Limitations of gradient-based Born Machines over tensor networks on learning quantum nonlocality

Khadijeh Najafi
Department of Physics
Harvard & Caltech
Cambridge, MA
knajafi@g.harvard.edu

Ahmadreza Azizi
Department of Physics
Virginia Tech
Blacksburg, VA
arazizi@vt.edu

Miles E. Stoudenmire
CCQ, Flatiron Institute
New York, NY
mstoudenmire@flatironinstitute.org

Xun Gao
Department of Physics
Harvard, Cambridge, MA
xungao@g.harvard.edu

Susanne Yelin
Department of Physics
Harvard, Cambridge, MA
syelin@g.harvard.edu

Mikhail D. Lukin
Department of Physics
Harvard, Cambridge, MA
lukin@physics.harvard.edu

Masoud Mohseni
Google AI
Venice, CA
mohseni@google.com

Abstract

Nonlocality is an important constituent of quantum physics which lies at the heart of many striking features of quantum states such as entanglement. An important category of highly entangled quantum states is Greenberger-Horne-Zeilinger (GHZ) states which play key roles in various quantum-based technologies and are particularly of interest in benchmarking noisy quantum hardware. A novel quantum-inspired generative model known as Born Machine which leverages on probabilistic nature of quantum physics has shown great success in learning classical and quantum data over tensor network (TN) architecture. To this end, we investigate the task of training the Born Machine for learning the GHZ state over two different architectures of tensor networks. Our result indicates that gradient-based training schemes over TN Born Machine fail to learn the nonlocal information of the coherent superposition (or parity) of the GHZ state. This leads to an important question of what kind of architecture design, initialization, and optimization schemes would be more suitable to learn the nonlocal information hidden in quantum states and whether we can adapt quantum-inspired training algorithms to learn such quantum states.

1 Introduction

In the realm of quantum mechanics which is intrinsically high dimensional and probabilistic, the interface of quantum mechanics and machine learning has become one of the most active fields [1].

Submitted to 34th Conference on Neural Information Processing Systems (NeurIPS 2020). Do not distribute.

One area of research in quantum computing that has attracted considerable interest is the potential of quantum dynamics to accelerate the performance of certain machine learning algorithms. Historically, this interest has originated from the ability of quantum computers in performing fast linear algebra on states space that grows exponentially with the number of qubits. While these quantum accelerated linear-algebra based techniques for machine learning considered as the first generation of quantum Machine learning (QML) algorithms [2, 3, 4], there are other quantum machine learning tasks that do not have any classical analog.

While reaching fault-tolerant regime require monumental progress in hardware development and error correction schemes, with the noisy intermediate-scale quantum (NISQ) devices performing on the order of hundreds of qubits, recently a family of variational quantum algorithms with application in quantum simulation, optimization, and inference of NISQ devices has become the subject of vast investigation. These techniques provide a path to extract knowledge from observables which are constructed over *parametrized quantum circuits* and thus can iteratively be improved similar to parametrized activation functions of classical neural networks. Variational Quantum Eigensolvers [5], Quantum Approximate Optimization Algorithms (QAOA) [6], Quantum Neural Networks (QNNs) [7], and Quantum Convolutional Neural Networks (QCNN) [8] are among some of the most interesting examples of such novel optimization and inference tools. This set of variational algorithms has led to the second generation of QML algorithms which have a strong focus on delivering applications for NISQ technologies and are based on running and benchmarking algorithms on the actual quantum devices.

With an ongoing efforts in harnessing machine learning tools in improving diverse tasks in the field of quantum computing, it becomes crucial to address the expressive and training power of quantum machine learning algorithms in terms of the choice of models and architectures, loss function, optimization algorithms and initialization schemes. In particular, with the inherent nonlocal nature of quantum states, it is natural to ask the following questions: What architecture design would be suitable to learn specific quantum states? How should one proceed with the choice of a loss function? With reliance of gradient-based method on local properties of loss function [9] what kind of optimization scheme would be suitable to learn the nonlocal quantum information? In general, in absence of any concrete learning architectures, one usually resorts to initial random circuits that lead to tremendous challenges in training QNNs due to vanishing gradients in barren plateau optimization landscape [10]. Here, we present another example of the failure of the gradient-based training schemes in learning nonlocal quantum information over tensor network architecture. More recently, it has been shown that the Adam optimizer have poor performance for tensor networks and Baysian tensor networks [11, 12]. In particular, it has been shown that better convergence can be achieved via tangent-space gradient optimization (TSGO) in comparison to the off-the-shelf Adam.

It is known that nonlocality lies at the heart of quantum physics and manifests itself through some of the most striking features of quantum states such as entanglement. An important category of highly entangled states are generalized Greenberger-Horne-Zeilinger (GHZ) states which are coherent superpositions of two maximally distinct quantum states [13] and has shown to play a key role in establishing the nonlocal notion of quantum physics [14, 15]. These states also serve as universal quantum resources [16] and play an important role in quantum-based technologies ranging from error correction [17], quantum communication [18] to quantum meteorology [19]. Furthermore, due to their sensitivity to coherence, they became standard benchmarking tools in characterization of the NISQ devices [20, 21, 22]. In correspondence with the importance of GHZ states, in this paper, we propose the task of learning the GHZ state via quantum inspired generative model known as Born Machine which leverages on probabilistic nature of quantum mechanics[23]. We show that by training a Born Machine over two different tensor network architecture, namely, the matrix product state (MPS) and tree tensor network (TTN) based on gradient-based training schemes one fails to learn the nonlocal information (or parity) indicating the coherent superposition of the GHZ state.

In section 2, we first introduce quantum inspired generative model known as Born Machine over an important categories of quantum many body states known as tensor networks [24, 25]. In section 3, we demonstrate the preparation of noisy GHZ circuit and we gather training data by performing measurement in informationally complete basis. Subsequently, in section 4, we present our results for loss function and output samples generated from our model indicating a challenging task of learning the coherent superposition of the GHZ state. We conclude the paper by discussions on the possible reasons of failure of gradient-based training and discuss the necessity for novel quantum inspired

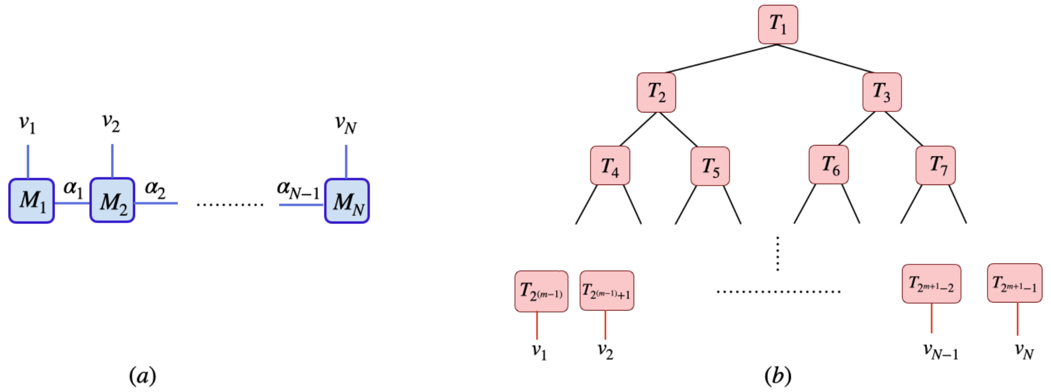

Figure 1: a) Graphical representation of MPS which is a one dimensional array of tensors contracted to each other via virtual bonds. Each tensor has dangling bonds ν indicating the physical degrees of freedoms. b) Graphical representation of TTN with m layers. The tensors in adjacent layers gets contracted to each other and only the last layer contains the physical legs.

training algorithms which would lead to an efficient training over nonlocal quantum or classical data[26].

2 Born Machine as quantum inspired generative model

Generative models which learn the underlying probability distribution of the unlabeled data and generate new samples accordingly has become one of the cornerstones of probabilistic machine learning. Inspired by the probabilistic nature of quantum mechanics, recently, a new paradigm of generative model known as Born Machine has been introduced which uses quantum state representation and learns the joint probabilities over such quantum degrees of freedom accordingly to

$$p_{\boldsymbol{\theta}}(x) = \frac{|\psi_{\boldsymbol{\theta}}(x)|^2}{\mathcal{N}}, \tag{1}$$

where $\mathcal{N} = \sum_{\boldsymbol{\theta}} |\psi_{\boldsymbol{\theta}}(x)|^2$ is the normalization of the wavefunction to ensure the positivity of the probabilities and θ denote the parameter of the model. One promising class of physically-motivated models which has a great potential in efficient representation of a wide class of quantum many body states are tensor networks. The complexity of the tensor network architectures is based on the number of quantum correlations that could be generated in a (local) many-body quantum system with a given topology efficiently, thus capturing the structure of quantum data. The hierarchical structure of tensor networks provides a potential to efficiently encode a complicated quantum state with an appropriate amount of resources which can be controlled, e.g., by the value of a bond dimension, for a given task. In this section, we introduce two known classes of tensor network architectures known as matrix product state (MPS) and tree tensor network (TTN).

Matrix Product state (MPS): The MPS also known as tensor train decomposition is one of the most studied families of tensor network. Owing to a particular variational scheme known as density matrix renormalization group (DMRG) [27, 28], the MPS has shown great potential in representing the ground-state of a wide range of quantum many body states. The parameterization of the wavefunction using the MPS can be represented as

$$\boldsymbol{\psi}^{MPS}_{\alpha,\nu} = \sum_{\{\alpha_i=1\}}^{D} \mathrm{M}^{\alpha_1}_{1,\nu_1}\, \mathrm{M}^{\alpha_1,\alpha_2}_{2,\nu_2} \ldots \mathrm{M}^{\alpha_{\mathrm{N}-1}}_{\mathrm{N},\nu_{\mathrm{N}}}, \tag{2}$$

where $\mathrm{M_i}$ are rank-3 tensors and α indicates virtual bonds that tensors are contacted to each other while the dangling bond indicated by ν are physical legs. The expressive power of tensor network arises from the amount of entanglement they can represent over some local structure which sets the lower bound for the value of a bond dimension.

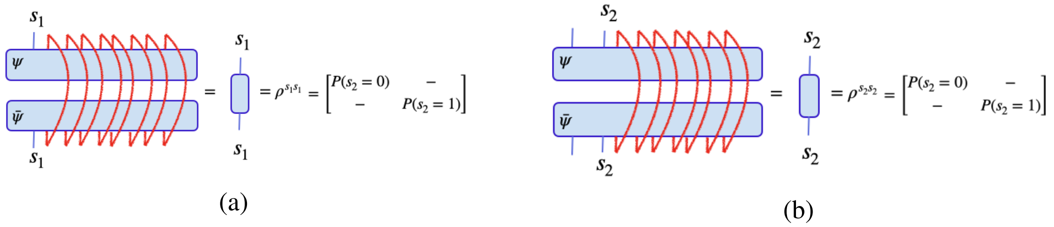
s_1
ψ
$\bar{\psi}$
s_1
$= \rho^{s_1 s_1} = \begin{bmatrix} P(s_2=0) & - \\ - & P(s_2=1) \end{bmatrix}$
(a)
s_2
ψ
$\bar{\psi}$
s_2
$= \rho^{s_2 s_2} = \begin{bmatrix} P(s_2=0) & - \\ - & P(s_2=1) \end{bmatrix}$
(b)

Figure 2: The sampling process in MPS can be performed by building the density matrix (ρ^{s_i,s_i}). a) Starting from the left one obtains the density matrix by performing contraction among the two copies of the MPS while leaving the first tensor to be free. This allows one to assign the value of the first tensor by comparing the outcome of the contraction with a value between 0 and 1 drawn randomly. b) Consequently, by evaluating the marginal probabilities on the second tensor given the assigned value of the first bit string one can assign the second tensor as well. This gets repeated to the next tensor until all the tensors get assigned to a value of 0 or 1.

Tree Tensor Network (TTN): The second family of the tensor network that we use as an ansatz for generative Born Machine are TTN. Because of the architecture of the TTN, they have shown to exhibit better expressibility over data with long-range correlations [25]. The TTN provides a unique decomposition of the quantum state as contractions of N tensors in the following form:

$$\psi^{TTN}_{\alpha,\nu} = \sum_{\{\alpha_i=1\}}^{D} \mathrm{T}^{\alpha_2,\alpha_3}_{[1]} \prod_{n=2}^{N} T^{\alpha_n,\alpha_{2n},\alpha_{2n+1}}_{[n]}, \tag{3}$$

here, the top tensor, T_1 is rank-2 tensor while tensors in middle layers are rank-3 tensors. The TTN has hierarchical structure and only tensors in the last layer have physical legs, see figure 1.

Training of the Born Machine: Training of the Born Machine over tensor networks can be achieved by adjusting the parameters of tensors so that the Born probabilties obtained from the wavefunction after learning can faithfully describe a joint probability distribution of the target data. One of the standard loss function for generative models is known as forward Kullback-Leibler (KL) divergence which is a measure of closeness of the data and model distribution. For dataset $\mathcal{T}$ consists of binary strings $x \in \mathcal{V} = \{0,1\}^{\bigotimes N}$ described by $p_{data}(x)$ we have

$$D_{KL}(p_{data}||p_\theta) = \sum_x p_{data}(x) \ln[\frac{p_{data}(x)}{p_\theta(x)}]. \tag{4}$$

It is straightforward to show that minimizing the forward KL divergence of the data and model distributions is equivalent to minimizing another quantity known as negative log-likelihood (NLL) of the data. Expanding the above equation and rearranging the terms we have

$$\langle \ln p_\theta \rangle_{data} = -S_{data} - D_{KL}(p_{data}||p_\theta), \tag{5}$$

where $S_{data} = -\sum_x p_{data}(x) \ln p_{data}(x)$ is the Shannon entropy. Since the first term does not depend on estimated parameter θ, one can ignore it. Then, according to the law of large numbers the remaining term can be written as

$$\mathcal{L} = -\frac{1}{|\mathcal{T}|} \sum_x \ln p_\theta(x). \tag{6}$$

which is generally known as Negative Log Likelihood (NLL). Consequently, the training can be performed by taking the gradient over the NLL

$$\nabla_\theta \mathcal{L} = -\frac{1}{\mathcal{T}} \sum_x \nabla_\theta \ln |\psi_{\boldsymbol{\theta}}(x)|^2 + \nabla_\theta \ln \mathcal{N}, \tag{7}$$

Notice that $\mathcal{N}$ is the normalization factor which was introduce in equation 1. While the calculation of the first term is straightforward, the second term requires one to perform a summation over all possible configurations. However, in the case of tensor networks presented here, the calculation of normalizing factor and its gradient can be done exactly [24, 25].

Direct sampling: The final step is to generate new samples from the trained TN. Similar to graphical models, a unique feature of tensor network ansatz is their ability in direct sampling. For simplicity, here, we describe the case for MPS and the case for TTN can be performed similarly [29]. Note that after training, ψ^{MPS} is not normalized to 1. Therefore, for the sampling process, we first normalize ψ^{MPS} by dividing it over $\mathcal{N}$. Then, sampling new configuration $s = \{s_1, s_2, ..., s_N\}$ can be started from one end of the normalized ψ, let say s_1. To do so, we first build the density matrix of ρ^{s_1,s_1} as follows: taking a copy of the wave function ψ^{MPS} one contracts all the physical legs of the two copies except s_1. The resulting tensor will be a matrix with two dangling physical legs (see Figure 2). Since the wave function was already normalized, the diagonal elements of the resulting matrix will be the probabilities of s_1 being sampled in either of the possible states.

In other words, $\rho^{s_1=0,s_1=0} = P(s_1 = 0)$ and $\rho^{s_1=1,s_1=1} = P(s_1 = 1)$ represent the probabilities of the first tensor as '0' and '1' respectively. Next, these probabilities are compared with a drawn random number in the range of [0,1) and the value of the first qubit is determined accordingly. The second tensor is drawn conditioned on the state of the first tensor. Therefore, the density function of the second tesnor conditioned on s_1, $(\rho^{s_2,s_2}|s_1)$ is computed similar to Figure 2, but this time the value of s_1 is given and contraction is carried out over remaining physical legs $(s_3, s_4, ..., s_N)$. Again, the density matrix elements $(\rho^{s_2=0,s_2=0}|s_1) = P(s_2 = 0|s_1)$ and $(\rho^{s_2=1,s_2=1}|s_1) = P(s_2 = 1|s_1)$ are the probabilities of the second tensor being drawn as '0' and '1', respectively. This process will be repeated on the next tensors , one-by-one, until the values of all tensors in the s configurations gets generated.

3 Learning local and nonlocal information via tensor network Born Machine

As we mentioned earlier, the GHZ states are important category of quantum states that play a significant role in quantum information science and due to their sensitivity to small amount of noise are particularly suitable for benchmarking the NISQ devices [20, 21, 22]. The GHZ state is defined as superposition of two macroscopically distinct states

$$|\psi_{GHZ}\rangle = \frac{1}{\sqrt{2}}(|11...1\rangle + |00...0\rangle), \tag{8}$$

It is straightforward to show that the GHZ state can be prepared in quantum circuit via a single Hadamard gate on first qubit followed by a series of CNOT gates on neighbouring qubits [21, 22]. In addition, we introduce the noise via depolarizing channel in quantum circuit. The characterization of quantum hardware can be obtained via the fidelity of the GHZ state

$$\mathcal{F} = \langle\psi_{GHZ}|\rho|\psi_{GHZ}\rangle \tag{9}$$

where $(\mathcal{F} > 0.5)$ is shown to be a multipartite entangled witness [30]. The fidelity of the GHZ state can be calculated by only knowing two diagonal elements and corresponding off-diagonal overlap. The information about diagonal elements can be extracted via measurement in Z basis, however, in order to extract information about the coherence of the GHZ state, one is required to perform measurement in another orthogonal basis as well. In the experiment the standard method to evaluate the fidelity is done via parity oscillation to extract the nonlocal information regarding the coherence of the GHZ states [20, 21, 22].

The results we report here are based on simulation of the noisy GHZ circuit on CIRQ. We have run our simulation for various numbers of qubits $L = 4, 8, 12$ and depolarizing noise strength of $p = 0.1$. However, for the sake of clarity, we only present the results for $L = 8$ as it allows one to compare the generated data with the training data. Here, we list the main steps of the simulation followed by the training and sampling: i) First, we perform the measurement on all qubits both on Z and X basis which leads to two vectors of bit strings with length of L. ii) Then, we feed the data into our Born Machine by contracting the input vectors with physical legs of MPS and TTN tensors. iii) Consequently, we perform the training procedure by optimizing the tensor parameters. iv) Once the training is done, we generate new samples according to direct sampling procedure. In the optimization process, we carried out Stochastic Gradient Descent (SGD) and its variations, AdaGrad and Adam with different learning rate values. Results indicate that in almost all cases the loss functions have approached similar final values but that is always faster when ADAM is used. Here, we only show our results for the Adam optimizer with a decaying learning rate from 0.01 which delivers the best

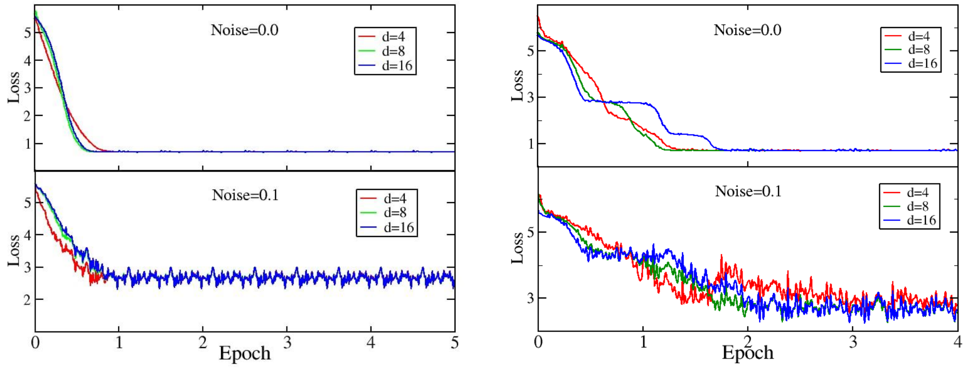

Figure 3: Loss value (NLL) as a function of number of epochs during the training for MPS depicted in left and TTN in the right panel for size of $L = 8$ and various bond dimensions $d = 4, 8, 16$ depicted with red, green and blue respectively. For all of the bond dimensions and both cases of noiseless (top) and noisy circuit (bottom), the loss function approaches a value that is very close to Shannon entropy indicating the power of MPS and TTN in memorizing the training data.

performance. We further examined different initialization for the starting MPS such as starting from some fixed constant values or random values, and for the range of different initialization examined here, we observe the same results. However, one can perform alternative procedures to come up with a good initial guess. In the context of vanishing gradients in barren plateau of energy landscape of quantum circuit learning, alternative non-random initialized has shown successful results [31, 32]. Consequently, similar techniques can be explored in the context of Born Machines on tensor network.

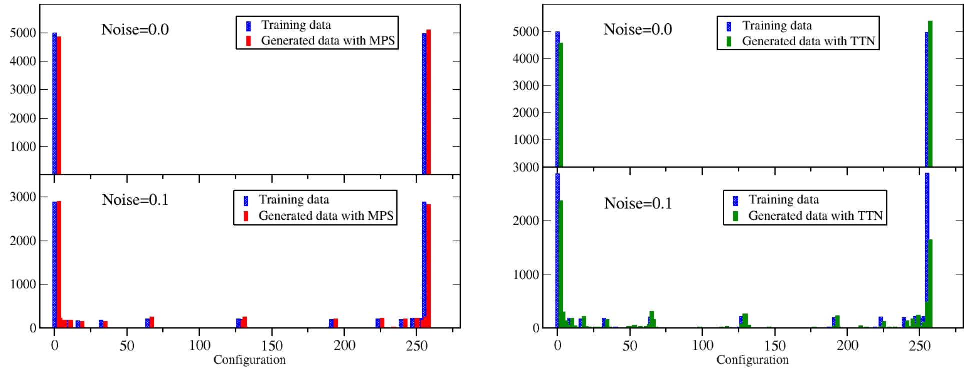

Figure 4: Comparing the outcome of measurement of noisy GHZ circuit in Z basis with generated data for MPS in left and TTN in the right panel. In both cases of noiseless (top) and noisy circuit (bottom), the Born Machine based on MPS and TTN was able to generate new samples very similar to the training data indicating a good generalization of the corresponding Born machine.

First, we study the trainability of the Born Machine over the TN for the measurement outcome of the GHZ state in the Z basis. As shown in 4, for the case with zero noise, there is only two possible outcome $|00..0\rangle$ and $|11...1\rangle$ state. However, as one increases the depolarizing noise strength in the circuit, due to the noise other unwanted states also appear in the measurement outcome. In figure 3, the loss function has been depicted as a function of epoch for MPS and TTN architecture (4 layers) both with bond dimension of 4. While loss function has a slower decay in the case of TTN as there are more parameters to be optimized, in both cases, the negative log likelihood converges to a value which is the Shannon entropy of the training data, indicating that the learning is happening. We further look at new samples generated after the training procedure. The results are shown in 4 Remarkably, for both cases, we observe that the Born Machine based on MPS and TTN is able to generate samples that are very similar to the trained data even for noisy case which is a mixed state, although the poor generalization of TTN in the noisy case might be due to over-fitting as there are more parameter to be optimized in the TTN architecture compare to MPS.

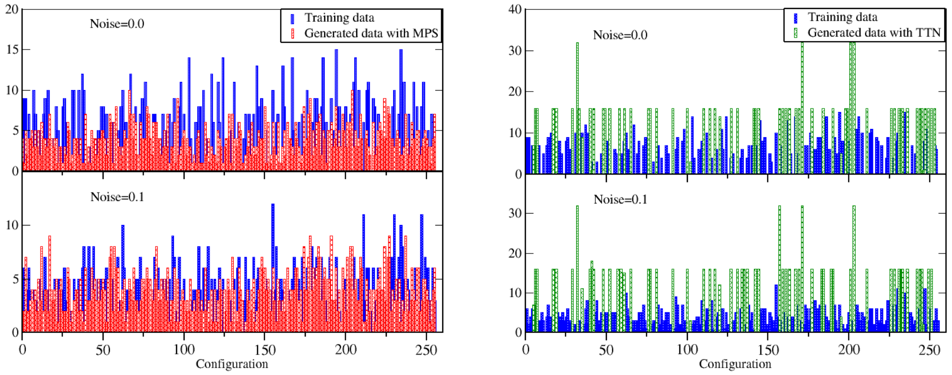

Figure 5: Comparing the outcome of measurement of noisy GHZ circuit in X basis with generated data for MPS in left and TTN in the right panel. In both cases, while loss functions decays as a function of epochs (not shown here) the gradient descent gets stuck in some local minima. This results in poor MPS (TTN) ansatz leading to a poor generalization indicated by generated samples being far from the training data.

As we discussed earlier, in order to faithfully learn the coherence of the GHZ state, one requires to extract the nonlocal information by performing measurement in an orthogonal basis as well. In figure 5, we depicted the sampling results of MPS and TTN Born Machine in learning the outcome of measurement in X basis which resemble the data also obtained via parity measurement in experiments [20, 21, 22]. We observe that, for both cases, the training based on gradient descent schemes fails to learn the nonlocal data (or parity data) and leads to a poor estimation of the fidelity.

In order to quantify the properties of the trained and target states, one could possibly think of measures such as KL divergence defined in equation 4 and its generalized form known as Rényi divergence, i.e., $D_\alpha(p_{data}||p_\theta) = \frac{1}{\alpha-1}\log(\sum \frac{p_{data}^{\alpha}}{p_{\theta}^{\alpha-1}})$ with $0 < \alpha < \infty$ and $\alpha \neq 1$ [33, 34, 35]. However, both of these measures are valid under the condition of absolute continuous (e.g $p_{data} = 0$ implies $P_\theta = 0$) which is not satisfied for the training and generated data from our Born Machine. Instead, we choose the classical fidelity known as the Bhattacharyya coefficient which is the inner product of squares of the two probability distributions

$$F(p_{data}, p_\theta) = [\sum_i \sqrt{p_{data}(i)p_\theta(i)}]^2. \tag{10}$$

For the case of $p_{Data} = p_\theta$, the fidelity becomes 1 and in general, $0 \leq F(p_{data}, p_\theta) \leq 1$. Also notice that the classical fidelity is used as distinguishability of two classical distributions and is different from quantum fidelity which is defined as distinguishability of two quantum states, see equation 9.

Here, we report the values of the fidelity obtained for the MPS with various bond dimensions of $d = 2, 4$ and 8 for cases of noiseless and noisy $= 0.1$ which is repeated for both measurement outcomes in the Z and X basis. First, we observe that for the measurements in Z basis and noiseless case, by increasing the bond dimension to 4, the fidelity reaches the value of 0.99 indicating the high quality of the generated data by the Born-MPS Machine as shown in Figure 4. While similar increases in fidelity can be observed for the noise= 0.1, however, the value is about $\simeq 0.92$. Finally, the situation for the classical fidelity is different for the case of X basis, where the fidelity is way smaller being around 0.45 and 0.82 for the noiseless and noisy case, respectively. While we observe that increasing in bond dimension does not improve the fidelity for both noiseless and noisy case, indicating a sign of poor optimization and training performance, surprisingly the noisy case indicates higher fidelity compared to the noiseless cases which might seem counter intuitive. However, this can be understood as the noise increases, the data becomes more mixed including the data with both odd and even parity which is in contradiction with the noiseless case where all bitstring has even parity. Consequently, in the noiseless case, even a single bit flip would lead to a change in parity lowering the fidelity, while the case of the noisy data is less sensitive to a single bit-flip error.

	noise = 0.0			noise = 0.1		
	d = 2	d = 4	d = 8	d = 2	d = 4	d = 8
$Z\ basis$	0.6579827	0.99599605	0.99699379	0.6015738	0.9300246	0.92310117
$X\ basis$	0.4723854	0.45538960	0.46662666	0.8230743	0.8206209	0.82242423

4 Discussion

Here, we discuss possible reasons for the failure of the gradient-based training schemes in learning the nonlocal or parity data. One reason is that at any given optimization step, one is freezing many other MPS tensors and the training data gets projected through these frozen MPS. If the frozen MPS turns out to be far away from the ideal solution, then, the MPS projection significantly distorts the true nature of the training objectives and the local gradient update sees the wrong landscape. Consequently, this leads to a poor MPS and by looping through these projected MPS's tensors one gets stuck in the wrong minima. While the Z measurement outcome does not seem to be sensitive to the MPS initialization, the parity set is more likely to get stuck in local minima obtained via the poor MPS ansatz. This can be understood since a single bit flip error can change the parity set significantly. Thus, when one performs gradient descent, locally the objective function is only correct if the frozen MPS tensors preserves nonlocal information about "all" of the bit strings connected to the frozen MPS tensors no matter how far the bit strings are from each other. Consequently, it's very unlikely that a generated MPS would be able to preserve the nonlocal information unless it is specially constrained or initialized by a more expressive ansatz which is an interesting subject for future investigation.

Finally, this motivates the idea of developing gradient free algorithms that are more adaptable with tensor networks. Indeed, in a recent paper, it has been shown that the even-parity dataset has been learned via deterministic learning method which leverages on entanglement or quantum correlations of the quantum state. More specifically, the learning happens through a deterministic approach based on reduced density matrix which encodes information of the global states into tensor networks and allows a generalization of the model through a mechanism similar to Density Matrix Renormalization Group [26]. This opens a new direction of adapting quantum inspired gradient free training schemes in learning the GHZ state and other exotic quantum state which we leave for a future study.

Acknowledgments and Disclosure of Funding

Khadijeh Najafi would like to acknowledge the Google Quantum Research Awards 2019 for financial support of the joint postdoc position in Caltech and Harvard.

References

[1] M. Broughton, G. Verdon, T. McCourt, A. J. Martinez, J. H. Yoo, S. V. Isakov, P. Massey, M. Y. Niu, R. Halavati, E. Peters, M. Leib, A. Skolik, M. Streif, D. V. Dollen, J. R. McClean, S. Boixo, D. Bacon, A. K. Ho, H. Neven, and M. Mohseni, "Tensorflow quantum: A software framework for quantum machine learning," 2020.

[2] S. Lloyd, M. Mohseni, and P. Rebentrost, "Quantum principal component analysis," *Nature Physics*, vol. 10, p. 631–633, Jul 2014.

[3] S. Lloyd, M. Mohseni, and P. Rebentrost, "Quantum algorithms for supervised and unsupervised machine learning," 2013.

[4] P. Rebentrost, M. Mohseni, and S. Lloyd, "Quantum support vector machine for big data classification," *Phys. Rev. Lett.*, vol. 113, p. 130503, Sep 2014.

[5] J. R. McClean, J. Romero, R. Babbush, and A. Aspuru-Guzik, "The theory of variational hybrid quantum-classical algorithms," *New Journal of Physics*, vol. 18, p. 023023, Feb 2016.

[6] E. Farhi, J. Goldstone, and S. Gutmann, "A quantum approximate optimization algorithm," 2014.

[7] E. Farhi and H. Neven, "Classification with quantum neural networks on near term processors," 2018.

[8] I. Cong, S. Choi, and M. D. Lukin, "Quantum convolutional neural networks," *Nature Physics*, vol. 15, pp. 1273–1278, Dec 2019.

[9] S. Shalev-Shwartz, O. Shamir, and S. Shammah, "Failures of gradient-based deep learning," 2017.

[10] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, and H. Neven, "Barren plateaus in quantum neural network training landscapes," *Nature Communications*, vol. 9, Nov 2018.

[11] Z.-Z. Sun, S.-J. Ran, and G. Su, "Tangent-space gradient optimization of tensor network for machine learning," *Physical Review E*, vol. 102, Jul 2020.

[12] S.-J. Ran, "Bayesian tensor network with polynomial complexity for probabilistic machine learning," 2020.

[13] M. . Kafatos, *Bell's theorem, quantum theory and conceptions of the universe*. Netherlands: Kluwer, 1989.

[14] A. Einstein, B. Podolsky, and N. Rosen, "Can quantum-mechanical description of physical reality be considered complete?," *Phys. Rev.*, vol. 47, pp. 777–780, May 1935.

[15] L. Hardy, "Nonlocality for two particles without inequalities for almost all entangled states," *Phys. Rev. Lett.*, vol. 71, pp. 1665–1668, Sep 1993.

[16] D. Gottesman and I. L. Chuang, "Demonstrating the viability of universal quantum computation using teleportation and single-qubit operations," *Nature*, vol. 402, pp. 390–393, Nov 1999.

[17] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition*. USA: Cambridge University Press, 10th ed., 2011.

[18] Z. Zhao, Y.-A. Chen, A.-N. Zhang, T. Yang, H. J. Briegel, and J.-W. Pan, "Experimental demonstration of five-photon entanglement and open-destination teleportation," Jul 2004.

[19] L. Pezzè, A. Smerzi, M. K. Oberthaler, R. Schmied, and P. Treutlein, "Quantum metrology with nonclassical states of atomic ensembles," *Rev. Mod. Phys.*, vol. 90, p. 035005, Sep 2018.

[20] A. Omran, H. Levine, A. Keesling, G. Semeghini, T. T. Wang, S. Ebadi, H. Bernien, A. S. Zibrov, H. Pichler, S. Choi, J. Cui, M. Rossignolo, P. Rembold, S. Montangero, T. Calarco, M. Endres, M. Greiner, V. Vuletić, and M. D. Lukin, "Generation and manipulation of schrödinger cat states in rydberg atom arrays," *Science*, vol. 365, no. 6453, pp. 570–574, 2019.

[21] C. Song, K. Xu, H. Li, Y.-R. Zhang, X. Zhang, W. Liu, Q. Guo, Z. Wang, W. Ren, J. Hao, and et al., "Generation of multicomponent atomic schrödinger cat states of up to 20 qubits," *Science*, vol. 365, p. 574–577, Aug 2019.

[22] K. X. Wei, I. Lauer, S. Srinivasan, N. Sundaresan, D. T. McClure, D. Toyli, D. C. McKay, J. M. Gambetta, and S. Sheldon, "Verifying multipartite entangled greenberger-horne-zeilinger states via multiple quantum coherences," *Phys. Rev. A*, vol. 101, p. 032343, Mar 2020.

[23] J. Chen, S. Cheng, H. Xie, L. Wang, and T. Xiang, "Equivalence of restricted boltzmann machines and tensor network states," *Phys. Rev. B*, vol. 97, p. 085104, Feb 2018.

[24] Z.-Y. Han, J. Wang, H. Fan, L. Wang, and P. Zhang, "Unsupervised generative modeling using matrix product states," *Phys. Rev. X*, vol. 8, p. 031012, Jul 2018.

[25] S. Cheng, L. Wang, T. Xiang, and P. Zhang, "Tree tensor networks for generative modeling," *Phys. Rev. B*, vol. 99, p. 155131, Apr 2019.

[26] T.-D. Bradley, E. M. Stoudenmire, and J. Terilla, "Modeling sequences with quantum states: a look under the hood," *Machine Learning: Science and Technology*, vol. 1, p. 035008, Jul 2020.

[27] S. R. White, "Density matrix formulation for quantum renormalization groups," *Phys. Rev. Lett.*, vol. 69, pp. 2863–2866, Nov 1992.

[28] U. Schollwöck, "The density-matrix renormalization group in the age of matrix product states," *Annals of Physics*, vol. 326, p. 96–192, Jan 2011.

[29] A. J. Ferris and G. Vidal, "Perfect sampling with unitary tensor networks," *Phys. Rev. B*, vol. 85, p. 165146, Apr 2012.

[30] C. A. Sackett, D. Kielpinski, B. E. King, C. Langer, V. Meyer, C. J. Myatt, M. Rowe, Q. A. Turchette, W. M. Itano, D. J. Wineland, and C. Monroe, "Experimental entanglement of four particles," *Nature*, vol. 404, pp. 256–259, Mar 2000.

[31] A. Skolik, J. R. McClean, M. Mohseni, P. van der Smagt, and M. Leib, "Layerwise learning for quantum neural networks," 2020.

[32] E. Grant, L. Wossnig, M. Ostaszewski, and M. Benedetti, “An initialization strategy for addressing barren plateaus in parametrized quantum circuits,” *Quantum*, vol. 3, p. 214, Dec 2019.

[33] A. Rényi, “On measures of entropy and information,” *In: Proceedings of the 4th Berkeley Symposium on Mathematics, Statistics and Probability on Mathematics, Statistics and Probability*, vol. 1, pp. 547–561, 1960.

[34] P. Harremoës, “Interpretations of rényi entropies and divergences,” *Physica A: Statistical Mechanics and its Applications*, vol. 365, no. 1, pp. 57–62, 2006.

[35] T. van Erven and P. Harremoes, “Rényi divergence and kullback-leibler divergence,” *IEEE Transactions on Information Theory*, vol. 60, p. 3797–3820, Jul 2014.