A Compressed Sensing View of Unsupervised Text Embeddings, Bag-of-n-Grams, and LSTMs

Sanjeev Arora, Mikhail Khodak, Nikunj Saunshi
Princeton University
{arora,mkhodak,nsaunshi}@cs.princeton.edu

Kiran Vodrahalli
Columbia University
kiran.vodrahalli@columbia.edu

***Abstract*—Low-dimensional embeddings, computed by LSTMs or other techniques, are a popular approach for capturing the "meaning" of text and a useful form of unsupervised learning. However, their power is not theoretically understood. We derive formal understanding by looking at the subcase of linear embedding schemes. Using compressed sensing theory we show that representations combining the constituent word vectors can be information-preserving linear measurements of Bag-of-n-Grams (BonG) representations of text. This leads to a new theoretical result about LSTMs: embeddings derived from a low-memory LSTM are provably at least as powerful on classification tasks as a linear classifier over BonG vectors, a result that extensive empirical work has thus far been unable to show. Our experiments support these findings and establish strong baselines on standard benchmarks. We also show a surprising new property of pretrained word embeddings: they form a sensing matrix for text that is more efficient than random matrices, which may explain why they lead to better representations in practice.**

The full version of this work appears in the *Proceedings of the 6th International Conference on Learning Representations* (ICLR 2018).[1]

I. Introduction

Much attention has been paid to using LSTMs [9] and similar models for text embedding [2], [5], [11]. LSTMs process text in limited memory and output a vector that can be used as a featurization for downstream tasks. However, their powers and limitations have not been formally established. For example, can they compete with traditional linear classifiers over trivial but surprisingly powerful Bag-of-n-Grams (BonG) representations [19], which continue to give better performance on many downstream tasks? Meanwhile evidence suggests that simpler *linear* schemes, consisting of adding up standard pretrained word embeddings [13], [16], give compact representations that provide most of the benefits of LSTM embeddings [1].

We tie these threads together via an information-theoretic account of linear text embeddings with schemes that preserve n-gram information as low-dimensional embeddings with *provable* guarantees for any text classification task. Furthermore, we show that the original information can be extracted from the low-dimensional embedding using compressed sensing [4]. The following are our main results:

1) Using random vectors in our scheme we show that low-memory LSTMs are *provably* at least as good for linear classification as the full BonG. This novel theoretical result is obtained by generalizing a theorem by [3]. Extensive empirical study of this issue has been inconclusive, and we do not know of any previous provable quantification of the power of embeddings.
2) We study experimentally how our scheme improves with pretrained embeddings (e.g. GloVe) instead of random vectors. We find that they allow better preservation of Bag-of-Words (BoW) information, i.e. they are better than random matrices for "sensing" BoW signals. We give some theoretical justification for this surprising finding via a new sparse recovery property characterizing nonnegative signal recovery.
3) Finally, we support our theoretical work with empirical results showing that our embeddings are consistently competitive with recent schemes and perform much better than all previous linear methods on standard tasks. As our representations are fast and simple to implement they are strong baselines for future work.

[1] https://openreview.net/forum?id=B1e5ef-C-

II. Related Work

Distributed representations have long been studied in connectionist approaches [8], [17]. Our method is closely related to the sparse distributed memory of [10], while in the unigram case it reduces to the familiar sum of word embeddings, known to be surprisingly powerful [1]. Compression of BonGs has been studied using classical lossless algorithms by [15] and by linear schemes by [14], though this motivation is not made in the latter. The novelty in our paper is the connection to compressed sensing, which is concerned with sparse recovery of $x \in \mathbb{R}^N$ from low-dimensional measurements Ax by studying conditions on matrix $A \in \mathbb{R}^{d \times N}$ when this is possible. Our approach is closely related to related results in learning by [3].

III. Document Embeddings

Our analysis relates feature counting vectors and low-dimensional embeddings via linear compression. Given a vocabulary of V words we define a document's *Bag-of-Words* (BoW) x^{BoW} to be the V-dimensional vector counting each word's occurrence. An extension is the *Bag-of-n-Grams* (BonG), which counts all k-grams for $k \leq n$. For ease of analysis we merge all n-grams containing the same words in a different order, calling the resulting vector a *Bag-of-n-Cooccurrences* (BonC); this does not affect performance significantly.

Now let word w have a vector $v_w \in \mathbb{R}^d$ for $d \ll V$. For document $w_1, \ldots, w_T$ we define the *unigram embedding* as $z^{\text{u}} = \sum_{t=1}^{T} v_{w_t}$. This has a straightforward relation with BoW: if $A \in \mathbb{R}^{d \times V}$ is a matrix with word vector columns then $z^{\text{u}} = Ax^{\text{BoW}}$. To include n-grams while remaining low-dimensional, we use elementwise multiplication, so that for $g = \{w_1, \ldots, w_n\}$ we have the *distributed cooccurrence* (DisC) embedding $\tilde{v}_g = d^{\frac{n-1}{2}} \bigodot_{t=1}^{n} v_{w_t}$. Then the *DisC document embedding* is defined as the nd-dimensional concatenation, over $k \leq n$, of the sum of all k-gram DisC vectors. As with the unigram subcase one can construct a matrix $A^{(n)}$ such that $z^{(n)} = A^{(n)} x^{\text{BonC}}$.

Finally, we show how these linear schemes are related to LSTMs. Starting with $h_0 = \mathbf{0}_m$ an *m-memory LSTM with word vectors* $v_w \in \mathbb{R}^d$ takes in words $w_1, \ldots, w_T$ one-by-one and computes

$$h_{t+1} = f(\mathcal{T}_f(v_{w_{t-1}}, h_t)) \circ h_t + i(\mathcal{T}_i(v_{w_{t-1}}, h_t)) \circ g(\mathcal{T}_g(v_{w_{t-1}}, h_t))$$

for hidden state $h_t \in \mathbb{R}^m$, "activation" functions f, i, g, and affine transformations $\mathcal{T}_*(x, y) = W_* x + U_* y + b_*$ with weight matrices $W_* \in \mathbb{R}^{m \times d}, U_* \in \mathbb{R}^{m \times m}$ and bias vectors $b_* \in \mathbb{R}^m$. The *LSTM representation* is then the state at the last time step, i.e. $z^{\text{LSTM}} = h_T$. Importantly, we can show an initialization of gates and input functions that constructs the previously-defined DisC embeddings:

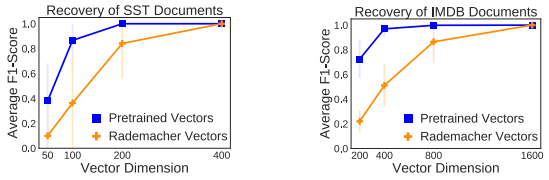

Fig. 1. Average F_1-score of recovered BoW vectors from SST and IMDB vs. dimension. Pretrained word embeddings need half the dimensionality of random vectors to achieve near-perfect recovery.

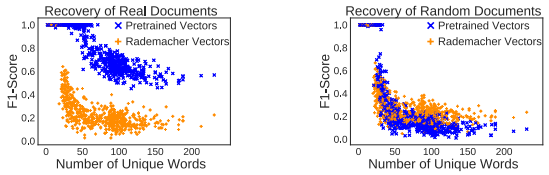

Fig. 2. F_1-score of 1000 recovered BoWs compared to number of words. For $d = 200$, pretrained embeddings are better than random vectors as sensing vectors for natural language BoW but are worse for random sparse signals.

Proposition 3.1: Given word vectors $v_w \in \mathbb{R}^d$, one can initialize an $\mathcal{O}(nd)$-memory LSTM that takes in words $w_1, \ldots, w_T$ and constructs a DisC embedding, i.e. for all documents $z^{\text{LSTM}} = z^{(n)}$.

IV. LSTMs as Compressed Learners

Our main contribution is an analysis of our embedding schemes showing that they are at least as good as BonCs for classification. This requires two simplifying assumptions: the BonCs are scaled by $\frac{1}{T\sqrt{n}}$ and no n-cooccurrence contains a word more than once.

Theorem 4.1: Let $S = \{(x_i, y_i)\}_{i=1}^m$ be drawn i.i.d. from a distribution $\mathcal{D}$ over BonCs of documents of length at most T satisfying the above assumptions and let w_0 be the linear classifier minimizing the logistic loss $\ell_{\mathcal{D}}$. Then for $d = \tilde{\Omega}\left(\frac{T}{\varepsilon^2} \log \frac{nV}{\gamma}\right)$ and appropriate choice of regularization coefficient one can initialize an $\mathcal{O}(nd)$-memory LSTM over i.i.d. word embeddings $v_w \sim \mathcal{U}^d\{\pm 1/\sqrt{d}\}$ such that w.p. $(1-\gamma)(1-2\delta)$ the classifier $\hat{w}$ minimizing the ℓ_2-regularized logistic loss over its representations satisfies

$$\ell_{\mathcal{D}}(\hat{w}) \leq \ell_{\mathcal{D}}(w_0) + \mathcal{O}\left(\|w_0\|_2 \sqrt{\varepsilon + \frac{1}{m} \log \frac{1}{\delta}}\right)$$

This bound shows that LSTMs match BonC performance as $\varepsilon \to 0$ i.e. by increasing the embedding dimension d. To prove this theorem, we generalize a result in [3] and show that learning is possible under linear compression if the matrix satisfies a strong recovery property (RIP). We then show that the matrix $A^{(n)}$ for random embeddings is a bounded orthonormal system and hence satisfies this property [7].

V. Sparse Recovery with Pretrained Embeddings

Theorem 4.1 is proved using random embeddings, but in practice LSTMs use vectors such as GloVe that do not satisfy the required recovery property. Here we present the surprising empirical finding that pretrained embeddings are *more efficient* at encoding and recovering BoWs by ℓ_1-minimization. We take documents from the SST [18] and IMDB [12] datasets, embed them as $z^{\text{u}} = Ax^{\text{BoW}}$ for $d = 50, 100, 200, \ldots, 1600$ (where $A \in \mathbb{R}^{d \times V}$ is the embedding matrix), and solve Basis Pursuit (BP): $\min \|w\|_1$ s.t. $Aw = z^{\text{u}}$.

Figures 1 and 2 show that pretrained embeddings need a lower dimension than random vectors to recover BoWs. This is surprising as their objective goes against usual conditions such as incoherence; indeed as seen in Figure 2 recovery is poor for random signals.

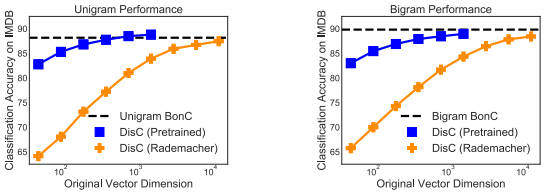

Fig. 3. DisC embedding performance vs. original dimension.

Representation	n	d	SST (± 1)	SST	IMDB
BonC	1-3	20K-200K+	80.9	42.3	90.0
DisC	1-3	1600-4800	**85.5**	**46.7**	89.6
SIF [1]	1	1600	84.4	45.8	89.2
Sent2Vec [14]	1-2	700	80.2	31.0	85.5
CFL [15]	5	100K+			**90.4**
skip-thoughts [11]		4800	85.1	45.8	

The latter indicates that the fact that documents are meaningful sets of words is important for sparse recovery using cooccurrence-based embeddings. While properties for recovering signals with support $S \subset [V]$ are hard to check, we use geometric results for nonnegative BP [6] to formulate a verifiable condition, the *supporting hyperplane property*, that is indeed more likely to be satisfied by pretrained embeddings. As similarity properties that may explain these results also relate to downstream tasks, we conjecture a relationship between embeddings, recovery, and classification that may be understood under a generative model. Though these experiments do not directly apply to the Section IV bounds, they show that the compressed sensing framework is relevant even for pretrained word embeddings.

VI. Empirical Findings

Our theoretical results show that our n-gram embeddings can approach BonG performance. We find that DisC is comparable to other representations on several standard tasks, being the top performer on the SST tasks, and verify that DisC performance on the IMDB task approaches that of BonCs as dimensionality increases (Figure 3), as predicted by Theorem 4.1. Using pretrained vectors, DisC performance reaches BonC earlier, surpassing it for unigrams.

References

[1] Arora et al. *A Simple but Tough-to-Beat Baseline for Sentence Embeddings*. ICLR 2017.
[2] Bengio et al. *A Neural Probabilistic Language Model*. JMLR **3**, 2003.
[3] Calderbank et al. *Compressive Learning*. Technical Report, 2009.
[4] Candès & Tao. *Decoding by LP*. IEEE Trans. Info. Theory **51**, 2005.
[5] Collobert & Weston. *A Unified Architecture for NLP*. ICML 2008.
[6] Donoho & Tanner. *Sparse Nonnegative Solution of Underdetermined Linear Equations by LP*. PNAS **102**, 2005.
[7] Foucart & Rauhut *A Mathematical Introduction to Compressive Sensing*. Springer 2013.
[8] Hinton. *Mapping Part-Whole Hierarchies into Connectionist Networks*. Artificial Intelligence **46**, 1990.
[9] Hochreiter & Schmidhuber. *LSTM*. Neural Computation **9**, 1997.
[10] Kanerva. *Hyperdimensional Computing*. Cognitive Computation **1**, 2009.
[11] Kiros et al. *Skip-Thought Vectors*. NIPS 2015.
[12] Maas et al. *Learning Word Vectors for Sentiment Analysis*. ACL 2011.
[13] Mikolov et al. *Distributed Representations of Words*. NIPS 2013.
[14] Pagliardini et al. *Unsupervised Learning of Sentence Embeddings using Compositional N-Gram Features*. NAACL 2018.
[15] Paskov et al. *Compressive Feature Learning*. NIPS 2013.
[16] Pennington et al. *GloVe*. EMNLP 2014.
[17] Plate. *Holographic Reduced Representations*. IEEE Trans. NN, 1995.
[18] Socher et al. *Recursive Models for Semantic Composition*. EMNLP 2013.
[19] Wang & Manning. *Bigrams and Baselines*. ACL 2012.