differ on a single "semantic axis," the polarity axis.[1] The second operation is "neutral version of word v":

$$\text{neutral}(v) = \text{nn}(u_v - u_v^p) \quad (5)$$

Thus, our hypothesis is that neutral words are words with a value close to zero in the polarity subspace. The third operation produces the polarity spectrum of v:

$$\text{spectrum}(v) = \{\text{nn}(u_v + xu_v^p) \mid \forall x \in \mathbb{R}\} \quad (6)$$

This means that we keep the semantics of the query word fixed, while walking along the polarity axis, thus retrieving different shades of polarity. Figure 1 shows two example spectra. The fourth operation is "word v with POS of word w":

$$\text{POS}_w(v) = \text{nn}(u_v - u_v^m + u_w^m) \quad (7)$$

This is similar to analogies like *king* $-$ *man* $+$ *woman*, except that the analogy is *inferred by the subspace relevant for the analogy*.

We create word spectra for some manually chosen words using the Google News corpus (W2V) and a Twitter corpus. As the transformation was orthogonal and therefore did not change the length of a dimension, we multiply the polarity dimension with 30 to give it a high weight, i.e., paying more attention to it. We then use Eq. 6 with a sufficiently small step size for x, i.e., further reducing the step size does not increase the spectrum. We also discard words that have a cosine distance of more than .6 in the non-polarity space. Table 1 shows examples. The results are highly domain dependent, with Twitter's spectrum indicating more negative views of politicians than news. While *fall* has negative associations, *autumn*'s are positive – probably because *autumn* is of a higher register in American English.

[1] See discussion/experiments below for exceptions

Corpus, Type	Spectrum
News, Polarity	hypocrite, **politician**, legislator, businessman, reformer, statesman, thinker
	fall, winter, summer, **spring**, autumn
	drunks, booze, liquor, lager, **beer**, beers, wine, beverages, wines, tastings
Twitter, Polarity	corrupt, coward, **politician**, journalist, citizen, musician, representative
	stalker, neighbour, gf, bf, cousin, frnd, **friend**, mentor
	#stupid, #problems, #homework, #mylife, #reality, **#life**, #happiness
News, Concreteness	imperialist, conflict, **war**, Iraq, Vietnam War, battlefields, soldiers
	love, friendship, dear friend, friends, **friend**, girlfriend
News, Frequency	redesigned, newer, revamped, **new**
	intellect, insights, familiarity, skills, **knowledge**, experience

Table 1: Example word spectra for polarity, concreteness and frequency on two different corpora. Queries are bold.

	dev set			test set		
	P	R	F_1	P	R	F_1
Adel, 2014	.79	.65	.72	.75	.58	.66
our work	.81	.90	.85	.76	.88	.82

Table 2: Results for Antonym Classification

4 Evaluation

4.1 Antonym Classification.

We evaluate on Adel and Schütze (2014)'s data; the task is to decide for a pair of words whether they are antonyms or synonyms. The set has 2,337 positive and negative pairs each and is split into 80% training, 10% dev and 10% test. Adel and Schütze (2014) collected positive/negative examples from the nearest neighbors of the word embeddings to make it hard to solve the task using word embeddings. We train an SVM (RBF kernel) on three features that are based on the intuition depicted in Figure 1: the three cosine distances in: the polarity subspace; the orthogonal complement; and the entire space. Table 2 shows that improvement of precision is minor (.76 vs. .75), but recall and F_1 improve by a lot (+.30 and +.16).

4.2 Polarity Spectrum Creation

consists of two subtasks. PSC-SET: Given a query word how well can we predict a spectrum? PSC-ORD: How good is the order in the spectrum? Our gold standard is Word Spectrum, included in the Oxford American Writer's Thesaurus (OAWT) and therefore also in MacOS. For each query word

	newsgroups		reviews		weblogs		answers		emails		wsj	
	ALL	OOV	ALL	OOV	ALL	OOV	ALL	OOV	ALL	OOV	ALL	OOV
1 LSJU	89.11†	56.02†	91.43†	58.66†	94.15†	77.13†	88.92†	49.30†	88.68†	58.42†	96.83	90.25
2 SVM	89.14†	53.82†	91.30†	54.20†	94.21†	76.44†	88.96†	47.25†	88.64†	56.37†	96.63	87.96†
3 F	**90.86**	66.42†	**92.95**	75.29†	94.71	83.64†	90.30	62.15†	89.44	62.61†	96.59	90.37
4 F+W2V	90.51	**72.26**	92.46†	78.03	94.70	86.05	90.34	65.16	89.26	63.70†	96.44	91.36
5 F+UD	90.79	72.20	92.84	**78.80**	**94.84**	**86.47**	**90.60**	**65.48**	**89.68**	**66.24**	**96.61**	**92.36**

Table 3: Results for POS tagging. LSJU = Stanford. SVM = SVMTool. F=FLORS. We show three state-of-the-art taggers (lines 1-3), FLORS extended with 300-dimensional embeddings (4) and extended with UD embeddings (5). †: significantly better than the best result in the same column ($\alpha = .05$, one-tailed Z-test).

this dictionary returns a list of up to 80 words of shades of meaning between two polar opposites. We look for words that are also present in Adel and Schütze (2014)'s Antonym Classification data and retrieve 35 spectra. Each word in a spectrum can be used as a query word; after intersecting the spectra with our vocabulary, we end up with 1301 test cases.

To evaluate PSC-SET, we calculate the 10 nearest neighbors of the m words in the spectrum and rank the $10m$ neighbors by the distance to our spectrum, i.e., the cosine distance in the orthogonal complement of the polarity subspace. We report mean average precision (MAP) and weighted MAP where each MAP is weighted by the number of words in the spectrum. As shown in Table 4 there is no big difference between both numbers, meaning that our algorithm does not work better or worse on smaller or larger spectra.

To evaluate PSC-ORD, we calculate Spearman's ρ of the ranks in OAWT and the values on the polarity dimension. Again, there is no significant difference between average and weighted average of ρ. Table 4 also shows that the variance of the measures is low for PSC-SET and high for PSC-ORD; thus, we do well on certain spectra and worse on others. The best one, *beautiful* $\leftrightarrow$ *ugly*, is given as an example. The worst performing spectrum is *fat* $\leftrightarrow$ *skinny* ($\rho = .13$) – presumably because both extremes are negative, contradicting our modeling assumption that spectra go from positive to negative. We test this hypothesis by separating the spectrum into two subspectra. We then report the weighted average correlation of the optimal separation. For *fat* $\leftrightarrow$ *skinny*, this improves ρ to .67.

	PSC-SET: MAP	PSC-ORD: ρ	avg(ρ_1, ρ_2)
average	.48	.59	.70
weighted avg.	.47	.59	.70
variance	.004	.048	.014
beautiful/ugly	.48	.84	.84
fat/skinny	.56	.13	.67
absent/present	.43	.72	.76

Table 4: Results for Polarity Spectrum Creation: MAP, Spearman's ρ (one spectrum) and average ρ (two subspectra)

4.3 Morphological Analogy.

The previous two subspaces were one-dimensional. Now we consider a POS subspace, because POS is not one-dimensional and cannot be modeled as a single scalar quantity. We create a word analogy benchmark by extracting derivational forms from WordNet (Fellbaum, 1998). We discard words with $\geq$2 derivational forms of the same POS and words not in the most frequent 30,000. We then randomly select 26 pairs for every POS combination for the dev set and 26 pairs for the test set.[2] An example of the type of equation we solve here is *prediction* $-$ *predict* $+$ *symbolize* $=$ *symbol* (from the dev set). W2V embeddings are our baseline.

We can also rewrite the left side of the equation as POS(*prediction*) $+$ Semantics(*symbolize*); note that this cannot be done using standard word embeddings. In contrast, our method can use meaningful UD embeddings and Eq. 7 with POS(v) being u_v^m and Semantics(v) being $u_v - u_v^m$. The dev set indicates that a 8-dimensional POS subspace is optimal and Table 5 shows that this method out-

[2]This results in an even number of $25 * 26 = 650$ questions per POS combination, $4{*}2{*}650 = 5200$ in total (4 POS combinations, where each POS can be used as query POS).

	W2V		UD	
	A→B	B→A	A→B	B→A
noun-verb	35.69	6.62	59.69†	50.46†
adj-noun	30.77	27.38	53.85†	43.85†
adj-verb	20.62	3.08	32.15†	24.77†
adj-adverb	45.38	35.54	46.46	43.08†
all	25.63		44.29†	

Table 5: Accuracy @1 on test for Morphological Analogy. †: significantly better than the corresponding result in the same row ($\alpha = .05$, one-tailed Z-test).

performs the baseline.

4.4 POS Tagging

Our final evaluation is extrinsic. We use FLORS (Schnabel and Schütze, 2014), a state-of-the-art POS tagger which was extended by Yin et al. (2015) with word embeddings as additional features. W2V gives us a consistent improvement on OOVs (Table 3, line 4). However, training this model requires about 500GB of RAM. When we use the 8-dimensional UD embeddings (the same as for Morphological Analogy), we outperform W2V except for a virtual tie on news (Table 3, line 5). So we perform better even though we only use 8 of 300 dimensions! However, the greatest advantage of UD is that we only need 100GB of RAM, 80% less than W2V.

5 Related Work

Yih et al. (2012) also tackled the problem of antonyms having similar embeddings. In their model, the antonym is the inverse of the entire vector whereas in our work the antonym is only the inverse in an ultradense subspace. Our model is more intuitive since antonyms invert only part of the meaning, not the entire meaning. Schwartz et al. (2015) present a method that switches an antonym parameter on or off (depending on whether a high antonym-synonym similarity is useful for an application) and learn *multiple* embedding spaces. We only need a *single* space, but consider different subspaces of this space.

An unsupervised approach using linguistic patterns that ranks adjectives according to their intensity was presented by de Melo and Bansal (2013). Sharma et al. (2015) present a corpus-independent approach for the same problem. Our results (Table 1) suggest that polarity should not be considered to be corpus-independent.

There is also much work on incorporating the additional information into the original word embedding training. Examples include (Botha and Blunsom, 2014) and (Cotterell and Schütze, 2015). However, postprocessing has several advantages. DENSIFIER can be trained on a normal work station without access to the original training corpus. This makes the method more flexible, e.g., when new training data or desired properties are available.

On a general level, our method bears some resemblance with (Weinberger and Saul, 2009) in that we perform supervised learning on a set of desired (dis)similarities and that we can think of our method as learning specialized metrics for particular subtypes of linguistic information or particular tasks. Using the method of Weinberger and Saul (2009), one could learn k metrics for k subtypes of information and then simply represent a word w as the concatenation of (i) the original embedding and (ii) k representations corresponding to the k metrics.[3] In case of a simple one-dimensional type of information, the corresponding representation could simply be a scalar. We would expect this approach to have similar advantages for practical applications, but we view our orthogonal transformation of the original space as more elegant and it gives rise to a more compact representation.

6 Conclusion

We presented a new word embedding calculus based on meaningful ultradense subspaces. We applied the operations of the calculus to Antonym Classification, Polarity Spectrum Creation, Morphological Analogy and POS Tagging. Our evaluation shows that our method outperforms previous work and is applicable to different types of information. We have published test sets and word embeddings at `http://www.cis.lmu.de/~sascha/Ultradense/`.

Acknowledgments

This research was supported by Deutsche Forschungsgemeinschaft (DFG, grant 2246/10-1).

[3] We would like to thank an anonymous reviewer for suggesting this alternative approach.

References

Heike Adel and Hinrich Schütze. 2014. Using mined coreference chains as a resource for a semantic task. In *Proceedings of EMNLP*.

Jan A Botha and Phil Blunsom. 2014. Compositional morphology for word representations and language modelling. *arXiv preprint arXiv:1405.4273*.

Marc Brysbaert, Amy Beth Warriner, and Victor Kuperman. 2014. Concreteness ratings for 40 thousand generally known english word lemmas. *Behavior research methods*, 46(3):904–911.

Ryan Cotterell and Hinrich Schütze. 2015. Morphological word-embeddings. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1287–1292, Denver, Colorado, May–June. Association for Computational Linguistics.

Gerard de Melo and Mohit Bansal. 2013. Good, great, excellent: Global inference of semantic intensities. *Transactions of the Association for Computational Linguistics*, 1:279–290.

Christiane Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. Bradford Books.

Minqing Hu and Bing Liu. 2004. Mining and Summarizing Customer Reviews. In *KDD*.

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).

Sascha Rothe, Sebastian Ebert, and Hinrich Schütze. 2016. Ultradense word embeddings by orthogonal transformation. *arXiv preprint arXiv:1602.07572*.

Tobias Schnabel and Hinrich Schütze. 2014. Flors: Fast and simple domain adaptation for part-of-speech tagging. *Transactions of the Association for Computational Linguistics*, 2:15–26.

Roy Schwartz, Roi Reichart, and Ari Rappoport. 2015. Symmetric pattern based word embeddings for improved word similarity prediction. In *Proceedings of CoNLL*.

Raksha Sharma, Mohit Gupta, Astha Agarwal, and Pushpak Bhattacharyya. 2015. Adjective intensity and sentiment analysis. In *Proceedings of EMNLP*.

Kilian Q. Weinberger and Lawrence K. Saul. 2009. Distance metric learning for large margin nearest neighbor classification. *J. Mach. Learn. Res.*, 10:207–244.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of HLT/EMNLP*.

Wen-tau Yih, Geoffrey Zweig, and John C Platt. 2012. Polarity inducing latent semantic analysis. In *Proceedings of EMNLP*.

Wenpeng Yin, Tobias Schnabel, and Hinrich Schütze. 2015. Online updating of word representations for part-of-speech tagging. In *Proceedings of EMNLP*.