Active learning with quantics tensor networks

Krzysztof Bieniasz$^{[0000-0003-2583-4338]}$, Hans-Martin Rieser$^{[0000-0002-1921-1436]}$

DLR Institute for AI Safety and Security, Wilhelm-Runge-Straße 10, 89081 Ulm, Germany
krzysztof.bieniasz@dlr.de
www.dlr.de/en/ki

Keywords: Active learning · Tensor networks · Quantics

1 Introduction

Tensor networks (TNs) are a computational method originating from condensed matter physics, where they enjoy a reputation as the most efficient method for numerically exact simulation of entangled quantum systems in one spatial dimension [7]. For this reason, they are at the heart of most quantum computing simulators, and are considered to be the most straightforward way to translate data and algorithms from classical computers to quantum processors. Recent years have seen a growing interest in these methods in the machine learning (ML) community [8], due to the potential for better explainability owing to the application of physics based tools for analyzing their complexity (such as entanglement entropy, bond dimension, geometry, etc.). On the other hand, owing to their close connection to quantum physics, they are viewed as a natural language for quantum machine learning.

Here, we show some applications of a novel approach to TN-based ML that was developed in recent years by the group of Waintal [1, 5], which leverages the quantics representation of number grids, and the tensor cross-interpolation developed by Oseledets [4], to construct an efficient MPS (Matrix Product State, the elementary one-dimensional TN) representation of arbitrary continuous functions from very few samples, scaling logarithmically with respect to the number of grid points. Importantly, this method employs a DMRG-like algorithm (Density Matrix Renormalization Group, the standard TN algorithm for ground state calculation [6]) without any need for defining energy, which presents unique characteristics from the point of view of ML: It is variational in nature and converges to the represented function exponentially fast and with a well-defined error characteristic; It is not gradient-based and does not suffer from the barren-plateau problem; It is an active learning approach, where the algorithm determines by itself which samples to choose in order to minimize the representation error. This last property has the consequence that the method requires access to a black-box function from which it can learn its characteristics. Thus, it is most useful for creating fast surrogate models of functions which are hard to evaluate exactly numerically, i.e., when a lot of computations are required, or even when it is a result of an experiment.

On the other hand, this representation can be very efficient for certain numerical operations, particularly when they can be represented efficiently as a Matrix Product Operator (MPO). This is the case for differentials, integrals, and even Fourier transforms, whose evaluations on even very large grids are very efficient. Furthermore by constructing representations of parameterized orthogonal functions, one can leverage the physical interpretation of the MPS-MPS contraction of two TN states as a scalar product to calculate decompositions in orthogonal function bases. Below, we show some applications of these methods in practice.

2 Quantics tensor networks

Tensor networks mitigate the "curse of dimensionality" of multidimensional data arrays through factorization into separate low-order tensors. In particular an MPS is a decomposition into tensors of order 3. If a low rank decomposition can be found, such a representation can easily be shown to scale linearly with the number of dimensions of the array, or in other words logarithmically with respect to the number of data points in the array. Consequently, if one calculates the values of a function over a regular grid of 2^N points, they can be indexed with N binary numbers $\sigma_k = \{0, 1\}$, which corresponds to a N-dimensional tensor with each index size 2 (this limitation is not necessary, however it is the most common and practical):

$$f(x) \equiv f_{\sigma_1,\dots,\sigma_N}, \qquad x = (x_1 - x_0)\sum_{k=1}^{N} 2^{-k}\sigma_k + x_0, \qquad x \in [x_0, x_1], \tag{1}$$

which is called a quantics tensor. This tensor can in principle be decomposed into an MPS using SVD or QR factorization. However, this would be highly inefficient, requiring the calculation of 2^N values of the function, storing them, and performing factorizations of huge matrices. In physics problems, this issue is typically sidestepped using the DMRG algorithm: Instead of calculating a quantum state in an exponential Hilbert space and then decomposing it into an MPS, one starts from a translation-invariant Hamiltonian and constructs an MPS representation of its ground state by locally optimizing (pairs of neighboring) individual tensors with respect to the global energy minimum. Thanks to the isometric structure of the canonized MPS form, global errors resulting from local approximations (truncation of small singular values) are well-controlled and easy to estimate. This fact guarantees the numerical stability and fast convergence of the algorithm. The approximation is controlled with a hyperparameter called the bond dimension, the maximal number of singular values retained for any given pair of neighboring tensors [3].

Unfortunately, most ML problems do not allow for any obvious definition of energy, and thus DMRG cannot be applied directly to such problems. This issue was recently overcome by replacing SVD with the tensor cross-interpolation (TCI), which constructs a low-rank representation of a matrix by interpolation from carefully chosen pivot points, or more practically using LU decomposition, which is numerically more stable due to avoiding the explicit construction of the core inverse matrix [1]:

$$A = \begin{pmatrix} A_{11} \\ A_{21} \end{pmatrix} A_{11}^{-1} \begin{pmatrix} A_{11} & A_{12} \end{pmatrix} = \begin{pmatrix} L_{11} \\ L_{21} \end{pmatrix} \begin{pmatrix} U_{11} & U_{12} \end{pmatrix} \tag{2}$$

where $A_{11} = L_{11}U_{11}$ is the square matrix of pivot points—the cross points between the columns A_{*1} on the left and rows A_{1*} on the right chosen for the interpolation, while L_{*1} and U_{1*} are the corresponding blocks of the truncated lower- and upper-triangular matrices from the LU decomposition. This decomposition, while not rank-optimal, can be used to approximate the local tensors by keeping track of the global pivot points used for the local construction, which supersedes the energy formalism. Applying this algorithm to a quantics tensor is dubbed QTCI.

In a similar manner, MPO representations of various operators can be constructed and used for very fast numerical operations using TN algebra. Representations of bond dimension 2 for both integral and differential operator can be constructed explicitly [2], and a Fourier transform MPO with a bond dimension 11 can be constructed numerically with QTCI [1]. A definite integral over the whole range corresponds to a trivial MPO of dimension 1. Other operators could be combined from these, or constructed using QTCI directly.

3 Demonstration of the algorithm

Here we present two simple demonstrations of the QTCI algorithm. Fig. 1 on the left shows the plot of the function $\text{sinc}(x) = \sin(x)/x$, known as sinus cardinalis. The QTCI model was constructed with 8 tensors (2^8 grid points) with bond dimension 4 being sufficient to perfectly recreate the function using just two DMRG sweeps. In orange pluses we plot all the points as they are generated from the model, and in green the points that are explicitly stored in the MPS (the A_{ij} elements of Eq. (2)), the other ones being the result of interpolation. Finally, in purple pluses we plot the result of integration using an MPO representation on the integral operator, overlaid over the plot of $\text{Si}(x)$, which is the integral of sinc(x); a small systematic discrepancy can be seen which is due to the finite size effect. This is a rather simple example to visualize how the model operates—in this case the function is too simple to benefit from the exponential grid density. We can however clearly see that the algorithm finds a nontrivial subset of points to represent the function (this choice is nondeterministic if the function has no uniquely optimal representation) and the resulting interpolation is non-linear, produced not just from the nearest sampled points but from the global properties of the representation.

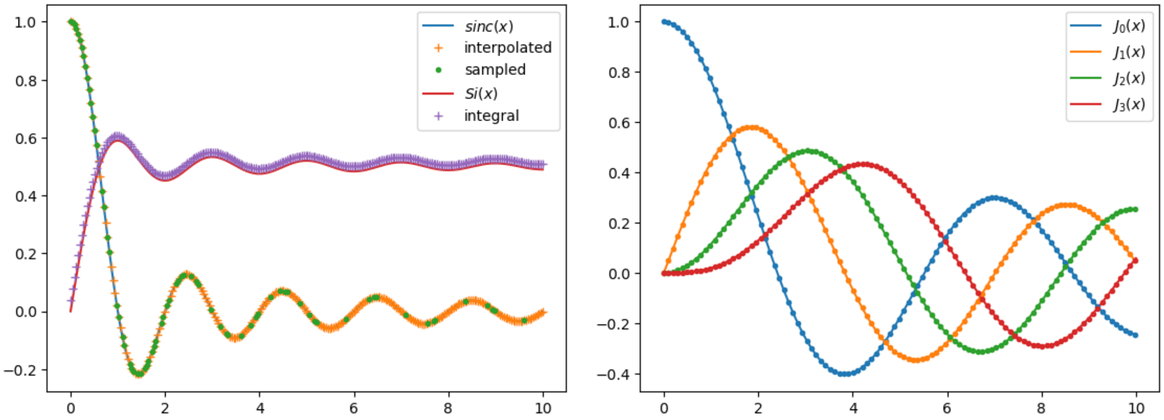

Fig. 1. Left: The function sinc(x), its QTCI representation, and its MPO integral. Right: Bessel functions obtained by contraction of the QTCI representation of Eq. (3) overlaid over actual Bessel functions.

On the right side of Fig. 1 we present the result of calculating the Bessel functions of the first kind $J_n(x)$ for $n = \{0, 1, 2, 3\}$. Bessel functions are a family of special functions, the solutions to a particular differential equation, which are of huge relevance in many areas of physics and engineering. They can also be expressed using the Hansen-Bessel integral formula:

$$J_n(x) = \frac{1}{\pi} \int_0^\pi \cos(nt - x\sin(t))dt. \tag{3}$$

This is a fairly complicated integral to evaluate. We construct a QTCI representation of the integrand function with 10 tensors for $x \in [0, 10]$, 20 tensors for $t \in [0, \pi]$, and 2 for the four values of n, for a total of 2^{32} grid points. We then perform a definite integration over the t argument using a partial MPS contraction. The result recreates the Bessel functions exactly, as demonstrated by the line plots calculated with SciPy. Moreover, the result is obtained simultaneously for all the values of n without any additional processing.

These simple examples are only meant to illustrate the potential of QTCI—it can be used to build efficient models with very few samples, particularly for functions that are costly to calculate. Once a model is constructed, one can perform very fast numerical operations on them using TN contraction algorithms. A number of ML and scientific applications come to mind: Construction of surrogate ML models with much faster inference and memory performance; MPO representations of activation functions to construct full TN representations of neural networks; Very fast integration of functions that are hard to calculate with other methods but easy to represent with QTCI; Classification of data into orthogonal classes using parameterized models; Construction of effective models from dense multidimensional tensors, including from experimental data. In some regimes, construction of interpolation models from incomplete data could also be possible. In such cases one could perform TCI by excluding the missing entries from consideration for pivot points, leading to a natural calculation of interpolation. Further study would be necessary to assess the error estimation of such models. In other cases one could construct local interpolations from near data points using Gaussian interpolation and construct a global QTCI model based on that.

References

1. Fernández, Y.N., Ritter, M.K., Jeannin, M., Li, J.W., Kloss, T., Louvet, T., Terasaki, S., Parcollet, O., von Delft, J., Shinaoka, H., Waintal, X.: Learning tensor networks with tensor cross interpolation: New algorithms and libraries. SciPost Phys. **18**, 104 (2025). https://doi.org/10.21468/SciPostPhys.18.3.104
2. Kazeev, V.A., Khoromskij, B.N.: Low-rank explicit qtt representation of the laplace operator and its inverse. SIAM Journal on Matrix Analysis and Applications **33**(3), 742–758 (2012). https://doi.org/10.1137/100820479
3. Orús, R.: A practical introduction to tensor networks: Matrix product states and projected entangled pair states. Annals of Physics **349**, 117–158 (2014). https://doi.org/https://doi.org/10.1016/j.aop.2014.06.013
4. Oseledets, I., Tyrtyshnikov, E.: TT-cross approximation for multidimensional arrays. Linear Algebra and its Applications **432**(1), 70–88 (2010). https://doi.org/https://doi.org/10.1016/j.laa.2009.07.024
5. Ritter, M.K., Núñez Fernández, Y., Wallerberger, M., von Delft, J., Shinaoka, H., Waintal, X.: Quantics tensor cross interpolation for high-resolution parsimonious representations of multivariate functions. Phys. Rev. Lett. **132**, 056501 (Jan 2024). https://doi.org/10.1103/PhysRevLett.132.056501
6. Schollwöck, U.: The density-matrix renormalization group in the age of matrix product states. Annals of Physics **326**(1), 96–192 (2011). https://doi.org/https://doi.org/10.1016/j.aop.2010.09.012, january 2011 Special Issue
7. Silvi, P., Tschirsich, F., Gerster, M., Jünemann, J., Jaschke, D., Rizzi, M., Montangero, S.: The Tensor Networks Anthology: Simulation techniques for many-body quantum lattice systems. SciPost Phys. Lect. Notes p. 8 (2019). https://doi.org/10.21468/SciPostPhysLectNotes.8, https://scipost.org/10.21468/SciPostPhysLectNotes.8
8. Wang, M., Pan, Y., Xu, Z., Li, G., Yang, X., Mandic, D., Cichocki, A.: Tensor networks meet neural networks: A survey and future perspectives (2025), https://arxiv.org/abs/2302.09019

Acknowledgment

This project was made possible by the DLR Quantum Computing Initiative and the Federal Ministry for Economic Affairs and Climate Action; qci.dlr.de/projects/qutenet