Learning Efficient Tensor Representations with Ring Structure Networks

Qibin Zhao[1], Masashi Sugiyama[1,2], Longhao Yuan[1,4], Andrzej Cichocki[3]

[1]RIKEN Center for Advanced Intelligence Project, [2]The University of Tokyo, [3]Skolkovo Institute of Science and Technology, [4]Saitama Institute of Technology

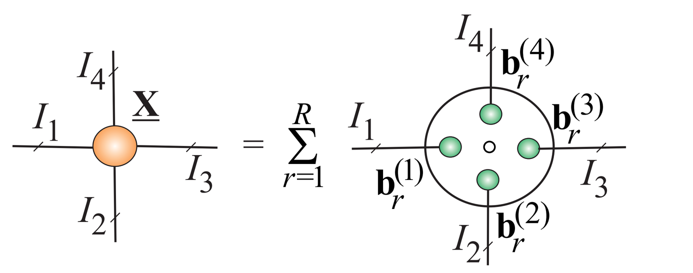
Background

- Tensor decompositions and tensor networks aim to represent high-dimensional data by multilinear operations of latent factors.
- Canonical polyadic (CP) decomposition represents a tensor as the sum of rank-one tensors by $\mathcal{O}(dnr)$ parameters, where d is the dimensions of tensor, n is the mode size, and r denotes the tensor rank.
- Tucker decomposition represents a tensor as a core tensor and several factor matrices by $\mathcal{O}(dnr + r^d)$ parameters.
- Tensor train (TT) decomposition represents a tensor as a set of third-order tensors by $\mathcal{O}(dnr^2)$ parameters.
- TT representation scales linearly to the tensor order as the CP model, and its solution can be easily computed as the Tucker model.
- Problems: TT has limited flexibility due to the rank $r_1 = r_{d+1} = 1$; TT-ranks have a fixed pattern; Permutations of data yield inconsistency.

Tensor Decompositions

- CP decomposition:

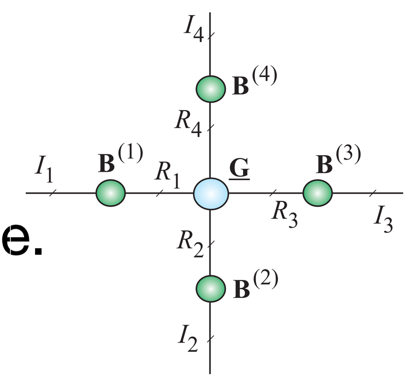
$$\underline{\mathbf{X}} = \sum_{r=1}^{R} \lambda_r \mathbf{b}_r^{(1)} \circ \mathbf{b}_r^{(2)} \circ \mathbf{b}_r^{(3)} \circ \mathbf{b}_r^{(4)}$$



'o' denotes the outer products of vectors, and R is CP-rank.

- Tucker decomposition:

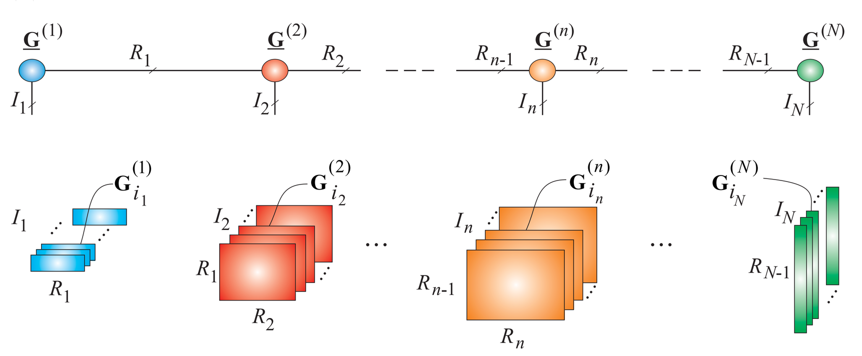
$$\underline{\mathbf{X}} = \underline{\mathbf{G}} \times_1 \mathbf{B}^{(1)} \times_2 \mathbf{B}^{(2)} \times_3 \mathbf{B}^{(3)} \times_4 \mathbf{B}^{(4)}$$



$\times_i$ denotes multilinear product on the ith mode. (R_1, R_2, R_3, R_4) are Tucker ranks.

- TT decomposition:

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



$(R_1, R_2, \ldots, R_{N-1})$ are TT-ranks.

Tensor Ring - Sequential SVDs

- Tensor ring (TR) decomposition can be performed by using sequential SVDs, which is called TR-SVD algorithm.

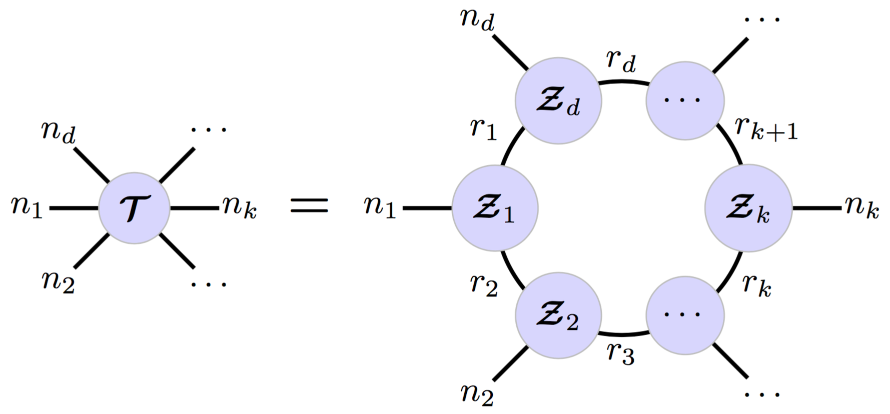
k-unfolding matrix

$$T_{\langle k \rangle}(\overline{i_1 \cdots i_k}, \overline{i_{k+1} \cdots i_d}) = \mathrm{Tr}\left\{ \prod_{j=1}^{k} \mathbf{Z}_j(i_j) \prod_{j=k+1}^{d} \mathbf{Z}_j(i_j) \right\} = \left\langle \mathrm{vec}\left(\prod_{j=1}^{k} \mathbf{Z}_j(i_j) \right), \mathrm{vec}\left(\prod_{j=d}^{k+1} \mathbf{Z}_j^T(i_j) \right) \right\rangle.$$

$$T_{\langle k \rangle}(\overline{i_1 \cdots i_k}, \overline{i_{k+1} \cdots i_d}) = \sum_{\alpha_1 \alpha_{k+1}} Z^{\leq k}(\overline{i_1 \cdots i_k}, \overline{\alpha_1 \alpha_{k+1}}) Z^{>k}(\overline{\alpha_1 \alpha_{k+1}}, \overline{i_{k+1} \cdots i_d}),$$

Tensor Ring Decomposition

- A generalization of TT without limitation of rank $r_1 = r_{d+1} = 1$.
- The additional operation between the first and last core tensors is added, yielding a circular tensor products of a set of cores.



$$T(i_1, i_2, \ldots, i_d) = \mathrm{Tr}\{\mathbf{Z}_1(i_1)\mathbf{Z}_2(i_2) \cdots \mathbf{Z}_d(i_d)\} = \mathrm{Tr}\left\{ \prod_{k=1}^{d} \mathbf{Z}_k(i_k) \right\}$$

$\mathbf{Z}_k(i_k)$ denotes i_kth slice matrix of core tensor $\boldsymbol{\mathcal{Z}}_k$.

- TR representation is equivalent to the sum of TTs with partially shared core tensors.
- TR-ranks $r_1 r_{k+1} \leq R_k$ where R_k is the rank of k-unfolding matricization of original tensor.
- TR solution is invariant to circularly dimensional permutation.

Tensor Ring - Stochastic Gradient Descent

- For large-scale datasets, stochastic gradient descent (SGD) shows high computational efficiency and scalability for matrix/tensor factorization.
- We develop a scalable and efficient TR decomposition by using SGD, which is also suitable for online learning and tensor completion problems.

$$L(\boldsymbol{\mathcal{Z}}_1, \boldsymbol{\mathcal{Z}}_2, \ldots, \boldsymbol{\mathcal{Z}}_d) = \frac{1}{2} \sum_{i_1, \ldots, i_d} \left\{ T(i_1, i_2, \ldots, i_d) - \mathrm{Tr}\left(\prod_{k=1}^{d} \mathbf{Z}_k(i_k) \right) \right\}^2 + \frac{1}{2}\lambda_k \|\mathbf{Z}_k(i_k)\|^2$$

$$\frac{\partial L}{\partial \mathbf{Z}_k(i_k)} = -\left\{ T(i_1, i_2, \ldots, i_d) - \mathrm{Tr}\left(\prod_{k=1}^{d} \mathbf{Z}_k(i_k) \right) \right\} \left(\prod_{j=1, j \neq k}^{d} \mathbf{Z}_j(i_j) \right)^T + \lambda_k \mathbf{Z}_k(i_k)$$

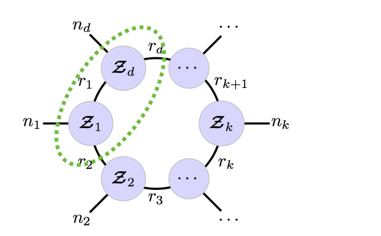
Block-Wise Alternating Least-Squares (ALS)

- ALS is firstly applied to optimize the block of core tensors at each iteration.
- The low-rank matrix decomposition can be employed to separate the block into two core tensors.

$$\min_{\boldsymbol{\mathcal{Z}}_1, \ldots, \boldsymbol{\mathcal{Z}}_d} \|\boldsymbol{\mathcal{T}} - \Re(\boldsymbol{\mathcal{Z}}_1, \ldots, \boldsymbol{\mathcal{Z}}_d)\|_F$$

$$T(i_1, i_2, \ldots, i_d) = \sum_{\alpha_1, \ldots, \alpha_d} Z_1(\alpha_1, i_1, \alpha_2) Z_2(\alpha_2, i_2, \alpha_3) \cdots Z_d(\alpha_d, i_d, \alpha_1)$$

$$= \sum_{\alpha_k, \alpha_{k+1}} \left\{ Z_k(\alpha_k, i_k, \alpha_{k+1}) Z^{\neq k}(\alpha_{k+1}, \overline{i_{k+1} \cdots i_d i_1 \cdots i_{k-1}}, \alpha_k) \right\}$$



Experimental Results

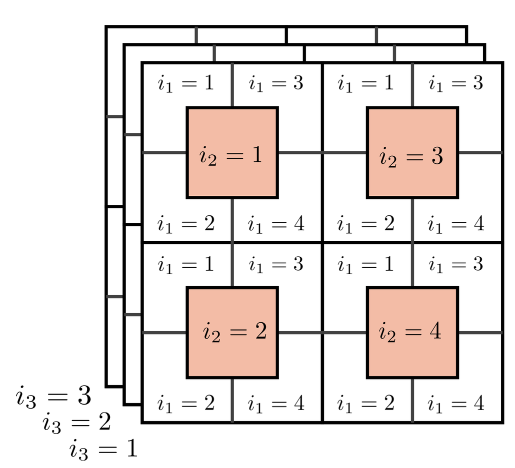
- The number of parameters for tensor representation of an image under varying approximation error ϵ.

Data	$\epsilon = 0.1$		$\epsilon = 0.01$		$\epsilon = 9e-4$		$\epsilon = 2e-15$	
$n = 256, d = 2$	SVD	TT/TR	SVD	TT/TR	SVD	TT/TR	SVD	TT/TR
	9.7e3	9.7e3	7.2e4	7.2e4	1.2e5	1.2e5	1.3e5	1.3e5
Tensorization	$\epsilon = 0.1$		$\epsilon = 0.01$		$\epsilon = 2e-3$		$\epsilon = 1e-14$	
	TT	TR	TT	TR	TT	TR	TT	TR
$n = 16, d = 4$	5.1e3	3.8e3	6.8e4	6.4e4	1.0e5	7.3e4	1.3e5	7.4e4
$n = 4, d = 8$	4.8e3	4.3e3	7.8e4	7.8e4	1.1e5	9.8e4	1.3e5	1.0e5
$n = 2, d = 16$	7.4e3	7.4e3	1.0e5	1.0e5	1.5e5	1.5e5	1.7e5	1.7e5

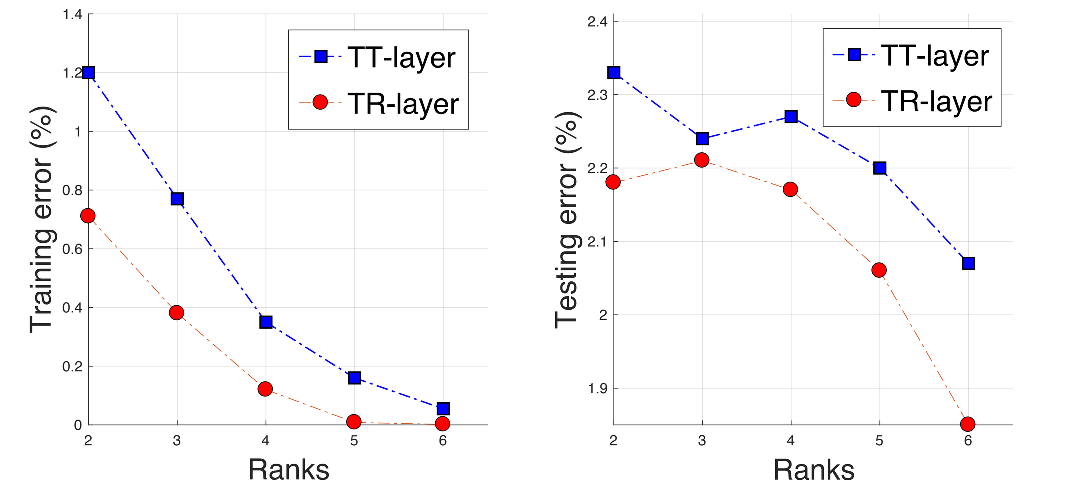
- Based on tensorization operations, TR decomposition is able to capture the intrinsic structure information and provides a more compact representation than TT representation.
- Each core tensor corresponds to a specific scale of resolution.

An individual core tensor is corrupted by random disturbance.

- TR representation can be used for low-rank approximation of model parameters in deep neural networks.
- The model complexity can be compressed by 1300 times.



The tensorization of an image



The classification performances of tensorizing neural networks by using TR representation.

Summary

- A novel tensor decomposition model which can provide an compact representation for a very high-order tensor.
- A scalable SGD algorithm which is useful for large-scale tensors, online learning, and tensor completion.
- TR representation achieves much more compressive deep learning models compared to TT representation.

Our monographs (2017)