Learning Low-Rank Tensors for Transitive Verbs

Daniel Fried Tamara Polajnar Stephen Clark
Computer Laboratory
University of Cambridge
{df345,tp366,sc609}@cam.ac.uk

1 Overview

Coecke et al. (2010) describe a type-driven framework for composing distributional word representations in vector space, producing distributional representations for phrases and sentences. One barrier to the practical implementation of this framework is the large number of parameters, in the form of matrices and higher-order tensors, required to specify composition functions on vectors. Focusing on the case of transitive verbs, we use distributional representations for nouns and subject-verb-object (SVO) triples from a large text corpus to train tensor representations for verbs.

We use low-rank tensor decompositions to reduce the number of parameters for each verb, and evaluate the ability of these low-rank tensors to match human judgements about similarity of SVO triples on two tasks. We present preliminary results showing that, while the low-rank tensors require about two orders of magnitude fewer parameters per verb, they can achieve performance comparable to unconstrained-rank tensors on a sentence similarity task, and occasionally surpass the performance of unconstrained-rank tensors on a verb disambiguation task.

2 Model

Tensor Models for Verbs: Following Coecke et al. (2010), we model transitive verbs using third-order tensors (arrays with three indices). Each tensor specifies a function that takes two noun vectors as input and outputs a sentence vector. Given a tensor $\mathcal{V} \in \mathbb{R}^{I \times J \times K}$ representing a transitive verb, and vectors $\mathbf{s} \in \mathbb{R}^J$, $\mathbf{o} \in \mathbb{R}^K$ representing subject and object nouns, respectively, we produce a compositional vector representation for the SVO triple by contracting the tensor with the two vectors (Maillard et al., 2014), producing a vector $(\mathcal{V}\mathbf{o})\mathbf{s}$ with ith component given by:

$$((\mathcal{V}\mathbf{o})\mathbf{s})_i = \sum_j \sum_k \mathcal{V}_{ijk}\mathbf{o}_k\mathbf{s}_j \tag{1}$$

Low-Rank Tensors and Decompositions: Following Lei et al. (2014), we represent tensors using low-rank *CP decompositions* to reduce the numbers of parameters that must be learned during training. As a higher-order analogue to matrix decomposition, CP decomposition factors a tensor into a sum of R outer products of vectors. Given a third-order tensor $\mathcal{V} \in \mathbb{R}^{I \times J \times K}$, the CP decomposition of $\mathcal{V}$ is:

$$\mathcal{V} = \sum_{r=1}^{R} \mathbf{u}^{(r)} \otimes \mathbf{v}^{(r)} \otimes \mathbf{w}^{(r)} \tag{2}$$

where we have R each of the vectors $\mathbf{u}^{(r)} \in \mathbb{R}^I$, $\mathbf{v}^{(r)} \in \mathbb{R}^J$, and $\mathbf{w}^{(r)} \in \mathbb{R}^K$. The smallest R that allows the tensor to be expressed this way is the *rank* of the tensor (Kolda and Bader, 2009). By choosing a value for R and representing the tensor in this form, we force the tensor to have rank no more than R. If R is sufficiently small compared to I, J, and K, this allows us to learn a low-rank tensor representable with fewer parameters than the full third-order tensor.

Representing tensors in this form allows us to formulate tensor contraction as matrix multiplication. For example, for the transitive verb case, we can represent the vectors $\mathbf{u}, \mathbf{v}, \mathbf{w}$ as the rows of matrices

	Unconstrained	Low Rank							
		Joint Optimization				Alternating Optimization			
	Rank	R=5	R=10	R=20	R=30	R=5	R=10	R=20	R=30
GS2011 correlation	0.25	0.21	0.23	**0.30**	0.26	0.23	0.25	0.28	0.20
KS2013 correlation	**0.42**	0.26	0.32	0.30	0.31	0.30	0.37	0.39	0.36
parameters per verb	400K	1.2K	2.4K	4.8K	7.2K	1.2K	2.4K	4.8K	7.2K

Table 1: Spearman correlation on verb disambiguation (GS2011) and transitive sentence similarity (KS2013) tasks, compared across unconstrained- and low-rank tensors and optimization methods. All experiments use 100-dimensional noun vectors and 40-dimensional sentence space vectors.

$\mathbf{U} \in \mathbb{R}^{R \times I}, \mathbf{V} \in \mathbb{R}^{R \times J}, \mathbf{W} \in \mathbb{R}^{R \times K}$. Then the SVO vector representation is given by

$$(\mathcal{V}\mathbf{o})\mathbf{s} = \mathbf{U}^{\top}(\mathbf{V}\mathbf{s} \circ \mathbf{W}\mathbf{o}) \tag{3}$$

where $\circ$ is the elementwise vector product.

3 Experiments

Training Data: We extract SVO triples from an October 2013 dump of Wikipedia, parsed with the C&C parser (Curran et al., 2007), using the most frequent lemmatised words within sentence boundaries as context. We use the standard tTest and singular value decomposition methods to re-weight and reduce the dimensionality of the co-occurrence counts, producing 100-dimensional distributional vectors for nouns and 40-dimensional distributional vectors for SVO triples.

Training Methods: We learn unconstrained-rank tensors using the gradient descent regression method of Polajnar et al. (2014). For low-rank tensors, we investigate setting four different maximal ranks, $R = 5, 10, 20$, and 30, and compare two different optimization methods. The first optimization method, *joint optimization*, performs gradient descent on all three parameter sets $(\mathbf{U}, \mathbf{V}, \mathbf{W})$ of the tensor simultaneously. The second, *alternating optimization*, is similar to the alternating method described by Kolda and Bader (2009) and Lei et al. (2014), and fixes two of the parameter sets in turn while performing gradient descent on the third. For all methods, we use mini-batch ADADELTA optimization (Zeiler, 2012), using early stopping on a 10% held-out validation set of SVO triples. In the alternating optimization experiments, we perform 10 ADADELTA iterations on each parameter. For comparison, we limit the maximum total iterations to the same number as in the unconstrained-tensor optimization.

Evaluation Tasks: We compare the performance of the low-rank verb tensors against unconstrained-rank tensors on two tasks: verb disambiguation (Grefenstette and Sadrzadeh, 2011) and transitive sentence similarity (Kartsaklis and Sadrzadeh, 2013). Both tasks require the system to score SVO triples by similarity, measuring the correlation of the systems' similarity scores with human-produced judgements. To do this, we produce a vector representation for a given SVO triple by contracting the verb's tensor with the distributional vectors for the subject and object nouns. Then, we calculate a similarity score for two SVO triples by taking the cosine similarity of their resulting vectors, and compare these pairwise similarities with the human judgements using Spearman's rank correlation.

4 Results

Table 1 displays correlations between the systems' scores and human SVO similarity judgements on the two tasks. Low-rank tensor performance varies greatly across values of R, but is maximized at either $R = 10$ or $R = 20$ for both the joint and alternating optimization methods. In six out of eight combinations of dataset and maximal rank, the alternating optimization training method achieves higher results than the joint optimization method. For the combination $R = 20$ and alternating optimization method, which seems to provide the most stable performance, the low-rank tensor achieves results comparable to the unconstrained-rank tensor on both datasets: .03 higher on GS2011, and .03 lower on KS2013, despite using only 4,800 parameters per verb compared to the 400,000 parameters per verb of the unconstrained tensor model. This is ongoing work, and we plan to continue to compare the low-rank and unconstrained-rank tensor models on other evaluation tasks.

References

Coecke, B., M. Sadrzadeh, and S. Clark (2010). Mathematical foundations for a compositional distributional model of meaning. *arXiv preprint arXiv:1003.4394*.

Curran, J. R., S. Clark, and J. Bos (2007). Linguistically motivated large-scale NLP with C&C and Boxer. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, pp. 33–36. Association for Computational Linguistics.

Grefenstette, E. and M. Sadrzadeh (2011). Experimenting with transitive verbs in a DisCoCat. *CoRR abs/1107.3119*.

Kartsaklis, D. and M. Sadrzadeh (2013). Prior disambiguation of word tensors for constructing sentence vectors. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pp. 1590–1601. Association for Computational Linguistics.

Kolda, T. G. and B. W. Bader (2009). Tensor decompositions and applications. *SIAM review 51*(3), 455–500.

Lei, T., Y. Xin, Y. Zhang, R. Barzilay, and T. Jaakkola (2014). Low-rank tensors for scoring dependency structures. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Baltimore, Maryland, pp. 1381–1391. Association for Computational Linguistics.

Maillard, J., S. Clark, and E. Grefenstette (2014). A type-driven tensor-based semantics for CCG. *EACL 2014 Type Theory and Natural Language Semantics Workshop*.

Polajnar, T., L. Rimell, and S. Clark (2014). Using sentence plausibility to learn the semantics of transitive verbs. *CoRR abs/1411.7942*.

Zeiler, M. D. (2012). ADADELTA: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*.