Learning Phase Transition in Ising Model with Tensor-Network Born Machines

Abstract

Learning underlying patterns in unlabeled data with generative models is a challenging task. Inspired by the probabilistic nature of quantum physics, recently, a new generative model known as Born Machine has emerged that represents a joint probability distribution of a given data based on Born probabilities of quantum state. Leveraging on the expressibilty and training power of the tensor networks (TN), we study the capability of Born Machine in learning the patterns in the two-dimensional classical Ising configurations that are obtained from Markov Chain Monte Carlo simulations. The structure of our model is based on Projected Entangled Pair State (PEPS) as a two dimensional tensor network. Our results indicate that the Born Machine based on PEPS is capable of learning both magnetization and energy quantities through the phase transitions from disordered into ordered phase in 2D classical Ising model. We also compare learnability of PEPS with another popular tensor network structure Matrix Product State (MPS) and indicate that PEPS model on the 2D Ising configurations significantly outperforms the MPS model. Furthermore, we discuss that PEPS results slightly deviate from the Monte Carlo simulations in the vicinity of the critical point which is due to the emergence of long-range ordering.

1 Introduction

In the past years generative models have shown a remarkable progress in learning the probability distribution of the input data and generating new samples accordingly. The generative models can leverage labels in the data or they can be trained in an unsupervised fashion. Generative adversarial networks (GANs) [1] are the most well studied models that use the information of labels in the data, whereas Variational Auto Encoders (VAEs) [2], for instance, do not necessarily need the labels. Another kind of generative models is the family of Restricted Boltzmann machines (RBMs) [3] that are energy based. These physics-inspired generative models use a joint probability distribution ansatz based on the Boltzmann energy distribution defined as

$$p(\boldsymbol{\theta}) = \frac{e^{-E(\boldsymbol{\theta})}}{\mathcal{N}},$$

where $\mathcal{N} = \sum_{\theta} e^{-E(\boldsymbol{\theta})}$ is the partition function (or the normalization factor of the probability density), and E is the energy associated to the given input.

The success of the above generative models have been vastly examined on large datasets in both image domain and text domain. However, learning and generating physical properties in natural or engineered systems can be completely different from non-physical datasets as the physical configurations should obey key rules and certain inherent properties should be preserved in the generated data, such as conservation laws, invariances , or symmetries in the underlying Hamiltonians. In fact, it has been observed that machine learning generated configurations do not completely match with those obtained with physical simulations [4, 5, 6, 7, 8]. In the case of RBMs, even though RBMs follow a simple, yet powerful ansatz for learning the input data distribution, but the partition function $\mathcal{N}$ in RBMs is intractable and should be calculated through Markov Chain Monte Carlo (MCMC)

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

sampling. It has been shown that this sampling process becomes costly so that it affects the quality of learning of physical properties [8]. Also, it has been reported that RBMs do not fully learn the long range dependencies between the elements in the data [5, 9]. Here, we are seeking a generative model that can be trained over the two dimensional Ising configurations and could efficiently learn both local and non-local physical observables through a phase transition.

Inspired by probabilistic interpretation of quantum mechanics, recently a new paradigm of generative model known as Born Machine has been proposed that can be trained over unlabeled datasets [10, 11, 12]. Formally, Born Machine associate a wave function ψ to each of the training inputs and implies the square of ψ as the likelihood of the input. Mathematically, the likelihood function p can be written as:

$$p(\boldsymbol{\nu}) = \frac{|\psi(\boldsymbol{\nu})|^2}{Z}. \tag{1}$$

where ν is the input data and $Z = \sum_{\boldsymbol{\nu}} |\psi(\boldsymbol{\nu})|^2$ implies the normalization of the probability distribution.

Within the paradigm of the Born Machine, the state ψ can be represented by existing mathematical and computational tools in quantum many body systems.

Tensor networks are known as one of the promising class of quantum models that have shown a great potent in efficient representation of a wide class of quantum many body states. The compact representation of the quantum many body states can be achieved by decomposing the corresponding tensor network with an exponential number of degrees of freedom into a low-rank tensor. One of the most studied examples are matrix product state (MPS) that have efficient representation for one-dimensional gapped local Hamiltonian. With their intrinsic properties, MPS has been used in machine learning tasks such as pattern discrimination[13] and classification[14].

A natural extension of MPS into two dimension is projected pair entangled state known as PEPS. The terms arise from the fact that PEPS can be considered as maximally entangled paired state of some auxiliary systems which is locally projected into another subspace [15]. While PEPS are known to efficiently represent the ground state of two-dimensional system with area-law entanglement, however, unlike MPS, the exact classical contraction algorithms of finite size PEPS grows exponentially with the systems size; however, efficient approximate tensor network contraction schemes have been recently developed [16, 17, 18]. Inspired with this correspondence, and leveraging on the fact that the tensor network have tractable likelihood, in this paper, we interrogate the learning and generating power of the Born Machine based on projected entangled pair states (PEPS model).

In this work, we study the performance of PEPS model in learning the phase transition in the classical Ising configurations that are obtained by the Monte Carlo (MCMC) simulation. Similar to previous works [5, 6, 19], we first train our model on a set of configurations at various temperatures and produce new configurations. Next, we compare the magnetization and energy profiles on both MCMC and PEPS configurations. Also, we compare these results with those that we obtain from learning MPS and TTN over the same Ising configurations. We show that our PEPS model has an overall better performance in generating configurations with correct magnetization and energy values through the wide range of temperatures taken both at ordered and disordered phase of the Ising model.

The rest of the paper is organized as follows. In Section 2, we introduce the Born Machine over MPS and PEPS and we explain the training and sampling processes. In section 3, we briefly describe the Monte Carlo simulation of the Ising model as well as its physical properties. Next, we present physical measurements on the generated Born Machine based on MPS and PEPS configurations and compare them with their Monte Carlo versions. Finally, in Section 4 we discuss the advantages of PEPS model as well as possible new direction for enhanced learning close to critical point.

2 TN Born Machines

Here, we introduce a family of Born Machines based on two known tensor network architectures in one and two dimension, namely the MPS and PEPS. The first family is MPS which is defined as one dimensional array of tensors indicated by T_i that are contracted to each other via virtual bonds as shown in figure 1. Each tensors, also has dangling bonds known as physical legs v_i. The natural extension of MPS into two dimensions are PEPS which are defined as a two dimensional grid of tensors that are contracted to each other in a specific form as depicted 1. Similarly, each of the

tensors T_i with $i \in \{1, 2, ..., N\}$ in the PEPS structure has a physical leg (v_i) that is designed to be connected to the ith element of the input configuration. In many physical systems, including our Ising model, the input configurations have a binary values and hence it requires the dimension of physical legs to be set as 2. For simplicity, in our model, we consider a fixed bond dimension value b for all connections between the tensors.

Training the Born Machines: Given the Equation (1), the variables in those tensors are optimized by minimizing the negative log-likelihood (NLL) described by

$$\mathcal{L} = -\frac{1}{\mathcal{T}} \sum_{\nu \in \mathcal{T}} \ln p(\nu). \tag{2}$$

In the above equation, $\mathcal{T}$ is the number of input configurations. It's known that minimizing the objective function $\mathcal{L}$ is equivalent to minimizing the Kullback-Leibler (KL) divergence that provides a measure of similarity between the empirical and the parameterized probability distributions. Therefore, the gradient of the NLL is given by

$$-\nabla_\nu \mathcal{L} = -\frac{1}{\mathcal{T}} \sum_{\nu \in \mathcal{T}} \nabla_\nu \ln p(\nu) + \nabla_\nu \ln Z, \tag{3}$$

and is calculated to find the best values of the model variables. In the above formula, although the calculation of the first term can be simply carried out, evaluating the second term requires a summation over all possible configurations which is not normally easy. With tensor networks, however, this can be done tractably. To calculate the normalization function Z, in our case, another copy of PEPS is made and its physical legs are contracted through the physical legs of the original PEPS. Intuitively, this process is equivalent to measuring the expectation value of the Identity matrix where all possible values of inputs are considered.

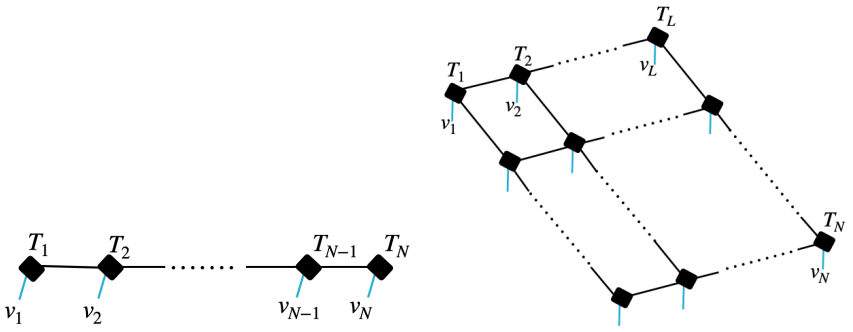


Figure 1: Graphical representation of MPS in left and PEPS in the right panel. The small black boxes represent the tensors Ts which are connected via virtual bonds. Each tensor has dangling bonds ν_i indicating the physical degrees of freedoms.

In addition, the exact computation of the tensor network leads to an efficient sampling which allows the generation of new data according to the learned joint probability distributing. Unlike other energy-based models such as RBM, where the normalizing factor is intractable and requires one to perform the MCMC from an initial configuration which makes it very expensive for computations [5, 6, 19]. Below, we describe the sampling process for PEPS.

Direct sampling from TN architectures: One of the advantage of Born machines over RBMs is the efficiency in generating new configurations. While in RBMs producing new samples is done through Monte Carlo Markov Chain (MCMC) (that significantly increase the model run time), in Born machines the samples are directly calculated via $p(v)$ [11, 12]. Here, we explain the samling for PPES, while similar procedure can be done for MPS and TTN [20]. Given the trained ψ, normalizing ψ with Z, i.e, $\psi \to \psi^N = \psi/Z$, makes the necessary computations in the sampling faster.

Now consider that a new configuration $s = \{s_1, s_2, ..., s_N\}$ is sampled from $p(v)$. Here, we explain the details of this process. From the first spin s_1 and according to the Equation (1), one samples s_1 with probability:

$$p(s_1 = 1) = \sum_{s_2, s_3, ..., s_N} \bar{\psi}^N_{1, s_2, ..., s_N} \psi^N_{1, s_2, ..., s_N} \tag{4a}$$

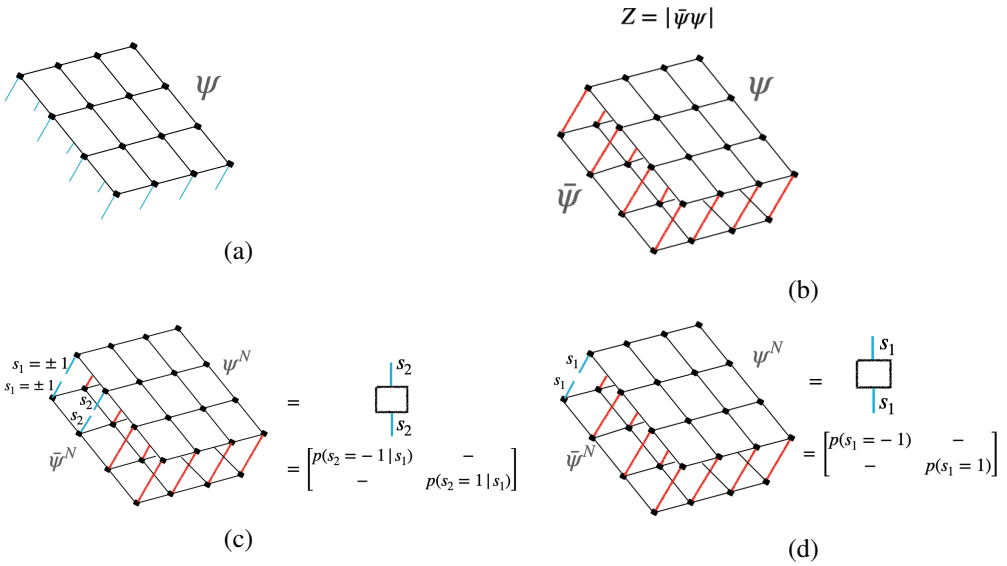


Figure 2: (a) The structure of PEPS model with 4x4 tensors and dangling physical legs shown in blue. (b) Calculation of the scalar Z. One copy of ψ ($\bar{\psi}$) is created and contracted to the original ψ through their physical legs. The red lines indicate the contraction of the physical legs (c) All physical legs of ψ^N (normalized ψ) will be contracted with $\bar{\psi}^N$ through all physical legs except ν_1. The resulting rank-2 tensor will be used to calculated $p(s_1 = \pm 1)$. (d) Calculation of $p(s_2|s_1)$. This time the value of s_1 is already drawn in the previous step.

or,

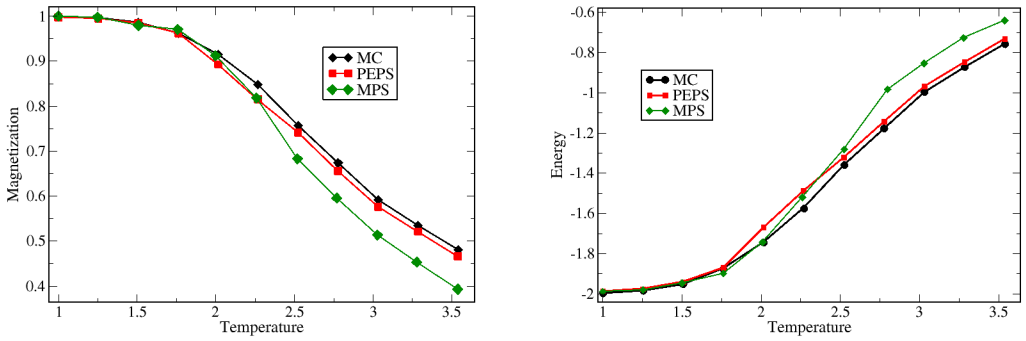
$$p(s_1 = -1) = \sum_{s_2, s_3, \ldots, s_N} \bar{\psi}^N_{-1, s_2, \ldots, s_N} \psi^N_{-1, s_2, \ldots, s_N} \tag{4b}$$

which means that ψ and $\bar{\psi}$ are multiplied to each other for all possible values of s_i with $i = \{2, 3, .., N\}$. Given Equations (4), the sampling of s_1 is drawn based on the values of $p(s1 = \pm 1)$. Once the value of s_1 is determined, the second spin is sampled but conditioned on the value of s_1. Therefore, $p(s_2|s_1)$ follows the same computation in Equations (4b) and (4b), except s_2 is not summed over and s_1 is fixed to the drawn value. Therefore given $p(s_2|s_1)$, one can obtain the value of s_2 by comparing $p(s_2|s_1)$ with a random number between 0 and 1. The processes of computing $p(s_1)$ and $p(s_2|s_1)$ are further illustrated in Figure 2 which shows that summations in Equations (4) are simply equivalent with contraction over corresponding physical legs in PEPS. Direct sampling process from $p(s_2|s_1)$ can be straightforwardly extended to the next s_k for $k = 3, ..., N$. To sample spin s_k, one has to obtain the values of previous spins, $(s_{k-1}, s_{k-2}, ..., s_2, s_1)$ and then calculate $p(s_k|s_{k-1}, s_{k-2}, ..., s_1)$.

It is noteworthy that in this sampling method, the exact contraction is carried out which means all bonds between the tensors have non-zero value. Later in Section 4, we briefly discuss possibles approximations in the contraction of tensors which make the sampling more efficient.

3 Learning 2D Ising Phase Transition

PEPS model performance is studied on the classical Ising model configurations and also compared with the performance of MPS model that was introduced in [11]. The size of PEPS model is set to 4x4 in all simulations and the bond dimension (b) of all connections is fixed to 2. Note that with higher bond dimension values more parameters are used for training, but the drawback is the increase of calculation complexity. For instance, middle tensors in Figure 2 are connected to their 4 surrounding tensors and hence the complexity of contractions will be in the order of (b^4) and so smaller values of b will lead to lower complexity in the learning algorithm. All tensors in PEPS are initialized by random values that are sampled from Gaussian distribution $N(0, 1)$. Also, the Adam optimization [21] with a decaying learning rate is employed.



(a) (b)

Figure 3: Comparison of Magnetization and Energy profiles in MC, MPS, and PEPS configurations. Figure (a) shows a consistency in the magnetization values of the MC and PEPS configurations within the range of temperatures far from the critical points, whereas MPS results are not satisfying except for the low temperatures. In Figure (b) the energy density in PEPS is much closer to that of MC if compared to MPS. However, both MPS and PEPS fail to produce configurations with correct energy values around the critical point.

The classical Ising model configurations will be the training dataset for the PEPS models and it is defined as follows: on a two dimensional square lattice of length L, consider $N = L^2$ classical spins s_i ($i \in \{0, N\}$) that can take on binary values of -1 or +1. Given the Hamiltonian of the system by

$$H = -J/k_B \sum_{\langle i,j \rangle}^{N} s_i \; s_j \tag{5}$$

where $\langle\rangle$ indicates that the summation only runs over nearest neighbor spins and k_B is the Boltzmann constant. Also, for simplicity the coefficient J is chosen such that $J/k_B = 1$. To reach the equilibrium state of the system, we apply the importance sampling Monte Carlo (MC) method. Therefore, from a random initial configuration, at each time step and for a randomly chosen spin s_i, the Metropolis-Hasting flip probability $P_i = min\{1, e^{-\Delta H_i/T}\}$ is calculated to determine whether s_i is accepted to be flipped or not. In this formula, T is the temperature and ΔH_i is the change of energy associated with the change of s_i to its opposite value. We run this simulation for different temperatures and when the system reaches its equilibrium, we evaluate the magnetization density which is defined as

$$M = \frac{1}{L^2} |\sum_{i}^{N} s_i|, \tag{6}$$

and the energy density , which is given by

$$E = \frac{1}{2L^2} \sum_{\langle i,j \rangle}^{N} s_i s_j. \tag{7}$$

In the thermodynamic limit, the two-dimensional classical Ising model shows a second order phase transition from a disordered phase to an ordered phase at $T_c \approx 2.269$. In the ordered phase, spins tend to align with their neighbors, consequently, all spins are in one of the binary directions. In the disordered phase, however, spins in the systems do not have any preferred state to choose and they follow a random distribution.

Figure 3 illustrates the value of magnetization and energy densities as functions of temperature in MC, MPS, and PEPS configurations. For the MPS, we tried different values of bond dimensions such as b= 2,4,8,16. Since our simulation results with the MPS do not significantly change with value of b, we only show our results for $b = 8$. The green graph in both energy and magnetization plots is for MPS results and as it can be clearly seen, at various temperatures specially those higher or closer to T_c, the MPS graph does not match with that of MC, we also repeat similar calculation over TTN

architecture with various bpon dimensions and were not able to detect any improvements (the results are not shown here). Interestingly, PEPS produces configurations with correct magnetization and energy at most of the temperatures which is clearly better than MPS and TTN. At the ordered and disordered phase, while both magnetization and energy learned by PEPS has a good agreement with those in MC, however, the values manifest small deviation at and near the critical point ($T_c \approx 2.269$). This observation can be explained in the following way: in the vicinity of the critical temperature, a finite long range correlation between spins takes place. Note that this range of this correlation increases as the system size become larger. Consequently, it becomes hard for the PEPS model with inherent local structure to capture the long range order and hence the generated configurations will not be physically reliable at critical point. This motivates the extension of Born machine over tensor network architecture known as multi-scale entanglement renormalization ansatz (MERA) which is variational ansatz for ground states of lattice system close to critical point [22].

4 Discussion

A recent generative model known as Born Machine based on family of tensor network ansatz shown a great success in learning various classical and quantum data [11, 12, 23]. In this paper, we design a Born Machine model based on the PEPS architecture and showed the learning power of our model in learning the phase transition of classical Ising model. Our results indicate that the PEPS model successfully learns local and non-local physical information (Magnetization and Energy) in the Ising configurations and generate new configurations with similar physical quality as input configurations. Our results indicated that while the two dimension structure of PEPS leads to a better performance in learning the spin configurations compare to one dimensional tensor network such as MPS and TTN (results not shown here), however, close to critical point it fails to learn the configurations accurately due to increase in the long range correlation in the vicinity of the phase transition. This further motivates the adaption of the Born Machine over another family of tensor network structure known as MERA which are shown to capture the long-range ordering [22]. This is an open research question that we leave for future investigations.

Finally, although PEPS models are excellent tools in learning the information in two dimensional systems, but soon with increasing the bond dimension they become very expensive to evaluate. Higher bond dimensions make the calculation extremely hard and as a result, PEPS models may not be easily extended to larger systems with larger bond dimension values and one requires to adapt new techniques for efficient tensor network contractions. There are several techniques for approximate tensor contractions over PEPS that have been applied for sampling or calculation of partition functions for 2D quantum many-body Hamiltonians, quasi-2D spin-glasses, or low-dimensional discrete optimization problems, including MPS snake-like contractions [24], layer-wise MPS-MPO contractions [17], and corner transfer matrix for tensor contractions [16]. More recently, there are general approaches for approximate tensor network contractions over arbitrary graphs that can be applied to probabilistic graphical models or spin-glass systems [18].

References

[1] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, Adv. in. NeurIPS, 2672-2680, (2014).

[2] D. P Kingma and M. Welling, arXiv:1312.6114 (2013).

[3] G. E. Hinton, *A practical guide to training restricted Boltzmann machines. In Neural networks: Tricks of the trade, (pp. 599-619)* (Springer, Berlin, Heidelberg, 2010).

[4] F. D'Angelo, and L. Böttcher, Phys. Rev. Res **2**,023266, (2020).

[5] A. Azizi and M. Pleimlin, arXiv:2007.09764, (2020).

[6] A. Morningstar and R. G. Melko, J. Mach. Learn. Res. **18**, 1 (2018).

[7] Z. Liu, S. P. Rodrigues, W. Cai, arXiv:1710.04987, (2017).

[8] D. yeverik and R. Melko, arXiv:2004.12867, (2020).

[9] A. Morningstar, and R. Melko, The Journal of Machine Learning Research **18**, 5975 (2017).

[10] J. Chen, S. Cheng, H. Xie, L. Wang, T. Xiang, Phys. Rev. B 97 (8), 085104 , (2018).

[11] Z. Y. Han, J. Wang, H. Fan, L. Wang, P. Zhang, Physical Review X 8 (3), 031012, (2018).

[12] S. Cheng, L. Wang, T. Xiang, P. Zhang, Physical Review B 99 (15), 155131, (2019).

[13] E. M. Stoudenmire and D. J. Schwab, Adv. in NeurIPS 29, 4799 (2016).

[14] A. Novikov, M. Trofimov, and I. Oseledets, arXiv:1605.05775 (2016).

[15] F. Verstraete, M. M. Wolf, D. Perez-Garcia, and J. I. Cirac, Phys. Rev. Lett. 96, 220601 (2006).

[16] R. Orús, and G. Vidal, phys. rev. B, 80, 094403, (2009).

[17] M. M. Rams, M. Mohseni, and B. Gardas, arXiv:1811.06518 (2018).

[18] J.-G. Liu, L. Wang, and P. Zhang, arXiv:2008.06888 (2020).

[19] G. Torlai and R. G. Melko, Phys. Rev. B **94**, 165134 (2016).

[20] K. Najfi, A. Azizi. M. Stoudenmire, Xun Gao, S. Yelin, M. Lukin, and M. Mohseni, in preparation.

[21] D. P. Kingma, and J. Ba (2014).

[22] G. Evenbly, G. Vidal, J. Stat. Phys. 145:891-918, (2011).

[23] Jin-Guo Liu, Yi-Hong Zhang, Yuan Wan, Lei Wang, Phys. Rev. Res. 1, 023025 (2019).

[24] S. Yan, D.A. Huse, and S.R. White, Science 332, 1173 (2011).