Documents Representation via Generalized Coupled Tensor Chain with the Rotation Group Constraint

Igor Vorona, Anh-Huy Phan, Alexander Panchenko and Andrzej Cichocki
Skolkovo Institute of Science and Technology, Moscow, Russia
{igor.vorona,a.phan, a.panchenko, a.cichocki}@skoltech.ru

Abstract

Continuous representations of linguistic structures are an important part of modern natural language processing systems. Despite the diversity, most of the existing log-multilinear embedding models are organized under vector operations. However, these operations can not precisely represent the compositionality of natural language due to a lack of order-preserving properties. In this work, we focus on one of the promising alternatives based on the embedding of documents and words in the rotation group through the generalization of the coupled tensor chain decomposition to the exponential family of the probability distributions. In this model, documents and words are represented as matrices, and n-grams representations are combined from word representations by matrix multiplication. The proposed model is optimized via noise-contrastive estimation. We show empirically that capturing word order and higher-order word interactions allows our model to achieve the best results in several document classification benchmarks.

1 Introduction

The current progress in natural language processing systems is largely based on the success of the representation learning of linguistic structures such as word, sentence and document embeddings. The most promising and successful methods are based on learning representations via two types of models: shallow log-multilinear models and deep neural networks. Despite the efficiency and interpretability of log-multilinear models, they can not use higher-order linguistic features like dependency between subsequences of words. To avoid these disadvantages, we usually use nonlinear predictors like recurrent, recursive, convolution neural networks, and more recently Transformers. Nevertheless, these methods can achieve high performance at the cost of loss of some interpretability and the cost of increased computation time.

However, there exist other ways to utilize higher-order interactions and still preserve the efficiency of linear models. In this research, we focus on more data-oriented, i.e., more linguistic grounded, and better interpretable methods which still can achieve high results in the practical tasks. Particularly, we investigate the matrix-space model of language, in which semantic space consists of square matrices of real values. The key idea behind this method goes from realization of the Frege's principle of compositionality through order-preserving property of matrix operations. As shown by Rudolph and Giesbrecht (2010), this type of models can internally combine various properties from statistical and symbolic models of natural language and therefore it is more flexible than vector space models.

In spite of that fact, such models are usually hard to optimize on the real data. To this end, Yessenalina and Cardie (2011) took attention to the needs of nontrivial initialization and proposed to learn the weights by the bag-of-words model. Asaadi and Rudolph (2017) used complex multi-stage initialization based on unigrams and bigrams scoring. Both approaches try to solve the sentiment analysis task. Recently Mai et al. (2019) considered the problem of self-supervised continuous representation of words via matrix-space models. They optimized a modified word2vec objective function (Mikolov et al., 2013) and proposed a novel initialization by adding small isotropic Gaussian noise to the identity matrix.

In this paper, we use a similarity function between matrices similar to Mai et al. (2019), but instead of neural network type of learning, we implement the model as the coupled tensor chain and impose the rotation group constraint. We focus on the document representation problem. Given a document collection, we try to find unsupervised doc-

Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 1674–1684
August 1–6, 2021. ©2021 Association for Computational Linguistics

ument and word representation suitable for downstream linear classification tasks. To this end, we represent words, n-grams and documents as matrices and train self-supervised model. Our intuition is based on the fact that modeling interaction between words and documents is insufficient for modeling relations between complex phrases and documents.

Contributions. The main contributions of this work can be summarized as follows:

- To the best of our knowledge, this is the first representation learning method based on the Riemannian geometry of matrix groups.
- We show that our approach to model the compositionality and word order allows us to increase the quality of document embedding on downstream tasks. Moreover, it is also more computationally efficient in comparison with neural network models.
- Our model achieves state-of-the-art performance on the task of representation learning for multiclass classification both on short and long document datasets.

Our implementation of the proposed model is available online[1].

2 Related work

Euclidean embedding models (Mikolov et al., 2013; Pennington et al., 2014) based on implicit word-context co-occurance matrix factorization (Levy and Goldberg, 2014) are an important framework for current NLP tasks. Proposed models achieve relatively high performance in various NLP tasks like text classification (Kim, 2014), named entity recognition (Lample et al., 2016), machine translation (Cho et al., 2014).

Riemannian embedding models have shown promising results by expanding embedding methods beyond Euclidean geometry. There are several models with negative sectional curvature like Poincare (Dhingra et al., 2018; Nickel and Kiela, 2017) and Lorentz models (Nickel and Kiela, 2018). Furthermore, Meng et al. (2019) proposed a text embedding model based on spherical geometry.

Tensor decomposition models have been applied to many tasks in the NLP. Particularly, Van de Cruys et al. (2013) proposed the Tucker model for decomposing subject-verb-object co-occurance tensor for computation of compositionality. The most similar to our task is the word embedding problem. In this direction, Sharan and Valiant (2017) explored the Canonical Polyadic Decomposition (CPD) of word triplet tensor. Bailey and Aeron (2017) used symmetric CPD of pointwise mutual information tensor. Frandsen and Ge (2019) extended the RAND-WALK model (Arora et al., 2016) to word triplets. The main drawback of existing approaches is that they can not preserve word order information of long n-grams properly. For example, in the case of the CPD, we need to use separate parameters for each word based on its position in the text. This restriction does not allow us to efficiently use the linguistic meaning of tensor modes. The symmetric CPD completely loses word order information and the Tucker model suffers from exponentially increasing parameter size in the case of long length n-grams. Our approach eliminates these disadvantages.

3 Problem and model description

In this section, we describe our model for the document representation task. We begin with a short introduction of the multilinear algebra, then present the proposed document modeling framework in the view of the coupled tensor decomposition and provide the detailed description of our model and indicate benefits/drawbacks which are related to rotational group constraints.

3.1 Basic multilinear algebra

A tensor is a higher-order generalization of vectors and matrices to multiple indices. The order of a tensor is the number of dimensions, also known as modes or ways. An N-th order tensor is represented as $\underline{\mathbf{X}} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$, and its element is denoted as $x_{i_1 \ldots i_N}$. We can always represent a tensor $\underline{\mathbf{X}}$ as sum of rank-1 tensors, where each of them is defined as outer product of N-vectors, i.e., $\mathbf{a}^{(1)} \circ \cdots \circ \mathbf{a}^{(N)}$ and $\mathbf{a}^{(n)} \in \mathbb{R}^{I_n}$ for $n = 1, \ldots, N$. The minimal number of rank-1 tensors in this sum defines tensor rank.

In this research we focus on a particular type of tensor decomposition called tensor chain (Perez-Garcia et al., 2007; Khoromskij, 2011; Zhao et al., 2019). It represents tensor via the following sum of rank-1 tensors

$$\underline{\mathbf{X}} = \sum_{r_1, \ldots, r_N = 1}^{R_1, \ldots, R_N} \mathbf{a}^{(1)}_{r_1 r_2} \circ \cdots \circ \mathbf{a}^{(N)}_{r_N r_1}, \qquad (1)$$

[1] https://github.com/harrycrow/RDM

Notation	Description
a	Scalar
$\mathbf{a}$	Vector or tuple
$\mathbf{A}$	Matrix
$\underline{\mathbf{A}}$	Higher-order tensor
$\circ$	Outer (tensor) product
$\otimes$	Kronecker product
$\mathrm{tr}(\cdot)$	Trace of matrix
$\mathrm{vec}(\cdot)$	Vectorization of tensor
$[\cdot]_+$	Euclidean projection onto nonnegative orthant, i.e., $\max\{0, \cdot\}$
$\mathrm{QR}(\cdot)$	QR decomposition

Table 1: Basic notation.

where $R_1, \ldots, R_N$ are called ranks of the TC model and the element-wise form of the following decomposition is given by

$$x_{i_1 i_2 \ldots i_N} = \mathrm{tr}(\mathbf{A}^{(1)}_{i_1}\mathbf{A}^{(2)}_{i_2} \cdots \mathbf{A}^{(N)}_{i_N}), \tag{2}$$

and $\mathbf{A}^{(n)}_{i_n} \in \mathbb{R}^{R_n \times R_{n+1}}$ represents the i_n-th frontal slice of the core tensors $\underline{\mathbf{A}}^{(n)} \in \mathbb{R}^{R_n \times I_n \times R_{n+1}}$ for $n = 1, \ldots, N$ and $R_{N+1} = R_1$ and $a^{(n)}_{r_n r_{n+1}}$ are tubes of $\underline{\mathbf{A}}^{(n)}$.

3.2 Document modelling setting

In our research, we use the fact that the same text can be represented in different ways via different sets of n-grams with fixed lengths $\{\mathbb{W}^n\}_{n=1}^N$, where $\mathbb{W}^n = \underbrace{\mathbb{W} \times \cdots \times \mathbb{W}}_{n}$ and $\mathbb{W}$ is the word set. The main hypothesis is that the occurance statistics of the each of these n-grams sets contains some new information about this text, which can not be extracted from any other n-grams set. If we combine information from all of these sets we can achieve better quality for our document embedding model.

Due to the fact that the occurrence of the sequence of words depends on the occurrence of each word from this sequence, it is reasonable to treat the distribution of each fixed-length n-grams set separately. Otherwise, by the reason of dependence between the length of word sequence and their frequency, small length n-grams can downweight the importance of long length n-grams. Thus the effect of higher-order interaction can become low. Also, we notice that consider $p(\mathbf{w})$ as a distribution over unordered sets is a quite restrictive assumption on the structure of the model due to the importance of the order of the words in the language semantics. For all these reasons, we work with each n-gram distribution as with the separate distribution of the single random variable rather than define joint distribution for all n-grams sets and assume that particular distribution for each n-gram set can be constructed through marginalization from this joint distribution.

Following this intuition for each n-gram set, n, we assign appropriate joint distribution, $p(\mathbf{w}, d)$, where $\mathbf{w} \in \mathbb{W}^n$ and $d \in \mathbb{D}$. We represent co-occurrence of each n-gram, $\mathbf{w} = (w_1, \ldots, w_n)$, and each document, $d \in \mathbb{D}$, as $(n+1)$th-order tensor $\underline{\mathbf{X}}^{(\mathbf{w}d)} \in \mathbb{R}^{|\mathbb{W}| \times \cdots \times |\mathbb{W}| \times |\mathbb{D}|}$, where

$$x^{(\mathbf{w}d)}_{\mathbf{i}j} = \begin{cases} 1, & \text{if } \mathbf{i} = \mathbf{w}, j = d \\ 0, & \text{otherwise}. \end{cases} \tag{3}$$

Then we represent probability $p(\mathbf{w}, d)$ as the mean of these tensors

$$\bar{\underline{\mathbf{X}}}^{(n)} = \mathbb{E}_{(\mathbf{w},d)\sim p(\mathbf{w},d)} \underline{\mathbf{X}}^{(\mathbf{w}d)}.$$

Note that co-occurrences of n-grams and documents define bipartite graphs between them and $\bar{\underline{\mathbf{X}}}^{(n)}$ can be interpreted as adjacency tensors of these graphs.

3.3 Proposed model

Following compositional matrix-space modelling approach we represent each word, $w \in \mathbb{W}$, as a matrix $\mathbf{U}_w \in \mathbb{R}^{R \times R}$, a n-gram, $\mathbf{w} \in \mathbb{W}^n$, as $\mathbf{U}_\mathbf{w} = \prod_{k=1}^n \mathbf{U}_{w_k}$, and each document, $d \in \mathbb{D}$, as a matrix $\mathbf{V}_d \in \mathbb{R}^{R \times R}$. To measure dependence between n-gram, $\mathbf{w}$, and document, d, we use the Frobenius inner product, defined as $\langle \mathbf{U}_\mathbf{w}, \mathbf{V}_d \rangle_F = \mathrm{tr}(\mathbf{U}_\mathbf{w}^T \mathbf{V}_d) = \mathrm{tr}(\mathbf{U}_\mathbf{w} \mathbf{V}_d^T) = \mathrm{vec}(\mathbf{U}_\mathbf{w})^T \mathrm{vec}(\mathbf{V}_d)$. We assume that embeddings organized accordingly to this operation can be suitable for linear classifiers.

The resulting model is the generalized to exponential family of probability distributions coupled tensor decomposition (Collins et al., 2001; Yilmaz et al., 2011) of the set of tensors $\{\bar{\underline{\mathbf{X}}}^{(n)}\}_{n=1}^N$ by the corresponding set of tensor chain models with restricted set of parameters

$$\Theta = \{\mathbf{U}_w\}_{w=1}^{|\mathbb{W}|} \cup \{\mathbf{V}_d\}_{d=1}^{|\mathbb{D}|}, \tag{4}$$

in the following optimization problem

$$\min_\Theta \sum_{n=1}^N \mathrm{KL}[\bar{\underline{\mathbf{X}}}^{(n)} || \hat{\underline{\mathbf{X}}}^{(n)}; \Theta], \tag{5}$$

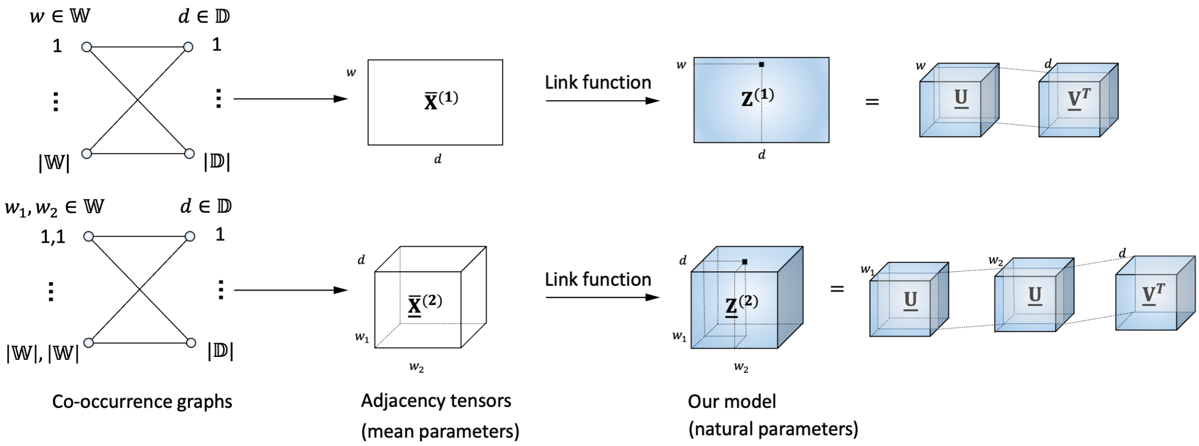

Figure 1: Illustration of representation of document collection in a multi-view way as a collection of bipartite graphs. Each of these graphs represents dependence (number of co-occurrence) between word's strings of length n and documents. Adjacency matrices $\bar{\mathbf{X}}^{(n)} \in \mathbb{R}^{|\mathbb{W}|^n \times |\mathbb{D}|}$ of these graphs can be appropriately tensorized to adjacency tensors $\underline{\bar{\mathbf{X}}}^{(n)} \in \mathbb{R}^{|\mathbb{W}| \times \cdots \times |\mathbb{W}| \times |\mathbb{D}|}$ which can be linked through modified multinomial link function with latent tensors $\underline{\mathbf{Z}}^{(n)} \in \mathbb{R}^{|\mathbb{W}| \times \cdots \times |\mathbb{W}| \times |\mathbb{D}|}$. Latent tensors can be decomposed via the Coupled Tensor Chain model. In our model, all core tensors $\underline{\mathbf{U}}$ ($\underline{\mathbf{V}}^T$) which represent words (documents) are additionally restricted to have the same parameters $\{\mathbf{U}_w\}_{w=1}^{|\mathbb{W}|}$ ($\{\mathbf{V}_d^T\}_{d=1}^{|\mathbb{D}|}$).

where each Kullback-Leibler divergence can be expressed as

$$\mathrm{KL}[\underline{\bar{\mathbf{X}}}^{(n)} || \underline{\hat{\mathbf{X}}}^{(n)}; \Theta] = \sum_{\mathbf{w} \in \mathbb{W}^n} \sum_{d \in \mathbb{D}} \bar{x}_{\mathbf{w}d}^{(n)} \log \left(\frac{\bar{x}_{\mathbf{w}d}^{(n)}}{\hat{x}_{\mathbf{w}d}^{(n)}} \right)$$

$$= \sum_{w_1=1}^{|W|} \cdots \sum_{w_n=1}^{|W|} \sum_{d=1}^{|D|} \bar{x}_{w_1 \ldots w_n d}^{(n)} \log \left(\frac{\bar{x}_{w_1 \ldots w_n d}^{(n)}}{\hat{x}_{w_1 \ldots w_n d}^{(n)}} \right),$$

and $\hat{x}_{\mathbf{w}d}^{(n)} = p(\mathbf{w}, d)$.

Our model represents $p(\mathbf{w}, d)$ by using following mean function

$$p(\mathbf{w}, d) \propto p(\mathbf{w}) p(d) \exp(\mathrm{vec}(\underline{\mathbf{X}}^{(\mathbf{w}d)})^T \mathrm{vec}(\underline{\mathbf{Z}}^{(n)})),$$

where $\mathrm{vec}(\underline{\mathbf{X}}^{(\mathbf{w}d)})^T \mathrm{vec}(\underline{\mathbf{Z}}^{(n)}) = z_{\mathbf{w}d}^{(n)}$ is one of natural parameters, which organized in the latent tensors $\underline{\mathbf{Z}}^{(n)} \in \mathbb{R}^{|\mathbb{W}| \times \cdots \times |\mathbb{W}| \times |\mathbb{D}|}$. This latent tensors contain pointwise mutual information between n-grams and documents and we assume that each of this tensor has low tensor chain rank, i.e., $z_{\mathbf{w}d}^{(n)} = \mathrm{tr}(\mathbf{U}_{\mathbf{w}} \mathbf{V}_d^T)$.

3.4 Intuition from geometric interpretation

If we avoid generative assumptions (Saunshi et al., 2019), our task can be interpreted as maximizing the similarity between document d and n-grams from its document distribution $p(\mathbf{w}|d)$ with simultaneous minimization of similarity between this document and n-grams from common n-gram distribution $p(\mathbf{w})$. As shown in previous works (Kumar and Tsvetkov, 2019; Meng et al., 2019) enforcing the spherical geometry constraints is a promising choice for tasks focusing on directional similarity. For doing so it can be reasonable to constrain our model to the orthogonal group. In this case Frobenius inner product became proper similarity measure and the sequential matrix product always preserves fixed norm and group structure (i.e., invertibility of matrix multiplication). Due to the group structure, our model has an interesting property to uniquely determine each word in the n-gram by their left and right aggregated context matrices and general n-gram matrix.

However, orthogonal group is a disjoint set of two connected components: set of rotations

$$\mathrm{SO}(R) = \{\mathbf{A} | \mathbf{A}^T\mathbf{A} = \mathbf{A}\mathbf{A}^T = \mathbf{I}, \det(\mathbf{A}) = +1\}$$

and set of reflections $\{\mathbf{A} | \mathbf{A}^T\mathbf{A} = \mathbf{A}\mathbf{A}^T = \mathbf{I}, \det(\mathbf{A}) = -1\}$. We impose constraints on our model parameters enforcing rotation matrices since the product of any number of rotations is always rotation, i.e. rotation set forms a matrix group. While the product of an even number of reflections becomes rotation.

3.5 Noise-contrastive estimation

In practice we do not need to construct set of tensors $\{\underline{\bar{\mathbf{X}}}^{(n)}\}_{n=1}^N$ explicitly. Instead, since each $\underline{\bar{\mathbf{X}}}^{(n)}$ represents a higher-order frequency table, we can optimize the sum of MLE tasks:

$$\mathbb{E}_{w,d\sim\underline{\mathbf{X}}^{(1)}}\log(\hat{x}^{(1)}_{wd})+\cdots+\mathbb{E}_{\mathbf{w},d\sim\underline{\mathbf{X}}^{(N)}}\log(\hat{x}^{(N)}_{\mathbf{w}d}).$$

Usually, we have huge amount of data which make problem of computing partition function for each $\hat{x}^{(n)}_{\mathbf{w}d}$ intractable for many current computing architectures. We can avoid this problem by using noise-contrastive estimation (Gutmann and Hyvärinen, 2012) for conditional model (Ma and Collins, 2018). Similar to Chen et al. (2017), we construct negative samples from our batch by connecting non-linked n-grams and documents. Finally, for parameter set

$$\Theta=\{\mathbf{U}_w\}_{w=1}^{|\mathbb{W}|}\cup\{\mathbf{V}_d\}_{d=1}^{|\mathbb{D}|}\cup\{\kappa^{(n)}\}_{n=1}^{N},\qquad(6)$$

we formulate optimization problem in the following way:

$$\begin{aligned}\min_{\Theta}\ &\sum_{n=1}^{N}\mathcal{L}^{(n)}(\Theta)+\frac{\lambda}{2}\sum_{n=1}^{N}(\kappa^{(n)})^2\\ \text{s.t.}\ &\mathbf{U}_w\in\mathrm{SO}(R),\ \ w=1,\ldots,|\mathbb{W}|\\ &\mathbf{V}_d\in\mathrm{SO}(R),\ \ d=1,\ldots,|\mathbb{D}|\\ &\kappa^{(n)}\geq 0,\ \ n=1,\ldots,N,\end{aligned}\qquad(7)$$

where each risk function is equal to

$$\begin{aligned}\mathcal{L}^{(n)}(\Theta)&=\mathbb{E}_{\{(\mathbf{w}_i,d_i)\}_{i=1}^{I}\sim\prod_{i=1}^{I}\bar{x}^{(n)}_{\mathbf{w}_i d_i}}\\ &-\log\frac{\exp\left(\kappa^{(n)}\mathrm{tr}(\mathbf{U}_{\mathbf{w}_i}\mathbf{V}_{d_i}^T)\right)}{\sum_{j=1}^{I}\exp\left(\kappa^{(n)}\mathrm{tr}(\mathbf{U}_{\mathbf{w}_j}\mathbf{V}_{d_i}^T)\right)}.\end{aligned}\qquad(8)$$

We add concentration parameters $\kappa^{(n)}$ to our loss function to overcome the problem of fixed scale. This makes our model more flexible to represent sharp distributions. Due to the fact that each n-gram distribution has its own scale, it can be reasonable to have a different $\kappa^{(n)}$ for different n-gram distributions.

4 Optimization setup

4.1 N-gram construction

We construct n-grams from text corpora by using sequentially moving of the sliding window of length n (from 1 to N) inside each document.

4.2 Parameters initialization

Initialization from the uniform distribution on the Stiefel manifold (Saxe et al., 2014) is one of promising ways to initialize deep neural network. To initialize parameters only from rotation component of Stiefel manifold we can swap two columns for each parameter matrix if the determinant of the this matrix is -1. However, this initialization can be below optimal way, because these rotation matrices can be far away from each other and due to the non-trivial structure of the loss function on this manifold we can stuck in local minima. To overcome this problem, we can fix particular point on the manifold for all matrices and perform small movement from this point in arbitrary direction. We use following strategy for each parameter $\mathbf{A}\in\{\mathbf{U}_w\}_{w=1}^{|\mathbb{W}|}\cup\{\mathbf{V}_d\}_{d=1}^{|\mathbb{D}|}$:

$$\begin{aligned}&\mathbf{A}_0\sim\mathcal{N}\left(0,\frac{1}{R^2}\right)\\ &[\mathbf{Q},\mathbf{R}]=\mathrm{QR}\left(\mathbf{I}+\frac{1}{2}(\mathbf{A}_0-\mathbf{A}_0^T)\right)\\ &\mathbf{A}=\mathbf{Q}.\end{aligned}\qquad(9)$$

We initialize all concentration parameters using following equation $\kappa^{(n)}=\frac{u}{R}$, where $u\sim\mathcal{U}(0.9,1.1)$ and $n=1,\ldots,N$.

4.3 Riemannian optimization

We solve our problem on the product manifold of rotation group and nonnegative orthant by simultaneous optimization of model parameters $\{\mathbf{U}_w\}_{w=1}^{|\mathbb{W}|}\cup\{\mathbf{V}_d\}_{d=1}^{|\mathbb{D}|}$ and $\{\kappa^{(n)}\}_{n=1}^{N}$ by Riemannian (Bécigneul and Ganea, 2019) and projected Adagrad (Duchi et al., 2011) respectively.

Let $\mathcal{M}$ be a real smooth manifold and $\mathcal{L}:\mathcal{M}\to\mathbb{R}$ a smooth real-valued function over parameters $\boldsymbol{\theta}\in\mathcal{M}$. Riemannian gradient descent (Gabay, 1982; Absil et al., 2008) based on two sequential steps. At first we compute Riemannian gradient by orthogonal projection of the Euclidean gradient on the tangent space at the same point on which we compute Euclidean gradient by $\mathrm{proj}_{\boldsymbol{\theta}_t}:\mathbb{R}\to\mathrm{T}_{\boldsymbol{\theta}}\mathcal{M}$

$$\nabla_R\mathcal{L}(\boldsymbol{\theta}_t)=\mathrm{proj}_{\boldsymbol{\theta}_t}(\nabla_E\mathcal{L}(\boldsymbol{\theta}_t)),\qquad(10)$$

and then we perform movement on the manifold by specific curve, which called retraction $\mathrm{R}_{\boldsymbol{\theta}_t}:\mathrm{T}_{\boldsymbol{\theta}}\mathcal{M}\to\mathcal{M}$

$$\boldsymbol{\theta}_{t+1}=\mathrm{R}_{\boldsymbol{\theta}_t}(-\alpha\nabla_R\mathcal{L}(\boldsymbol{\theta}_t)).\qquad(11)$$

For optimization on rotation group the orthogonal projector of matrix $\mathbf{G}\in\mathbb{R}^{R\times R}$ on the tangent space at the point $\mathbf{A}\in\mathrm{SO}(R)$ is given by:

$$\mathrm{proj}_{\mathbf{A}}(\mathbf{G})=\frac{1}{2}(\mathbf{G}-\mathbf{A}\mathbf{G}^T\mathbf{A}),\qquad(12)$$

For movement on the manifold in this direction we use QR-based retraction:

$$\begin{aligned}&[\mathbf{Q}, \mathbf{R}] = \mathrm{QR}\left(\mathbf{A}_t - \alpha \nabla_R \mathcal{L}(\mathbf{A}_t)\right)\\ &\mathbf{A}_{t+1} = \mathbf{Q}.\end{aligned} \tag{13}$$

We choose the QR-based retraction because it allows Riemannian Adagrad to achieve the fastest convergence in our experiments in comparison with Cayley retraction (first-order), Polar retraction (second-order), and geodesic (matrix exponential).

Algorithm 1 Optimization algorithm for RDM

```
Input: Learning rates $\alpha$ and $\beta$, number of iterations $T$, maximum n-gram length $N$.
Output: Embedding parameters $\{\mathbf{U}_w\}_{w=1}^{|\mathbb{W}|} \cup \{\mathbf{V}_d\}_{d=1}^{|\mathbb{D}|}$ and $\{\kappa^{(n)}\}_{n=1}^{N}$.
for $t = 1$ to $T$ do   ▷ after computing gradients:
    for all $\mathbf{A}_t \in \{\mathbf{U}_w\}_{w=1}^{|\mathbb{W}|} \cup \{\mathbf{V}_d\}_{d=1}^{|\mathbb{D}|}$ do
        $\nabla_E \mathcal{L}(\mathbf{A}_t) = \sum_{n=1}^{N} \nabla_E \mathcal{L}^{(n)}(\mathbf{A}_t)$
        $\nabla_R \mathcal{L}(\mathbf{A}_t) \leftarrow$ (12) with $\nabla_E \mathcal{L}(\mathbf{A}_t)$
        $\alpha_t = \frac{\alpha}{\sqrt{\sum_{i=1}^{t} \|\nabla_R \mathcal{L}(\mathbf{A}_i)\|_F^2}}$
        $\mathbf{A}_{t+1} \leftarrow$ (13) with $\alpha_t$
    end for
    for $n = 1$ to $N$ do
        $\mathrm{grad}(\kappa_t^{(n)}) = \nabla_E \mathcal{L}^{(n)}(\kappa_t^{(n)}) + \lambda \kappa_t^{(n)}$
        $\kappa_{t+\frac{1}{2}}^{(n)} = \kappa_t^{(n)} - \beta \frac{\mathrm{grad}(\kappa_t^{(n)})}{\sqrt{\sum_{i=1}^{t} \mathrm{grad}^2(\kappa_i^{(n)})}}$
        $\kappa_{t+1}^{(n)} = \left[\kappa_{t+\frac{1}{2}}^{(n)}\right]_+$
    end for
end for
```

4.4 Computational efficiency

The computational complexity of our model depends on the complexity R^3 of multiplication of matrices of size $R \times R$, and QR decomposition $\frac{4}{3}R^3$ (Layton and Sussman, 2020; Trefethen and III, 1997). Due to the number of elements in these matrices $d = R^2$, we can transform the complexity of our model in the dimension of embedding. In this view, the time complexity is $O(kd^{1.5})$, where k is the size of the context window. As shown in Table (2), our model has computational benefits in comparison with Transformer due to the linear dependence of time complexity on the word sequence length. We note that in comparison with Bi-LSTM models like ELMo, our model has lower complexity on embedding dimension, and can be computed in parallel using the associativity property of matrix multiplication. Although our model has a higher theoretical time complexity than the vector space models, the real gap between them is relatively small at ordinary embedding dimension ($\sim$ 400).

Method	**Time**	**Space**
PV-DBOW	$O(kd)$	$O(d)$
PV-DM (Concat)	$O(kd)$	$O(kd)$
ELMo	$O(kld^2)$	$O(ld^2)$
BERT	$O(k^2hld)$	$O(khld)$
RDM (Ours)	$O(kd^{1.5})$	$O(d)$

Table 2: Comparison of time and space complexity of several document embedding models, where k - size of context window, d - embedding dimension, l - number of layers, h - number of heads. The time complexity of other discussed here vector space models is equivalent to the complexity of PV-DBOW.

5 Numerical experiments

5.1 Experimental setup

Downstream Linear Protocol. We estimate quality of pre-trained representations on the multi-class document categorization tasks on the 20 Newsgroups and the ArXiv based long document dataset (He et al., 2019). We choose these datasets for our benchmarks because they are significantly different in the document's average length. This implies that statistics of long n-grams differ between these datasets too and in the case of the ArXiv dataset statistics of n-grams are significantly better converged than in the case of 20 newsgroups. This fact allow us to hypothesize that matrix-space models should less overfit on the ArXiv dataset.

We fix the document embeddings and optimize multinomial logistic regression with SAGA optimizer and l_2-norm regularization. Instead of test set we use nested 10-fold cross-validation to estimate statistical significance using Wilcoxon signed-rank test (Japkowicz and Shah, 2011; Dror et al., 2018). For each fold we estimate the hyperparameter of l_2-regularization on 10 point logarithmic grid from 0.01 to 100 by using additional 10-fold cross-validation with macro-averaged F1 score. For text preprocessing, we use CountVectorizer from the Scikit-learn package. Additionally we remove words which occur in the NLTK stopwords list or occur only in 1 document. In the case of ArXiv dataset we use half of this dataset and use only documents in the range from 1000 to 5000 words (smaller documents are removed and bigger documents are reduced to the first 5000 words).

Dataset	#cls	$\|\mathbb{W}\|$	$\|\mathbb{D}\|$	#w
20 Newsgroups	20	75752	18846	180
ArXiv	6	251108	16371	3829

Table 3: Datasets considered in the paper, where #cls - number of classes, #w - average number of words in documents, $|\mathbb{W}|$ - vocabulary size, $|\mathbb{D}|$ - number of documents.

Baselines. We compare our model with different vector space models: paragraph vectors (Le and Mikolov, 2014), weighted combinations of the word2vec skipgram vectors (Mikolov et al., 2013) (average, TF-IDF and SIF (Arora et al., 2017)), Doc2vecC (Chen, 2017), sent2vec (Pagliardini et al., 2018) and recently proposed JoSe (Meng et al., 2019). The comparison with the last two of these models seems to be more informative than with others because of some similarity of these models to our model (sent2vec can use n-gram information and JoSe also based on the spherical type of embedding geometry).

Due to the large number of possible values of hyperparameters for each model, we used the default values proposed by the authors of these models or proposed in subsequent studies of these models like in the case of paragraph vectors (Lau and Baldwin, 2016). We modify only the min_count to 1 and window size to be equal to the number of negative samples for paragraph2vec and word2vec models because it gives better results for these methods in our experiments. We choose n-grams number equal to 1 for sent2vec, because other values doesn't improve results. To preserve the fairness of comparison we use fixed embedding dimension, number of negative samples, and number of epochs for all models including ours. These values try to mimic the usual values of these hyperparameters in practice.

We compare our model not only with vector models but also with neural network models. We use 5.5B ELMo (Peters et al., 2018) version which is pre-trained on Wikipedia (1.9B) and all of the monolingual news crawl data from WMT 2008-2012 (3.6B). ELMo embedding dimension is equal to 1024. Also we use 768-dimensional embedding vectors from BERT model "bert-base-uncased". Following Devlin et al. (2019) we take the last layer hidden state corresponding to the [CLS] token as the aggregate document representation. If length of document is bigger than 512 we cut document on 512-length parts and average representation of this parts. Finally, we add Sentence BERT (Reimers and Gurevych, 2019) to the baseline models. This model is fine-tuned on SNLI and MultiNLI datasets for sentence embedding generation. We use 768-dimensional embedding vectors from model "bert-base-nli-mean-tokens".

We do not perform fine-tuning of BERT and ELMo models for our datasets, because in our experiments it doesn't give any positive effect on the final performance of these models. However, this is not true for Sentence BERT. Fine-tuning slightly improve performance of this model on the 20 newsgroup (50 epoch with maximum margin triplet loss).

We should notice that this experimental design gives some benefits to neural network models in comparison with log-multilinear models, but it is more consistent with the ordinary practical use case of the Transformers and RNN models. However, the next experiment will show that log-multilinear models can still outperform pre-trained neural networks.

Ablation study. For ablation study, we use different settings of our model. RDM means rotation document model, i.e. our model. RDM-R means our model without rotation group constraints. By (1) we mean model which utilize only unigrams and documents co-occurance information. By (3) and (5) we mean model which utilizes information from (1, 2, 3)-grams and (1, 2, 3, 4, 5)-grams respectively. For RDM, we use 1e-2 and 1e-3 as learning rates of Radagrad and projected Adagrad respectively and we use $\lambda = 15$ for 20 newsgroups and change λ to 5 for ArXiv dataset due to smaller number of epoch in this experiment. For RDM-R we use 1e-2 as learning rate for Adagrad.

5.2 Experimental results and comparison of performance

Comparison to baselines. As one may observe in the Table (4), our models yield results comparable or outperforming the baseline methods, including the simpler log-multilinear models (e.g. Skipgram) and more complex models featuring nonlinear transformations, such as recurrences (ELMo)

Models	20 Newsgroups				ArXiv			
	Acc	**Prec**	**Rec**	**F1**	**Acc**	**Prec**	**Rec**	**F1**
PV-DBOW	88.7	88.4	88.2	88.2	89.1	89.2	89.2	89.2
PV-DM	77.2	76.8	76.5	76.5	42.2	42.3	42.0	41.9
Skipgram+Average	90.3	90.2	90.0	90.0	92.4	92.3	92.3	92.3
Skipgram+TF-IDF	90.4	90.2	90.1	90.1	92.6	92.5	92.4	92.4
Skipgram+SIF	90.4	90.2	90.1	90.1	92.4	92.3	92.3	92.3
Sent2vec	87.9	87.6	87.5	87.5	91.9	91.7	91.7	91.7
Doc2vecC	90.0	89.8	89.7	89.7	93.2	93.1	93.1	93.1
JoSe	87.8	87.6	87.4	87.4	91.3	91.3	91.2	91.2
ELMo	79.2	78.8	78.8	78.7	91.6	91.5	91.4	91.4
BERT	74.3	73.6	73.6	73.5	92.3	92.2	92.1	92.1
Sentence BERT	79.5	79.1	79.0	79.0	89.0	88.9	88.8	88.8
RDM-R (1)	86.8	86.4	86.3	86.3	94.0*	93.9*	93.8*	93.8*
RDM-R (3)	87.9	87.6	87.4	87.4	94.5*	**94.4***	**94.4***	**94.4***
RDM-R (5)	88.3	88.0	87.9	87.9	**94.6***	**94.4***	**94.4***	**94.4***
RDM (1)	89.3	89.2	89.0	89.0	94.0*	93.9*	93.9*	93.9*
RDM (3)	90.7	90.5	90.4	90.4	94.0*	93.9*	93.9*	93.9*
RDM (5)	**91.1***	**90.9***	**90.8***	**90.8***	94.0*	93.9*	93.9*	93.9*

Table 4: Text classification performance on the 20 Newsgroups (short documents) and on the modified ArXiv (long documents) datasets. We fix the number of epochs to 50, the embedding dimension to 400, and the number of negative samples to 15 for all models on the 20 Newsgroups. On the ArXiv dataset, we use the same hyperparameters except for only the number of epochs which is equal to 5. We use macro-average for Precision, Recall, and F1.

and transformer blocks (BERT, SentenceBERT). More specifically, on the 20 newsgroups, the RDM (5) model yields the best results significantly[2] outperforming all the listed baseline approaches. It is interesting that contrary to the 20 Newsgroups, all RDM variants with any number of the n-grams sets show strong results and significantly outperform other models on the ArXiv dataset. Weighted combinations of the skipgram vectors and doc2vecC model achieve the closest to our result. This confirms that neural models like ELMo and BERT, are not the best way for all datasets and log-multilinear models can outperform them. We can see that performance of nonlinear models increase if we use a large document dataset and BERT can outperform some of the log-multilinear approaches (PV-DBOW, JoSe and sent2vec), but still, its result is not on the top level.

Comparison between our models. On observing the results we can see that our model increase the performance of classification when adding the n-gram set with a bigger length. This property has both models with rotation group constraints and without such constraints. Despite this fact, as we can see the model without rotation constraints is less robust in respect to noisy statistics of small document dataset and achieve performance less than the PV-DBOW model, while the rotation group model outperforms all other models. However, once we move on to a dataset with a larger document's average length (ArXiv), RDM-R performs better than all other models, including RDM. This confirms our hypothesis that the existence of good statistics of long n-grams has critical value for matrix space models. Due to strong associativity between sets of n-grams, our model needs more parameters to approximate all the co-occurrences. The RDM-R has a higher number of degrees of freedom than the RDM R^2 vs $\frac{R(R-1)}{2}$. We think that it allows RDM-R to outperform RDM in this experiment. This intuition can be also confirmed by the fact that RDM (1) and RDM-R (1) have the same performance level. And only if we increase the number of n-grams for the model from 1 to 3, then RDM-R can achieve better performance. However, if we increase the number of n-grams set from 3 to 5 both models stay on the same level of performance. This is the sign that we need to use a bigger embedding dimension if we want to achieve even better results.

[2] We use $*$ if the comparison of our model with the baseline models has p $<$ 0.05.

In summary, if we have short documents, it's better to use RDM. For long document dataset with restriction on the embedding dimension, we suggest to relax the rotation group constraints. This trick allows RDM to use more degrees of freedom to estimate data precisely.

6 Conclusion

In this paper, we proposed a novel unsupervised representation learning method based on the generalized tensor chain with rotation group constraints, which can utilize higher-order word interactions and preserve most part of the computational efficiency and interpretability of vector-based models. Our model achieves state-of-the-art results in the document classification benchmarks on the 20 newsgroups and modified ArXiv dataset. A further direction of research could be focused on adding tensor kernel functions to the model to eliminate problems with dependence on the dimension of embedding. It could be interesting to augment this type of model with the different loss functions based not only on n-gram-document interactions but also on word-word interactions from the knowledge graph or document-document interaction from the citation graph of the documents.

Acknowledgments

The Zhores supercomputer (Zacharov et al., 2019) was used for computations. This research was partially supported by the Ministry of Education and Science of the Russian Federation (grant 14.756.31.0001). The work of Alexander Panchenko was partially supported by the MTS-Skoltech laboratory.

References

Pierre-Antoine Absil, Robert E. Mahony, and Rodolphe Sepulchre. 2008. *Optimization Algorithms on Matrix Manifolds*. Princeton University Press.

Sanjeev Arora, Yuanzhi Li, Yingyu Liang, Tengyu Ma, and Andrej Risteski. 2016. A latent variable model approach to PMI-based word embeddings. *Transactions of the Association for Computational Linguistics*, 4:385–399.

Sanjeev Arora, Yingyu Liang, and Tengyu Ma. 2017. A simple but tough-to-beat baseline for sentence embeddings. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Shima Asaadi and Sebastian Rudolph. 2017. Gradual learning of matrix-space models of language for sentiment analysis. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 178–185, Vancouver, Canada. Association for Computational Linguistics.

Eric Bailey and Shuchin Aeron. 2017. Word embeddings via tensor factorization. *CoRR*, abs/1704.02686.

Gary Bécigneul and Octavian-Eugen Ganea. 2019. Riemannian adaptive optimization methods. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Minmin Chen. 2017. Efficient vector representation for documents through corruption. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Ting Chen, Yizhou Sun, Yue Shi, and Liangjie Hong. 2017. On sampling strategies for neural network-based collaborative filtering. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, August 13 - 17, 2017*, pages 767–776. ACM.

Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar. 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 [Neural Information Processing Systems: Natural and Synthetic, NIPS 2001, December 3-8, 2001, Vancouver, British Columbia, Canada]*, pages 617–624. MIT Press.

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. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Bhuwan Dhingra, Christopher Shallue, Mohammad Norouzi, Andrew Dai, and George Dahl. 2018. Embedding text in hyperbolic spaces. In *Proceedings of the Twelfth Workshop on Graph-Based Methods for Natural Language Processing (TextGraphs-12)*, pages 59–69, New Orleans, Louisiana, USA. Association for Computational Linguistics.

Rotem Dror, Gili Baumer, Segev Shlomov, and Roi Reichart. 2018. The hitchhiker's guide to testing statistical significance in natural language processing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1383–1392, Melbourne, Australia. Association for Computational Linguistics.

John C. Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *J. Mach. Learn. Res.*, 12:2121–2159.

Abraham Frandsen and Rong Ge. 2019. Understanding composition of word embeddings via tensor decomposition. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Daniel Gabay. 1982. Minimizing a differentiable function over a differential manifold. *Journal of Optimization Theory and Applications*, 37:177–219.

Michael Gutmann and Aapo Hyvärinen. 2012. Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics. *J. Mach. Learn. Res.*, 13:307–361.

Jun He, Liqun Wang, Liu Liu, Jiao Feng, and Hao Wu. 2019. Long document classification from local word glimpses via recurrent attention learning. *IEEE Access*, 7:40707–40718.

Nathalie Japkowicz and Mohak Shah, editors. 2011. *Evaluating Learning Algorithms: A Classification Perspective*. Cambridge University Press.

Boris N. Khoromskij. 2011. O(dlogn)-quantics approximation of n-d tensors in high-dimensional numerical modeling. *Constructive Approximation*, 34:257–280.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar. Association for Computational Linguistics.

Sachin Kumar and Yulia Tsvetkov. 2019. Von mises-fisher loss for training sequence to sequence models with continuous outputs. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270, San Diego, California. Association for Computational Linguistics.

Jey Han Lau and Timothy Baldwin. 2016. An empirical evaluation of doc2vec with practical insights into document embedding generation. In *Proceedings of the 1st Workshop on Representation Learning for NLP*, pages 78–86, Berlin, Germany. Association for Computational Linguistics.

William Layton and Myron Sussman. 2020. *Numerical Linear Algebra*. WSPC.

Quoc V. Le and Tomás Mikolov. 2014. Distributed representations of sentences and documents. In *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*, volume 32 of *JMLR Workshop and Conference Proceedings*, pages 1188–1196. JMLR.org.

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

Zhuang Ma and Michael Collins. 2018. Noise contrastive estimation and negative sampling for conditional models: Consistency and statistical efficiency. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 3698–3707. Association for Computational Linguistics.

Florian Mai, Lukas Galke, and Ansgar Scherp. 2019. CBOW is not all you need: Combining CBOW with the compositional matrix space model. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Yu Meng, Jiaxin Huang, Guangyuan Wang, Chao Zhang, Honglei Zhuang, Lance M. Kaplan, and Jiawei Han. 2019. Spherical text embedding. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 8206–8215.

Tomás Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013,* *Lake Tahoe, Nevada, United States*, pages 3111–3119.

Maximilian Nickel and Douwe Kiela. 2018. Learning continuous hierarchies in the lorentz model of hyperbolic geometry. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 3776–3785. PMLR.

Maximillian Nickel and Douwe Kiela. 2017. Poincaré embeddings for learning hierarchical representations. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 6338–6347. Curran Associates, Inc.

Matteo Pagliardini, Prakhar Gupta, and Martin Jaggi. 2018. Unsupervised learning of sentence embeddings using compositional n-gram features. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 528–540, New Orleans, Louisiana. Association for Computational Linguistics.

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. Association for Computational Linguistics.

David Perez-Garcia, Frank Verstraete, Michael M. Wolf, and J. Ignacio Cirac. 2007. Matrix product state representations. *Quantum Info. Comput.*, 7(5):401–430.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Sebastian Rudolph and Eugenie Giesbrecht. 2010. Compositional matrix-space models of language. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 907–916, Uppsala, Sweden. Association for Computational Linguistics.

Nikunj Saunshi, Orestis Plevrakis, Sanjeev Arora, Mikhail Khodak, and Hrishikesh Khandeparkar. 2019. A theoretical analysis of contrastive unsupervised representation learning. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 5628–5637. PMLR.

Andrew M. Saxe, James L. McClelland, and Surya Ganguli. 2014. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*.

Vatsal Sharan and Gregory Valiant. 2017. Orthogonalized ALS: A theoretically principled tensor decomposition algorithm for practical use. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 3095–3104, International Convention Centre, Sydney, Australia. PMLR.

Lloyd N. Trefethen and David Bau III. 1997. *Numerical Linear Algebra*, first edition. SIAM: Society for Industrial and Applied Mathematics.

Ainur Yessenalina and Claire Cardie. 2011. Compositional matrix-space models for sentiment analysis. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 172–182, Edinburgh, Scotland, UK. Association for Computational Linguistics.

Yusuf Kenan Yilmaz, Ali Taylan Cemgil, and Umut Simsekli. 2011. Generalised coupled tensor factorisation. In *Advances in Neural Information Processing Systems 24: 25th Annual Conference on Neural Information Processing Systems 2011. Proceedings of a meeting held 12-14 December 2011, Granada, Spain*, pages 2151–2159.

Igor Zacharov, Rinat Arslanov, Maksim Gunin, Daniil Stefonishin, Andrey Bykov, Sergey Pavlov, Oleg Panarin, Anton Maliutin, Sergey Rykovanov, and Maxim Fedorov. 2019. Zhores — petaflops supercomputer for data-driven modeling, machine learning and artificial intelligence installed in skolkovo institute of science and technology. *Open Engineering*, 9(1):512–520.

Qibin Zhao, Masashi Sugiyama, Longhao Yuan, and Andrzej Cichocki. 2019. Learning efficient tensor representations with ring-structured networks. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2019, Brighton, United Kingdom, May 12-17, 2019*, pages 8608–8612. IEEE.