Tensor and Tensor Networks for Machine Learning: An Hourglass Architecture

Xiao-Yang Liu\textsuperscript{1}, Qibin Zhao\textsuperscript{2}, Anwar Walid\textsuperscript{3}
\textsuperscript{1}Electrical Engineering, Columbia University, USA
\textsuperscript{2}Tensor Learning Team, Center for Advanced Intelligence Project, RIKEN, Japan
\textsuperscript{3}Nokia-Bell Labs, NJ, USA
XL2427@columbia.edu, qibin.zhao@riken.jp, anwar.walid@nokia-bell-labs.com

Abstract

Tensor and tensor networks are envisioned to have great potential to advance machine learning technologies. Recent works show that tensor networks provide powerful simulations of quantum machine learning algorithms on classical computers. We observe that tensor and tensor networks in machine learning exhibit a layered architecture that resembles an hourglass. In this paper, we describe a seven-layer architecture to characterize the role of tensor and tensor networks in machine learning, point out current challenges and discuss recent innovations. As a cornerstone data structure, tensor and tensor networks lie at the waist of the hourglass-shaped architecture, while the lower and upper layers tend to see frequent innovations. We expect tensor and tensor networks continue to serve as an amplifier for computational intelligence, a transformer for machine learning innovations, and a propeller for AI industrialization.

1 Introduction

Why do conventional machine learning algorithms use vectors and matrices, while deep learning algorithms and neural networks mostly rely on tensors? A direct answer is that deep learning usually involves hundreds, if not thousands, of features.

Tensor networks, a contracted networks of factor tensors, have arisen independently in several areas of science and engineering. Such networks appear in the description of physical processes and an accompanying collection of numerical techniques have elevated the use of tensor networks into a variational model of machine learning. Tensor networks have shown significant power in compactly representing deep neural networks [Novikov et al., 2015], and efficient training and theoretical understanding of deep neural networks. More potential tensor network technologies are rapidly emerging, such as approximating probability functions and probabilistic graphical models [Stoudenmire and Schwab, 2016; Han et al., 2018]. A merger of tensor network algorithms with state-of-the-art approaches in deep learning is now taking place.

We observe that tensor and tensor networks in machine learning exhibit a layered architecture that resembles an hourglass. Such an observation is analogy to the hourglass structure [Akhshabi and Dovrolis, 2011] of the Internet protocol stack (known as TCP/IP) that successfully provides end-to-end data communication by specifying how data should be packetized, addressed, transmitted, routed, and received.

The three wagons for the success of machine learning are

- **Big data:** the past decade witnesses an exponential explosion of sensory data due to the great advances in sensor manufacturing, leading to the debate *More is more!* or *More is less?* [Baraniuk, 2011].
- **Tensor data structure:** As a cornerstone data structure, tensor and tensor networks are envisioned to have great potentials to promote the development and deployment of machine learning technologies.
- **Intelligent computing for computational intelligence!** Deep learning [LeCun et al., 2015] are computational models with multiple processing layers that learn representations of data with multiple levels of abstraction.

In this paper, we attempt to initiate a layered architecture for tensor and tensor networks, which will benefit the development of machine learning theory, AI chip manufacturing, and AI applications. This seven-layer architecture resembles an hourglass, namely, tensor and tensor networks lie at the waist while the lower and upper layers tend to see frequent innovations. The bottom layer is the hardware, the highest layer is the AI applications and products. Further, we point out current challenges and discuss recent innovations.

Such an hourglass-shaped layer architecture enjoys disciplinary advantages, including layer-wise standardization, intra-layer modularity and inter-layer separability. The *layer-wise standardization* encourages an eco-system for machine learning research and industrialization. With the *intra-layer modularity*, one can update a functional module without interfering other modules. The *inter-layer separability* means that the lower layer is transparent to the upper layer that calls the APIs provided by the lower layer. We expect tensor and tensor networks continue to serve as an *amplifier* for computational intelligence, a *transformer* for machine learning innovations, and a *propeller* for AI industrialization.

We aim to promote discussions (by a series of workshops and academic events) among researchers investigating inno-
vative tensor network technologies from perspectives of fundamental theory and algorithms, novel approaches in machine learning and deep neural networks, and variour applications in computer vision, biomedical image processing, natural language processing, and many other related fields.

The remainder of this paper is organized as follows. Section 2 describes the proposed hourglass architecture. Section 3 discussed key challenges and recent innovations. We conclude this paper in Section 4.

2 The Proposed Hourglass Architecture

We propose a seven-layer architecture for tensor and tensor networks, which resembles an hourglass.

2.1 Layer 1: X Processing Unit

In the post Moor’s law era [Theis and Wong, 2017], the rise of deep learning [LeCun et al., 2015] can be largely credited to a new paradigm Intelligent computing for computational intelligence! The impetus to AI computation is made-for-AI chips-processors, called XPU, including GPUs, FPGAs, and ASICs (NPU).

There is an emergence of dedicated AI accelerator using the ASIC (Application Specific Integrated Circuit) technology, called NPU (neural processing unit). Of particular interest are tensor-based NPU, including Google TPU (tensor processing unit) [Jouppi et al., 2017], tensor cores in NVIDIA Volta/Turing Architecture, Intel Nervana neural network processors (NNP), Tensor Computing Processor BM1684, Alibaba Ali-NPU, Knupath Hermosa, Baidu XPU [Ouyang, 2017], the Huawei Ascend 910 using 32 DaVinci AI cores [Liao et al., 2019b], etc.

2.2 Layer 2: BLAS and Automatic Differentiation

To fully utilize the computing power of hardware XPU in Layer 1, BLAS (Basic Linear Algebra Subprograms, or Basic Tensor Algebra Subroutines B/TAS) and AutoDiff (Automatic differentiation) [Paszke et al., ] are “a knife and fork” for effective implementation of machine learning models.

\begin{itemize}
  \item BLAS level 1 (1969): “vector-vector”;
  \item BLAS level 2 (1972): “matrix-vector”;
  \item BLAS level 3 (1980): “matrix-matrix”;
  \item BLAS level 4 (Now?), “tensor-tensor”: tensor operations include tensor (Kronecker) product, Khatri-Rao product, Hadamard product, tensor contraction, t-product or L-product [Liu and Wang, 2017], etc.
\end{itemize}

Such BLAS standards are implemented and optimized in different programming languages. For example, numpy in Python, cuBLAS and cuTensor in NVIDIA CUDA. Multilinear is general-purpose linear algebra and multi-dimensional array library for Haskell.

Automatic differentiation is a technique to numerically evaluate the derivative of a function, which is believed to be very powerful when combining the back-propagation algorithm. Interested readers may refer to AutoDiff [Paszke et al., ], DDSP (differentiable digital signal processing) [Engel et al., 2020], etc.

2.3 Layer 3: Tensor Data Structure

Tensor is the most popular data structure in machine learning, especially in deep learning. For instance, a) input data: color image set, video sequence, MRI/fMRI, EEG, gene expression, traffic data, social network data, knowledge graph; b) High-order statistical information, high-order moment, covariance, cumulant, etc.; c) model parameters: fully connected layer, convolutional layer, multi-task weight parameters, multi-modal feature fusion, and etc.; and d) function: probability mass function of multiple discrete variables.

In the past, a unified notation set for tensors [Kolda and Bader, 2009] and tensor networks [Cichocki et al., 2016] successfully helps the adoption of tensor tools and the development of tensor network libraries in machine learning.

From a machine learning perspective, an N-th order tensor is a container that can house N-dimensional data and associates with linear/multi-linear operations. A scalar is 0-dimensional, a vector has a single dimension (1D), a matrix has two dimensions (2D), and a higher-order tensor has more than two dimensions.

From a spectral (or transform) perspective, tubal-scalars [Kilmer and Martin, 2011] [Kilmer et al., 2013] [Liu and Wang, 2017] are vectors with the multiplication operation defined according to the convolution theorem. Considering a graph transform, one can have graph-tensors [Malik et al., 2019] or connected matrices [Sun et al., 2018], and graph tensor neural networks [Liu and Zhu, 2020].

2.4 Layer 4: Tensor Decompositions and Tensor Networks

Many practically useful and efficient tensor models are built upon tensor decompositions and tensor networks.

**Tensor Decompositions:** Canonical Polyadic (CP) tensor decomposition, Tucker tensor decomposition, TT [Oseledets, 2011] or TR [Zhao et al., 2016] tensor decomposition, HT, tSVD, reshuffling TD. Sparse tensor decomposition and non-negative tensor decompositions are also developed as extensions of CP, Tucker, TT, and HT.

The uniqueness of CP tensor decompositions [Cichocki et al., 2015] indicates that multilinear algebra may have theoretical advantages over bilinear and linear algebra.

Other important applications includes tensor completion [Song et al., 2019; Liu et al., 2019], tensor time series [Rogers et al., 2013; Lu et al., 2018], spectral learning on matrix/tensor [Janzamin et al., 2019], and data privacy [Kong et al., 2019; Fu et al., 2020; Feng et al., 2020].

**Tensor Networks:** TNs show advantages mostly in space complexity reduction and computation efficiency. Tensor Networks have been employed to a) large-scale optimization problems, large-scale eigenvalue problem, large-scale SVD, large-scale matrix pseudo-inverse; b) model compression in DNN, including fully connected layer and convolutional layer; c) expressive power analysis of DNN.

Many complicated TN models including MERA, PEPS, and etc, which have not applied to machine learning but may have potential advantages in particular problems.
2.5 Layer 5: Tensor Libraries & Programming IDE

Widely used tensor IDEs are TensorFlow [Abadi et al., 2016], PyTorch [Paszke et al., 2019], TensorRT [Vanholder, 2016], Theano, Keras, Apache MXNet, Caffe2, CNTK, PaddlePaddle, MindSpore, MegEngine, etc.

Other libraries include TensorFlow [Dong et al., ], TensorLy [Kossaifi et al., 2019]; TensorNetwork Library [Roberts et al., 2019]; Tensor decomposition in TensorFlow [Novikov et al., 2020], sparse tensor computing [Phipps and Kolda, 2019], and differentiating tensor networks library [Liao et al., 2019a].

For quantum physics, iTensor (Intelligent Tensor) 1 provides a collection of optimized tensor network algorithms.

2.6 Layer 6: Machine Learning Models

There are active research on designing tensor-based machine learning models. We describe a few approaches in the following.

TensorFace [Vasilescu and Terzopoulos, 2002][Vasilescu and Terzopoulos, 2003] presents facial image ensembles, where the relevant factors include different faces, expressions, viewpoints, and illuminations. TensorMask [Chen et al., 2019] is proposed for dense object segmentation.

Tensor regression [Kossaifi et al., 2017] extends the conventional regression models to tensor representation, while tensor mixture model [Sharir et al., 2016] proposed a probabilistic graphic model in tensor form.

AutoEncoder can be extended to tensor form, such as tensor sparse coding [Jiang et al., 2018].

The generative adversarial network framework is extended to tensor GAN [Liu and Wang, 2020] with application to real-time indoor localization for smartphones.

In the model-based direction, tensor neural networks are proposed by unfolding tensor algorithms into deep neural networks, e.g., [Ma et al., 2019][Han et al., 2020] design fast decoders for snapshot compressive imaging cameras, [Liu and Zhu, 2020] considered recovery of nodes’ data matrices, and [Zhang et al., 2020b] investigated the video synthesis problem.

2.7 Layer 7: Applications and Products

Many products embracing AI is enjoying a booming market, penetrating our daily lives: from smartphones to self-driving cars and robotics, search engines, typing assistants (auto-completion), to healthcare services.

Compressing and optimizing neural networks for inference at mobile devices: (i) TVM (tensor virtual machine) [Chen et al., 2018]; (ii) the Tensor Algebra Compiler (taco) is a C++ library that computes tensor algebra expressions on sparse and dense tensors. It uses novel compiler techniques to get performance competitive with hand-optimized kernels in widely used libraries for both sparse tensor algebra and sparse linear algebra.

AutoML and neural architecture search (NAS) are promising, where the training and inference are performed at cloud servers. Many applications are now successfully deployed, including speech recognition, visual object recognition, object detection; others: drug discovery and genomics. Note that health-care is one of the hottest trends, while agriculture applications may have broad social impacts, including automatic quality check, mineral delivery optimization in hydroponics. Disaster recovery is also a critical application.

Big data analysis [Sidiropoulos et al., 2017] for image, video; sensory data processing; EEG brain data; finance, genetics, etc.

AI is now being applied massively in entertainment industry, such as chess and poker, medias (e.g. Netflix), music industry (IBM Watson), and online games, etc.

Other AI products that benefits tensor network algorithms are listed as follows:

- reCAPTCHA is a CAPTCHA-like system designed to establish that a computer user is human.
- SIRI is one of many voice assistants available today.
- Gmail recently introduced autocomplete tools.
- Plagiarism checking by searching for matches in billions of documents.
- FaceID is a feature recently introduced by Apple for authentication on iPhone.
- Recommendation systems in Amazon and Alibaba Taobao that suggests users other products based on their preferences and click history.
- Facebook face detection and tagging is a services of Facebook which automatically detects faces in images and tags people from the user friendship set.

3 Challenges and Innovations

3.1 Challenges

The “4V+P” challenge of big data: IBM data scientists break big data into four dimensions [Data and Hub, 2013]: volume for scale of data, variety for different forms of data, velocity for analysis streaming data, and veracity for uncertainty of data. We would like to advocate the privacy-preserving requirement as a plus aspect of tensor learning algorithms. Furthermore, the data acquisition process is expensive in terms of either time or budget.

The C3-challenges of machine learning algorithms are the intertwined computing, caching and communication:

- Computing: Training a model requires substantial amount of time, which in turn slows down the development. How do we speed up machine learning by 100×? Real-time operations requires fast inference, e.g., cuTensor in NVIDIA CUDA.
- Caching: How to support Billion/Trillion-scale tensor computing? How to compress neural network for mobile platforms?
- Communication: the link capacity of data centers, the communication between cloud and edge servers.

---

1iTensor: https://itensor.org/index.html
Quantitatively characterizing the tradeoff between model compression and performance: how to select tensor network models for different neural networks? How to tune the hyperparameters in the tensor network model?

Trustworthy AI: Explainability, interpretability, and understandability. Interpretableness is about the extent to which a cause and effect can be observed within a system. Explainability (for decision making), meanwhile, is the extent to which the internal mechanics of a machine or deep learning system can be explained in human terms.

Understanding neural-intelligence: a two-layer feedforward network [Janzamin et al., 2015] is analyzed using CP tensor decomposition and such a network is believed to learn a mapping between data distribution priors and labels. On the other hand, an elementary function of neural net’s intelligence is to recognize symmetry structures in the data [Shang and Liu, 2019]: The glove for the left hand is able to fit the right hand if we turn it inside out like placing an imaginary mirror near the opening. Analogously, neural networks play a similar role as a glove when dealing with inputs of symmetry structures. The classic Kruskal uniqueness theorem is exploited to provide a sufficient condition for the situations where such a generalization capability will hold.

Tensor networks provide a rigorous approach to investigate Why deep is good? Nadav [Cohen et al., 2016] considered sum-product networks and CNN with ReLU activation functions [Cohen and Shashua, 2016]. Khrulkov [Khrulkov et al., 2018][Khrulkov et al., 2019] took a similar approach to analyze RNNs.

Robustness of Machine Learning Models: deep adversarial learning; The notion of differential privacy is believed to be very power to construct ensemble methods that fuse sub-networks into a more robust one [Li et al., 2019].

3.2 Innovations

One recent trend regarding both AI software and hardware is to consider inference and training as two separate different phases with different computational approaches. It is becoming standard to develop specific chips for training and specific chips for inference.

Cross-layer Codeign. High performance tensor learning operations by exploiting the massive parallelisms are important for both training and inference: 1). Tensor decompositions on GPUs/FPGA such as cuTensor library [Zhang et al., 2019][Liu et al., 2020][Hong et al., 2020][Huang et al., 2020] and swTensor [Zhong et al., 2019]; 2). Tensor completion [Zhang et al., 2020a].

Federated learning [Kong et al., 2019] or Privacy-preserving tensor algorithms: homomorphic encryption methods for tensor decompositions.

Quantum Machine Learning [Levine et al., 2018]: tensor networks provide powerful simulations of quantum machine learning algorithms on classical computers, which may promise quantum advantages, such as potentially exponential speedups in training, quadratic speedup in convergence, etc.

Tensor network learning vs deep learning: TN has the power to express functions, will tensor network learning be used as a general machine learning model like deep learning?

4 Conclusion

Tensor and tensor networks are envisioned to have great potentials to promote the development and deployment of machine learning technologies. In this paper, we have proposed a seven-layer architecture to characterize the role of tensor and tensor networks in machine learning, point out current challenges and discuss the development trends. Such a layered architecture resembles an hourglass. As a cornerstone, data structure, tensor and tensor networks lie at the waist of the hourglass, while the lower and upper layers tend to see frequent innovations. We expect tensor and tensor networks continue to serve as a transformer for machine learning innovations, an amplifier for computational intelligence, and a propeller for AI industrialization.

The interplay between tensor networks and machine learning algorithms is rich. Indeed, this interplay is based not just on numerical methods but on the equivalence of tensor networks to various arithmetic circuits, rapidly developing algorithms from the mathematics and physics communities for optimizing and transforming tensor networks, and connections to low-rank methods for learning. A merger of tensor network algorithms with state-of-the-art approaches in deep learning is now taking place. A new community is forming, which this workshop aims to foster.

References


