Explaining and Generalizing Skip-Gram through Exponential Family Principal Component Analysis

Ryan Cotterell **Adam Poliak** **Benjamin Van Durme** **Jason Eisner**
Center for Language and Speech Processing
Johns Hopkins University
{ryan.cotterell,azpoliak,vandurme,jason}@cs.jhu.edu

Abstract

The popular skip-gram model induces word embeddings by exploiting the signal from word-context coocurrence. We offer a new interpretation of skip-gram based on exponential family PCA—a form of matrix factorization to generalize the skip-gram model to *tensor* factorization. In turn, this lets us train embeddings through richer higher-order coocurrences, e.g., triples that include positional information (to incorporate syntax) or morphological information (to share parameters across related words). We experiment on 40 languages and show our model improves upon skip-gram.

1 Introduction

Over the past years NLP has witnessed a veritable frenzy on the topic of word embeddings—low-dimensional representations of distributional information. The embeddings, trained on extremely large text corpora, e.g., Wikipedia and the Common Crawl, are claimed to encode semantic knowledge extracted from large text corpora. While numerous models have been proposed in the literature, the signal for learning the embeddings is typically a bag of contexts associated with each word type. The most popular models for this low-dimensional embedding are skip-gram (Mikolov et al., 2013) and GloVe (Pennington et al., 2014). Natural language text, however, contains richer structure than simple context-word pairs. In this work, we probe embedding n-tuples, allowing us to escape the bag-of-words assumption by encoding richer linguistic structures.

As a first step, we offer a novel interpretation of the skip-gram model (Mikolov et al., 2013). We show how skip-gram can be viewed as an application of exponential family principal components analysis (EPCA) (Collins et al., 2001) to an integer matrix of coocurrence counts. Previous work has related the negative sampling *estimator* for skip-gram model parameters to the factorization of a matrix of (shifted) positive pointwise mutual information (Levy and Goldberg, 2014b). We show the skip-gram *objective* is just EPCA factorization.

This EPCA factorization leads to the natural extension of tensor factorization and enables the model to move beyond skip-gram's bag-of-words assumptions by capturing richer linguistic structures. In this paper, we explore incorporating positional and morphological content in the model by factorizing a positional tensor and morphology tensor. The positional tensor directly incorporates word order into the model, and the morphology tensor adds word-internal information. We validate our models experimentally on 40 languages and show large gains under standard metrics.[1]

2 Matrix Factorization

To show the equivalence between the skip-gram model and EPCA, we briefly review the latter. Given a matrix $X \in \mathbb{R}^{n_1 \times n_2}$, where X_{ij} is the number of times word i appears in context j under some user-specified definition of "context." Vanilla PCA (Pearson, 1901) minimizes

$$\left|\left|X - CW^\top\right|\right|_{\text{F}}^2 = \sum_{ij} (X_{ij} - c_i \cdot w_j)^2 \quad (1)$$

$$= \sum_j ||X_{\cdot j} - Cw_j||^2 \quad (2)$$

by choosing matrices $C \in \mathbb{R}^{n_1 \times d}$ and $W \in \mathbb{R}^{n_2 \times d}$, whose rows are d-dimensional vectors that embed the contexts and the words, respectively. c_i and w_j denote the *column* vectors formed by the i^{th}

[1]The code developed is available at https://github.com/azpoliak/skip-gram-tensor

row of C and j^{th} row of W. Note that $CW^\top$ is an approximate factorization of X with rank $\leq d$. Globally optimizing Eq. (1) means finding the *best* rank $\leq d$ approximation to X (Eckart and Young, 1936), and can be done by SVD (Golub and Van Loan, 2012). Both Roweis (1997) and Tipping and Bishop (1999) interpreted Eq. (2) as the maximum-likelihood estimate of a certain Gaussian graphical model (drawn in Fig. 1a), which generates the column vector $X_{\cdot j} \sim \mathcal{N}(Cw_j, I)$.[2]

EPCA is a generalization of PCA, where the Gaussian family is replaced by any other exponential family of distributions over vectors. Our point is that skip-gram is precisely *multinomial EPCA with the canonical link function* (Mohamed, 2011), which generates the vector $X_{\cdot j}$ of integer counts from a multinomial with log-linear parameterization.[3] That is, skip-gram chooses embeddings that maximize a different log-likelihood

$$\sum_j \sum_i X_{ij} \log p(\mathtt{c}_i \mid \mathtt{w}_j), \tag{3}$$

where

$$p(\mathtt{c}_i \mid \mathtt{w}_j) = \frac{\exp(c_i \cdot w_j)}{\sum_{i'} \exp(c_{i'} \cdot w_j)}. \tag{4}$$

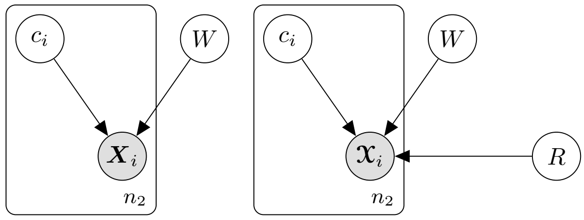
The typewriter-styled names ($\mathtt{c}_i$ and $\mathtt{w}_j$) denote the i^{th} context type and j^{th} word type, whereas c_i and w_j denote the embeddings of those types in $\mathbb{R}^d$.

Relation to Levy and Goldberg (2014). Levy and Goldberg (2014b) also interpreted skip-gram as matrix factorization. Specifically, they argued that the skip-gram estimation *by negative sampling* implicitly factorizes a shifted matrix of positive empirical pointwise mutual information values. We instead regard the skip-gram objective as demanding EPCA-style factorization of the matrix X: that is, X was generated stochastically from some unknown matrix of log-linear parameters (where column j of X is generated from column j of the parameter matrix), and we seek a rank-d estimate $CW^\top$ of *that* matrix. pLSI (Hofmann, 1999) is similar but factors an unknown matrix of multinomial probabilities, which is *multinomial EPCA with the identity link function*.

Our EPCA interpretation applies equally well to the component distributions that are used in hierarchical softmax (Morin and Bengio, 2005), which is an alternative to negative sampling. Additionally, it yields avenues of future research using Bayesian (Mohamed et al., 2008) and maximum-margin (Srebro et al., 2004) extensions to EPCA.

[2]Singh and Gordon (2008) offer a comprehensive discussion of PCA and other matrix factorization techniques in ML.

[3]The embeddings are real numbers rather than counts, so we continue to impose Gaussian priors on *them*, yielding an ℓ_2-regularized maximum likelihood estimate. Regularization has only minor effects with large training corpora, and is not in the original `word2vec` implementation of skip-gram.



(a) Matrix factorization (b) Tensor factorization

Figure 1: Comparison of the graphical model for matrix factorization (either PCA or EPCA) and 3-dimensional tensor factorization. Priors are omitted from the drawing.

3 Tensor Factorization

In contrast to matrix factorization, there are several distinct definitions of tensor factorization (Kolda and Bader, 2009). We focus on the polyadic decomposition (Hitchcock, 1927), which yields a satisfying generalization—the resulting graphical model is displayed in Fig. 1b. The tensor analogue to PCA is

$$\begin{aligned} ||\mathcal{X} - C \;{}_2\!\otimes_2 W \;{}_2\!\otimes_2 R||_F^2 &= \sum_{ijk} (\mathcal{X}_{ijk} - \mathbf{1} \cdot (c_i \odot w_j \odot r_k))^2 \\ &= \sum_{jk} ||X_{\cdot jk} - C(w_j \odot r_k)||^2, \end{aligned} \tag{5}$$

where the new matrix $R \in \mathbb{R}^{n_3 \times d}$ holds embeddings for *relations* between the contexts and words. Given a tensor $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, this objective attempts to predict each entry as the three-way dot product of the vectors $c_i, w_j, r_k \in \mathbb{R}^d$, thus factorizing $\mathcal{X}$ into C, W, R. The polyadic decomposition can be viewed as a Tucker decomposition (Tucker, 1966) that enforces a diagonal core.

In our setting, $\mathcal{X}$ is a collection of *count vectors* in $\mathbb{N}^{n_1}$ generated from $n_2 \times n_3$ multinomial distributions, so we now move from third-order PCA to third-order EPCA. Our higher-order skip-gram (HOSG) attempts to maximize the log-likelihood

$$\sum_{ijk} \mathcal{X}_{ijk} \log p(\mathtt{c}_i \mid \mathtt{w}_j, \mathtt{r}_k), \tag{6}$$

where we define the model p as

$$p(\mathtt{c}_i \mid \mathtt{w}_j, \mathtt{r}_k) = \frac{\exp(\mathbf{1} \cdot (c_i \odot w_j \odot r_k))}{\sum_{i'} \exp(\mathbf{1} \cdot (c_{i'} \odot w_j \odot r_k))}. \tag{7}$$

Approximate Learning. We locally optimize the parameters of our probability model—the word, context and relation embeddings—through stochastic gradient ascent (Robbins and Monro, 1951) on (6). Each stochastic gradient step computes $\log p(c_i \mid w_j, r_k)$ and its gradient for some (i, j, k) triple, which unfortunately requires summing over n_1 contexts in the denominator of (7). This is problematic as n_1 is often very large, e.g., 10^7. As speedups, Mikolov et al. (2013) offer two schemes: negative sampling and hierarchical softmax. Here we apply the negative sampling approximation to HOSG, but hierarchical softmax is also applicable. We direct the reader to Goldberg and Levy (2014) for an in-depth discussion.

4 Two Tensors for Word Embedding

There are many tensors that one could factorize with our approach. As examples, we offer two third-order generalizations of Mikolov et al. (2013)'s context-word matrix. We are still predicting the distribution of contexts of a given word type. Our first version *increases* the number of parameters (giving more expressivity) by conditioning on additional information. Our second version *decreases* the number of parameters (giving better smoothing) by factoring the word type.

4.1 Positional Tensor

When predicting the context words in a window around a given word token, Mikolov et al. (2013) uses the same distribution to predict each of them. We propose to use different distributions at different positions in the window; we define a "positional tensor": $\mathcal{X}_{\langle \texttt{dog},\texttt{ran},-2\rangle}$ is the number of times the context word `dog` was seen two positions to the left of `ran`. We will predict this count using $p(\texttt{dog} \mid \texttt{ran}, -2)$, defined from the embeddings of the word `ran`, the position -2, and the context word `dog` and its competitors. For a 10-word window, we have $\mathcal{X} \in \mathbb{R}^{|V|\times|V|\times 10}$. The incorporation of word position into the tensor should improve syntactic awareness.

4.2 Compositional Morphology Tensor

For Mikolov et al. (2013), related words such as `ran` and `running` are monolithic objects that do not share parameters. We decompose each word into a lemma and a morphological tag. This is a relaxation of the bag of words to a bag of morphemes. Thus, we predict the count $\mathcal{X}_{\langle \texttt{dog},\text{RUN},t\rangle}$ using $p(\texttt{dog} \mid \text{RUN}, t)$, where t is a morphological tag such as [pos=V,tense=PAST]. Our model is essentially a version of Mikolov et al. (2013) that parameterizes the embedding of the word `ran` as a Hadamard product $w_j \odot r_k$, where w_j embeds RUN and r_k embeds tag t. Where Cotterell et al. (2016) used additive embeddings such as $w_j + r_k$, our approach is more flexible, since $c_i \cdot (w_j + r_k)$ can be expressed using a Hadamard product in lieu of addition, as $(c_i; c_i) \cdot ((w_j; \mathbf{1}) \odot (\mathbf{1}; r_k))$ (using twice as many dimensions to embed each object).

5 Experiments

We build HOSG on top of the HYPERWORDS package [4]. All models (both skip-gram and higher-order skip-gram) are trained for 10 epochs and use 5 negative samples. All models for §5.1 are trained on the September 2016 dump of the full Wikipedia. All models for §5.2 were trained on the lemmatized and POS-tagged WaCky corpora (Baroni et al., 2009) for French, Italian, German and English (Joubarne and Inkpen, 2011; Leviant and Reichart, 2015). To ensure controlled and fair experiments, we use identical preprocessing for both models, following Levy et al. (2015).

5.1 Experiment 1: Positional Tensor

We postulate that the positional tensor should encode richer notions of syntax than standard bag-of-words vectors. Why? Positional information allow us to differentiate between the geometry of the coocurrence, e.g., `the` is found to the left of the noun it modifies and is—more often than—close to it. Our tensor factorization model explicitly encodes this information during training.

As a direct, intrinsic evaluation of the vectors, we use the QVEC evaluation framework (Tsvetkov et al., 2015; Tsvetkov et al., 2016), which measures Pearson's correlation between human-annotated research and the vectors using CCA (Hardoon et al., 2004). The QVEC metric will be higher if the vectors better correlate with the human-annotated resource. To measure the syntactic content of the vectors, we compute the correlation between our learned vector w_i for each word and its empirical distribution g_i over universal POS tags (Petrov et al., 2012) in the UD treebank (Nivre et al., 2016). g_i can be regarded as a vector on the $(|\mathcal{T}| - 1)$-dimensional simplex, where $\mathcal{T}$ is the tag set.

[4] https://bitbucket.org/omerlevy/hyperwords/

w2		ar	bg	ca	cs	da	de	el	en	es	et	eu	fa	fi	fo	fr	ga	gl	he	hi
	SG	.25	.22	.41	.20	.21	**.49**	.58	.44	.41	.09	.41	.39	.20	.32	**.41**	.22	.43	.31	.10
	HOSG	**.40**	**.46**	**.45**	**.36**	**.50**	.48	**.61**	**.48**	**.42**	**.28**	**.46**	**.43**	**.39**	**.40**	.40	**.29**	**.46**	**.44**	**.40**
	Δ	+.15	+.24	+.14	+.16	+.29	−.01	+.03	+.04	+.01	+.19	+.05	+.04	+.19	+.08	−.01	+.07	+.03	+.13	+.30
		hr	hu	id	it	kk	la	lv	nl	no	pl	pt	ro	ru	sl	sv	ta	tr	ug	vi
	SG	.51	.36	.41	.45	**.47**	.42	.21	.42	.30	.43	**.42**	.28	**.34**	.13	**.54**	**.60**	.22	.53	.57
	HOSG	**.53**	**.49**	.43	**.46**	.43	**.46**	**.38**	**.45**	**.47**	**.44**	**.42**	**.46**	.33	**.37**	.51	.58	**.41**	**.62**	**.60**
	Δ	+.02	+.13	+.02	+.01	−.04	+.04	+.17	+.03	+.17	+.01	0.0	+.18	−.01	+.24	−.03	−.02	+.21	+.09	+.03
w5		ar	bg	ca	cs	da	de	el	en	es	et	eu	fa	fi	fo	fr	ga	gl	he	hi
	SG	.24	.41	.39	.29	.44	.45	.54	.52	.45	.40	.40	.38	.37	.33	.39	.53	.40	.38	.48
	HOSG	**.29**	**.47**	**.42**	**.36**	**.49**	**.52**	**.60**	**.54**	**.48**	**.42**	**.45**	**.44**	**.43**	**.41**	**.42**	**.56**	**.45**	**.43**	**.51**
	Δ	+.05	+.06	+.03	+.07	+.04	+.07	+.06	+.02	+.03	+.02	+.05	+.06	+.06	+.08	+.08	+.06	+.06	+.05	+.03
		hr	hu	id	it	kk	la	lv	nl	no	pl	pt	ro	ru	sl	sv	ta	tr	ug	vi
	SG	.50	.46	.39	.42	**.47**	.43	.52	.43	.39	.41	.38	.38	.24	.40	.46	.59	.38	.57	.57
	HOSG	**.53**	**.49**	**.44**	**.50**	.40	**.46**	**.54**	**.50**	**.44**	**.47**	**.44**	**.43**	**.34**	**.46**	**.52**	**.58**	**.43**	**.63**	**.61**
	Δ	+.03	+.03	+.05	+.08	−.07	+.03	+.02	+.07	+.06	+.06	+.06	+.05	+.10	+.06	+.05	−.01	+.06	+.06	+.04

Table 1: The scores for QVEC-CCA for 40 languages. All embeddings were trained on the complete Wikipedia dump of September 2016. We measure correlation with universal POS tags from the UD treebanks.

We report results on 40 languages from the UD treebanks in Tab. 1, using context windows of two different sizes: 2 or 5 context words on either side. We find that for 77.5% of the languages, our positional tensor embeddings outperform the standard skip-gram approach on the QVEC metric. We highlight again that the positional tensor exploits *no* additional annotation, but better exploits the signal found in the raw text.

5.2 Experiment 2: Morphology Tensor

Since the compositional morphology tensor allows us to share parameters among related word forms, we get a single embedding for each *lemma*, i.e., the words `ran`, `run` and `running` all contribute signal to a single lemma embedding. We expect these lemma embedings to correlate well with human similarity judgments, since humans are presumably judging *conceptual* similarity and ignoring morphological inflection.

We evaluate using standard datasets on four languages: French, Italian, German and English using standard datasets. Given a list of pairs of words (always lemmata), multiple native speakers judged (with a integral value between 1 and 10) how "similar" the concepts those words represent are. The judgments were then averaged to yield a single score for the pair. Our model produces a similarity judgment for each pair using the cosine similarity of their embeddings. Tab. 2 shows how well the cosine distance between our learned vectors correlate with the human judgments, using Spearman's correlation coefficient. Our model does achieve higher correlation than standard skip-gram.

6 Related Work

Tensor factorization has already found uses in a few corners of NLP research. Van de Cruys et al. (2013) applied tensor factorization to model the compositionality of subject-verb-object triples. Similarly, Hashimoto and Tsuruoka (2015) use an implicit tensor factorization method to learn embeddings for transitive verb phrases. Tensor factorization also appears in semantic-based NLP tasks. Lei et al. (2015) explicitly factorize a tensor based on feature vectors for predicting semantic roles. Chang et al. (2014) use tensor factorization to create knowledge base embeddings optimized for relation extraction. See Bouchard et al. (2015) for a large bibliography.

Other researchers have likewise attempted to escape the bag-of-words assumption in word embeddings, e.g., Yatbaz et al. (2012) incorporates morphological and orthographic features into continuous vectors, Cotterell and Schütze (2015) consider a multi-task set-up to force morphological information into embeddings, Cotterell and Schütze (2017) jointly morphologically segment and embed words, Levy and Goldberg (2014a) derive contexts based on dependency relations and, finally, Schwartz et al. (2016) derived embeddings based on Hearst patterns (Hearst, 1992). Ling et al. (2015) propose structured word2vec models that specifically include word order information. As demonstrated in the experiments, our tensor factorization method enables us to include other syntactic properties besides for word order, e.g. morphology. Poliak et al. (2017) also create positional word embeddings. Our research direction is orthogonal to these efforts in that we provide a general purpose procedure for all sorts of higher-order coocurrence.

7 Conclusion

We have presented an interpretation of the skip-gram model as exponential family principal component analysis—a form of matrix factorization—and, thus, related it to an older strain of work. Build-

	fr	it		de				en						
	353	353	SimL	RG-65	353	SimL	Z222	RG-65	353	MEN	MTURK	SimL	SimV	RW
SG	48.31	43.63	21.33	44.90	28.39	50.39	29.75	70.60	**64.50**	64.33	58.77	41.62	30.48	40.78
HOSG	**58.21**	**45.00**	**28.54**	**68.08**	**40.09**	**53.97**	**31.11**	**71.71**	63.72	**66.66**	**62.64**	**49.70**	**29.96**	**42.40**
Δ	+9.90	+1.37	+7.21	+23.18	+11.7	+3.58	+1.36	+1.11	-0.78	+2.33	+3.87	+8.08	+0.52	+1.62

Table 2: Word similarity results comparing the compositional morphology tensor with the standard skip-gram model. Number indicate Spearman's ρ between human judgements and cosine distance between vectors.

ing on this connection, we generalized the model to the tensor case, allowing us to incorporate rich linguistic structure in our model. We illustrated two higher-order skip-gram methods that easily incorporate more structure, e.g. word order and morphology, into the objective function without sacrificing scalability. These methods achieved better word embeddings as evaluated by standard metrics on 40 languages.

Acknowledgements

We thank colleagues, particularly Yanif Ahmad, and anonymous reviewers for helpful discussion and feedback. The first author was supported by a DAAD Long-Term Research Grant and an NDSEG fellowship. This work was supported in part by the JHU Human Language Technology Center of Excellence (HLTCOE), and DARPA DEFT. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes. The views and conclusions contained in this publication are those of the authors and should not be interpreted as representing official policies or endorsements of DARPA or the U.S. Government.

References

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.

Guillaume Bouchard, Jason Naradowsky, Sebastian Riedel, Tim Rocktäschel, and Andreas Vlachos. 2015. Matrix and tensor factorization methods for natural language processing. In *Tutorials*, pages 16–18, Beijing, China, July. Association for Computational Linguistics.

Kai-Wei Chang, Wen-tau Yih, Bishan Yang, and Christopher Meek. 2014. Typed tensor decomposition of knowledge bases for relation extraction. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1568–1579, Doha, Qatar, October. Association for Computational Linguistics.

Michael Collins, Sanjoy Dasgupta, and Robert E. Schapire. 2001. A generalization of principal components analysis to the exponential family. In *Advances in Neural Information Processing Systems 14*, pages 617–624.

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.

Ryan Cotterell and Hinrich Schütze. 2017. Joint semantic synthesis and morphological analysis of the derived word. *CoRR*, abs/1701.00946.

Ryan Cotterell, Hinrich Schütze, and Jason Eisner. 2016. Morphological smoothing and extrapolation of word embeddings. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1651–1660, Berlin, Germany, August. Association for Computational Linguistics.

Carl Eckart and Gale Young. 1936. The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218.

Yoav Goldberg and Omer Levy. 2014. word2vec explained: deriving Mikolov et al.'s negative-sampling word-embedding method. *arXiv preprint arXiv:1402.3722*.

Gene H Golub and Charles F Van Loan. 2012. *Matrix Computations*, volume 3. JHU Press.

David R Hardoon, Sandor Szedmak, and John Shawe-Taylor. 2004. Canonical correlation analysis: An overview with application to learning methods. *Neural Computation*, 16(12):2639–2664.

Kazuma Hashimoto and Yoshimasa Tsuruoka. 2015. Learning embeddings for transitive verb disambiguation by implicit tensor factorization. In *Proceedings of the 3rd Workshop on Continuous Vector Space Models and their Compositionality*.

Marti A Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th conference on Computational linguistics-Volume 2*, pages 539–545. Association for Computational Linguistics.

Frank L Hitchcock. 1927. The expression of a tensor or a polyadic as a sum of products. *Journal of Mathematics and Physics*, 6(1):164–189.

Thomas Hofmann. 1999. Probabilistic latent semantic analysis. In *Uncertainty in Artificial Intelligence*, pages 289–296.

Colette Joubarne and Diana Inkpen. 2011. Comparison of semantic similarity for different languages using the Google n-gram corpus and second-order co-occurrence measures. In *Canadian Conference on Artificial Intelligence*, pages 216–221. Springer.

Tamara Kolda and Brett Bader. 2009. Tensor decompositions and applications. *Society for Industrial and Applied Mathematics*, 51(3):455–500.

Tao Lei, Yuan Zhang, Lluís Màrquez, Alessandro Moschitti, and Regina Barzilay. 2015. High-order low-rank tensors for semantic role labeling. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1150–1160, Denver, Colorado, May–June. Association for Computational Linguistics.

Ira Leviant and Roi Reichart. 2015. Separated by an un-common language: Towards judgment language informed vector space modeling. *arXiv*.

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)*, pages 302–308, Baltimore, Maryland, June. Association for Computational Linguistics.

Omer Levy and Yoav Goldberg. 2014b. Neural word embedding as implicit matrix factorization. In *Advances in Neural Information Processing Systems 27*, pages 2177–2185.

Omer Levy, Yoav Goldberg, and Ido Dagan. 2015. Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics*, 3:211–225.

Wang Ling, Chris Dyer, Alan W Black, and Isabel Trancoso. 2015. Two/too simple adaptations of word2vec for syntax problems. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1299–1304, Denver, Colorado, May–June. Association for Computational Linguistics.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.

Shakir Mohamed, Katherine A. Heller, and Zoubin Ghahramani. 2008. Bayesian exponential family PCA. In *Advances in Neural Information Processing Systems 21*, pages 1089–1096.

Shakir Mohamed. 2011. *Generalised Bayesian Matrix Factorisation Models*. Ph.D. thesis, University of Cambridge.

Frederic Morin and Yoshua Bengio. 2005. Hierarchical probabilistic neural network language model. In *Artificial Intelligence and Statistics Conference*.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajic, Christopher D Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, et al. 2016. Universal dependencies v1: A multilingual treebank collection. In *LREC*.

Karl Pearson. 1901. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, 6(2):559–572.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar, October. Association for Computational Linguistics.

Slav Petrov, Dipanjan Das, and Ryan McDonald. 2012. A universal part-of-speech tagset. In Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Mehmet Uğur Doğan, Bente Maegaard, Joseph Mariani, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*, pages 2089–2096, Istanbul, Turkey, May. European Language Resources Association (ELRA). ACL Anthology Identifier: L12-1115.

Adam Poliak, Pushpendre Rastogi, Michael Patrick Martin, and Benjamin Van Durme. 2017. Efficient, compositional, order-sensitive n-gram embeddings. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*.

Herbert Robbins and Sutton Monro. 1951. A stochastic approximation method. *The Annals of Mathematical Statistics*, pages 400–407.

Sam T. Roweis. 1997. EM algorithms for PCA and SPCA. In *Advances in Neural Information Processing Systems 10*, pages 626–632.

Roy Schwartz, Roi Reichart, and Ari Rappoport. 2016. Symmetric patterns and coordinations: Fast and enhanced representations of verbs and adjectives. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 499–505, San Diego, California, June. Association for Computational Linguistics.

Ajit Paul Singh and Geoffrey J. Gordon. 2008. A unified view of matrix factorization models. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 358–373.

Nathan Srebro, Jason D. M. Rennie, and Tommi S. Jaakkola. 2004. Maximum-margin matrix factorization. In *Advances in Neural Information Processing Systems 17*, pages 1329–1336.

Michael Tipping and Christopher Bishop. 1999. Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 61(3):611–622.

Yulia Tsvetkov, Manaal Faruqui, Wang Ling, Guillaume Lample, and Chris Dyer. 2015. Evaluation of word vector representations by subspace alignment. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2049–2054, Lisbon, Portugal, September. Association for Computational Linguistics.

Yulia Tsvetkov, Manaal Faruqui, and Chris Dyer. 2016. Correlation-based intrinsic evaluation of word vector representations. In *RepEval*.

Ledyard R. Tucker. 1966. Some mathematical notes on three-mode factor analysis. *Psychometrika*, 31:279–311.

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*, pages 1142–1151, Atlanta, Georgia, June. Association for Computational Linguistics.

Mehmet Ali Yatbaz, Enis Sert, and Deniz Yuret. 2012. Learning syntactic categories using paradigmatic representations of word context. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 940–951, Jeju Island, Korea, July. Association for Computational Linguistics.