Dynamic neuron approach to deep neural networks: Decoupling neurons for renormalization group analysis

Donghee Lee,* Hye-Sung Lee,† and Jaeok Yi‡
Department of Physics, Korea Advanced Institute of Science and Technology, Daejeon 34141, Korea

(Received 6 October 2024; accepted 30 May 2025; published 17 June 2025)

Deep neural network architectures often consist of repetitive structural elements. We introduce an approach that reveals these patterns and can be broadly applied to the study of deep learning. Similarly to how a power strip helps untangle and organize complex cable connections, this approach treats neurons as additional degrees of freedom in interactions, simplifying the structure and enhancing the intuitive understanding of interactions within deep neural networks. Furthermore, it reveals the translational symmetry of deep neural networks, which simplifies the application of the renormalization group transformation, a method that effectively analyzes the scaling behavior of the system. By utilizing translational symmetry and renormalization group transformations, we can analyze critical phenomena. This approach may open new avenues for studying deep neural networks using statistical physics.

DOI: 10.1103/tf5z-7q29

I. INTRODUCTION

Deep learning has demonstrated remarkable performance across a diverse array of fields. Despite this success, its theoretical foundations are still in their early stages, largely due to the numerous degrees of freedom and the complexity of the deep neural network (DNN) system [1,2]. Statistical physics is frequently used to understand complex systems with many degrees of freedom. Consequently, it is natural to expect that applying a statistical physical approach to deep learning will uncover new insights into the field.

Numerous studies have explored the connection between deep learning and statistical physics, driven by the complexity of their behaviors. Several key research areas have been thoroughly documented in the review [3]. Some studies address the relationship between deep learning models employing the gradient descent method and the Gibbs distribution, with the cost function acting as the Hamiltonian under specific conditions [4–10]. In the past, the spin glass model was proposed as a physical framework with similarities to deep learning systems, where the parameters serve as the degrees of freedom [10–13]. Some earlier research suggested that neurons act as the degrees of freedom in deep learning systems, with Hopfield's and Little's work, as well as the Boltzmann machine, being well-known examples [12,14–18].

Some studies have attempted to explain the success of deep learning through the renormalization group (RG) transformation. The hierarchical structure of DNNs has inspired efforts to explore the connection between DNNs and RG transformation. In these studies, unimportant scaled data diminishes as it passes through layers of neurons [19–25]. Additionally, the RG transformation has been suggested as a way to describe the training process of deep learning models [26]. An effective method for initializing DNNs from the perspective of RG has been proposed, and the scaling behavior of information is discussed in Ref. [27]. The RG transformation demonstrates its true power when applied to the study of phase transitions and critical phenomena in systems. Some studies have explored connections between phase transitions, critical phenomena, and deep learning [4,6,8,11,20,27–32].

Recently, intriguing phenomena have emerged in deep learning, particularly involving the (broken) neural scaling law, a principle that seems consistent across different systems. Several papers have studied the neural scaling law, with some suggesting that this behavior may be linked to phase transitions [33–42].

Despite significant efforts to establish a connection between deep learning and statistical physics, a concrete and general understanding of this relationship has yet to be achieved. This challenge partly stems from the complexity of the cost function. In DNNs, the iterative structure and nonlinear activation functions cause the perturbative expansion of degrees of freedom—the synaptic weights and biases—to generate infinitely many non-negligible higher-order terms. In physics, solving problems of this nature often involves identifying underlying symmetries. For instance, translation symmetry plays a key role in analyzing the Ising model [43]. Similarly, recognizing symmetries in a model is crucial for understanding its behavior.

In this paper, we develop a method to reformulate the cost function in a way that reveals its symmetry, thereby reducing its complexity. We refer to this as the "dynamic neuron approach" (DNA). It can be viewed as a decoupling technique that introduces additional degrees of freedom—namely,

*Contact author: dhlee641@kaist.ac.kr
†Contact author: hyesung.lee@kaist.ac.kr
‡Contact author: wodhr1541@kaist.ac.kr

Published by the American Physical Society under the terms of the Creative Commons Attribution 4.0 International license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI.

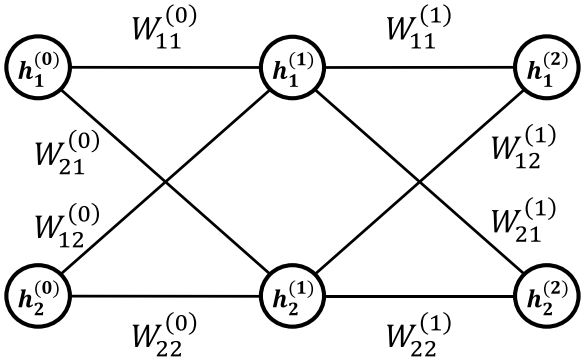


FIG. 1. A DNN with $N = 2$ and $M = 2$, where N is the width of each layer, and M represents the depth of the DNN. For simplicity, the bias term b has been omitted.

neurons in the bulk layers—coupled to the original degrees of freedom. Once these degrees of freedom have been decoupled, the system exhibits translational symmetry akin to that of the Ising model. In a specific case, we show that this approach yields a Hamiltonian resembling that of either the Ising model or the spin-glass model. Due to this symmetry, many tools from statistical physics, including RG transformations and correlation lengths, become applicable to DNNs.

The remainder of this paper is organized as follows. In Sec. II, we provide a brief review of the deep learning and the statistical physics approach. In Sec III, we introduce the DNA and a new Hamiltonian to analyze DNNs. In Sec. IV, we present an example of the RG transformation using our new method. Critical phenomena are discussed in Sec. V. The summary and discussion are provided in Sec. VI, a discussion of the spin glass model limit is included in Appendix B, and the methods for implementing the dynamic neuron system are outlined in Appendix A.

II. FUNDAMENTALS OF DEEP LEARNING

Before proceeding further, we overview the basic concepts of deep learning and its statistical physical approach. This section is largely based on Refs. [8,44,45].

A DNN is a system inspired by the brain [46,47]. It consists of neurons, synaptic weights, and synaptic biases, with neurons organized into layers. The ith neuron in the $(m+1)$th layer, denoted as $h_i^{(m+1)}$, is influenced by all neurons in the mth layer, which are connected via synaptic weights $W_{ij}^{(m)}$ and synaptic biases $b_i^{(m)}$. (See Fig. 1.) This relationship is described by the following equation:

$$h_i^{(m+1)} = \sigma\left(\sum_j W_{ij}^{(m)} h_j^{(m)} + b_i^{(m)}\right) \equiv \sigma_i^{(m)}. \tag{1}$$

In this formula, σ represents a nonlinear function known as the activation function.[1] The repetitive calculation of Eq. (1) determines the value of each neuron, starting from the input (initial layer) $x = h^{(0)}$ and eventually reaching the output (final layer) $z = h^{(M)}$.

Simply put, deep learning is a method for approximating an unknown function using a DNN. The process of building a neural network from a given dataset is called training. There are several types of deep learning classified by their training methods, but we will focus on supervised learning for the sake of simplicity. In supervised learning, the machine is trained to give a good approximation for a set of examples, known as a training dataset. The training dataset consists of L data vectors $x^{[l]}$, $l = 1, 2, \ldots, L$, and their corresponding label vectors $y^{[l]}$. For simplicity, we will often omit $[l]$, provided it does not cause confusion.

The goal of the training process is to minimize the discrepancy between the label $y^{[l]}$ and the output of the DNN $z^{[l]}$ by adjusting the synaptic weights $W_{ij}^{(m)}$ and biases $b_i^{(m)}$.

The discrepancy between the label and the output is typically measured using the L^2- or L^1-norm, referred to as the cost function or training error. The cost function using the L^2-norm is given as[2]

$$C = \sum_{i,l} \left(y_i^{[l]} - z_i^{[l]}\right)^2. \tag{2}$$

A popular method for reducing the cost function is gradient descent. In each step of adjusting the synaptic weights and biases, the gradient of the cost function is calculated, and the synaptic weights and biases are adjusted by subtracting this value:

$$\Delta W_{ij}^{(m)} = -\eta \frac{\partial C}{\partial W_{ij}^{(m)}}, \quad \Delta b_i^{(m)} = -\eta \frac{\partial C}{\partial b_i^{(m)}}, \tag{3}$$

where the step size η determines the magnitude of the weight and bias updates. These equations show that the DNN is constructed by determining the values of the weights and biases, not the neurons. In other words, only the weights and biases are the degrees of freedom in the system, while the neurons serve as placeholders in Eq. (1).

After the training of a DNN is complete, it is used for general data; this process is called inference. The main purpose of deep learning is to train a DNN that makes good inferences. Addressing inference presents many challenges, which we plan to discuss in future work.

From a physical perspective, Eq. (3) represents the equation of motion for the system.[3] To better capture the stochastic nature of training in deep learning, one may consider a Langevin-type equation of motion [48–52], where the update of weights includes both deterministic and stochastic components:

$$\frac{dw}{dt} = -\nabla_w C + \eta(t), \tag{4}$$

[1]Typically, the sigmoid function or rectified linear unit (ReLU) is used as an activation function. An example of the sigmoid function is $\sigma(x) = \frac{1}{1+e^{-x}}$, and for ReLU it is $\sigma(x) = \max(x, 0)$.

[2]Although it is customary to define the cost function by dividing Eq. (2) by the number of data points L, we do not follow this convention. Since L can be absorbed into the step size, it does not affect our discussion.

[3]In Eq. (3), the step size can be interpreted as the time interval between weight updates. Since it does not influence the underlying physics, we set $\eta = 1$.

where $\eta(t)$ represents a white Gaussian noise with zero mean and time correlation $\langle \eta(t)\eta(t')\rangle \propto T\delta_{ij}\delta(t-t')$. Here, T plays the role of an effective temperature that quantifies the strength of fluctuations in the system.

It is known that this equation of motion drives the system to a Gibbs distribution [8,53–55]. Specifically, after a sufficiently long period of training, the probability distribution for a particular synaptic configuration W, b is given by

$$P(W,b) = \frac{1}{Z}e^{-\beta C}, \tag{5}$$

where $\beta = \frac{1}{T}$, and the temperature T arises from the strength of the stochastic noise in the training dynamics, as described above. In this paper, we adopt a framework wherein the synaptic weights of the DNN follow a Gibbs distribution, with the cost function serving as the energy. The partition function, denoted by Z, is defined as

$$Z \equiv \sum_{\{W,b\}} e^{-\beta C}, \tag{6}$$

where $\{W, b\}$ denotes all possible configurations of synaptic weights and biases.[4]

Any expectation value of observables can be calculated as follows:

$$\langle O\rangle = \frac{1}{Z}\sum_{\{W,b\}} O e^{-\beta C}. \tag{7}$$

As one can see, the cost function C plays the role of the Hamiltonian. Therefore, it will be denoted as H from now on.

III. DYNAMIC NEURONS

As mentioned earlier, during training, the change in synaptic weights $W_{ij}^{(m)}$ and biases $b_i^{(m)}$ over time follows the equation of motion defined by the gradient descent. This indicates that synaptic weights and biases represent the system's degrees of freedom. The cost function, or Hamiltonian, governs the probability distribution of degrees of freedom after the training process.

$$H_{\text{original}} = \sum_{i,l}\left(y_i^{[l]} - z_i^{[l]}\right)^2, \tag{8}$$

where

$$z_i^{[l]} = \sigma\left(\sum_{i_1} W_{ii_1}^{(M)}\sigma\left(\sum_{i_2} W_{i_1i_2}^{(M-1)}\times\cdots \times\sigma\left(\sum_{i_M} W_{i_{M-1}i_M}^{(0)} x_{i_k}^{[l]} + b_i^{(0)}\right) + b_i^{(1)}\right)\cdots\right). \tag{9}$$

In this Hamiltonian, the synaptic weights and biases are iteratively plugged into the activation function, as specified by Eq. (1). Although DNNs are usually described as shown in Fig. 1, this iterative structure reveals highly complex interactions between degrees of freedom, as depicted in the upper figure of Fig. 2. This complexity can be understood by considering the Taylor series expansion of the activation function, which involves complicated products of synaptic weights and biases. In particular, even the distant degrees of freedom can interact. Additionally, synaptic weights and biases from different layers contribute differently to H, increasing the asymmetry and complicating the analysis further. To apply various concepts from statistical physics, we need to compute the partition function through the following integral:

$$Z = \int dW\,db\exp[-\beta H_{\text{original}}]. \tag{10}$$

This integration is challenging to compute, as the Hamiltonian consists of intricate terms involving synaptic weights and biases. Therefore, managing the Hamiltonian and the partition function in their original forms is immensely difficult.

To address this difficulty, we promote the neurons $h_i^{(m)[l]}$ to additional dynamic degrees of freedom and ensure that Eq. (1) is dynamically satisfied. Instead of using iterated activation functions, we introduce a series of Dirac delta functions and simplify the exponent to the absolute square as follows:

$$Z = \int dW\,db\,dh\prod_{i,l}\left[\exp\left[-\beta\left(y_i^{[l]} - h_i^{(M)[l]}\right)^2\right] \times\prod_m \delta\left(h_i^{(m+1)[l]} - \sigma_i^{(m)[l]}\right)\right]. \tag{11}$$

Here, i runs over the widths, l runs over the data numbers, and m runs over the layers. We refer to these new dynamic degrees of freedom, $h^{(m)[l]}$, as "dynamic neurons." While this partition function appears straightforward, it does not yet simplify the computation.

To further evaluate the calculations, we employ an approximation of the Dirac delta function[5] as the limit of a Gaussian distribution:

$$\delta(z) \approx \lim_{v\to 0^+}\frac{1}{v\sqrt{2\pi}}\exp\left[-\frac{z^2}{2v^2}\right]. \tag{12}$$

Applying this approximation, the partition function becomes

$$Z = \int dW\,db\,dh\lim_{v\to 0^+}\left(\frac{1}{v\sqrt{2\pi}}\right)^n \times\exp\left[-\beta\sum_{i,l}\left(y_i^{[l]} - \sigma_i^{(M)[l]}\right)^2 - \beta_v\sum_{i,l,m}\left(h_i^{(m+1)[l]} - \sigma_i^{(m)[l]}\right)^2\right], \tag{13}$$

[4]Considering all configurations of the system implies accounting for an infinite amount of time in the training process. However, in most examples from statistical physics, it is widely observed that a sufficiently long experimental run ensures that the system follows the Gibbs distribution. Therefore, if the training time is sufficiently long, we expect our argument to apply to such a system.

[5]Despite the name, the Dirac delta function is not a function but a distribution or generalized function. This means that the approximation may not be strictly valid in the rigorous sense, though it is frequently employed in the literature. We adopt this approximation because the partition function involves integration, and we expect it to yield accurate results once integrated.

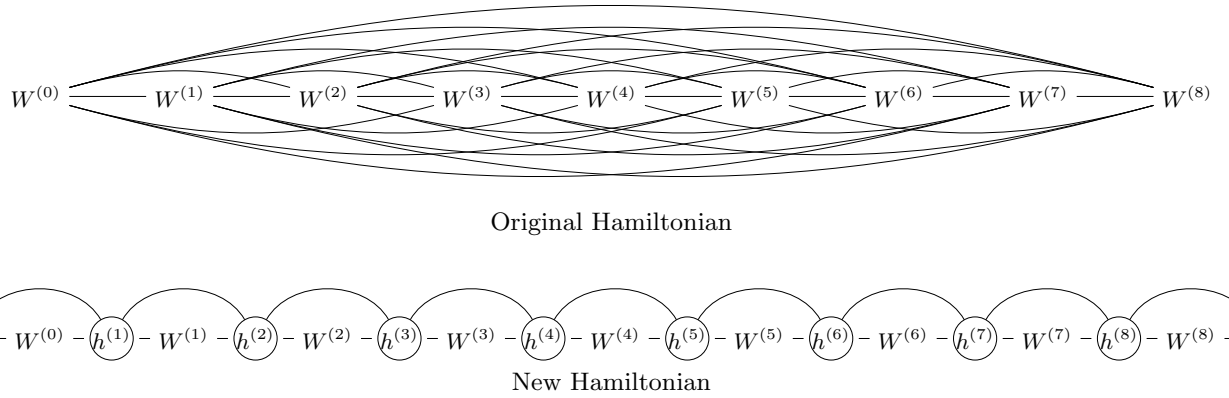


FIG. 2. Interactions in the original and new Hamiltonians. For simplicity, each layer includes only one neuron, and synaptic biases are omitted. The upper figure depicts the original Hamiltonian, while the lower figure illustrates the new Hamiltonian. Degrees of freedom are connected if a direct coupling exists between them. (Self-interactions are omitted in both cases.) Circles in the lower figure represent dynamic neurons, the newly introduced degrees of freedom.

where $\beta_v = 1/2v^2$.[6]

We observe that incorporating the constraint terms reveals the internal structure of a DNN within the Hamiltonian framework. By defining the new Hamiltonian[7] as

$$H_{\text{new}} = \sum_{i,l,m} \lambda_m \left(h_i^{(m+1)[l]} - \sigma_i^{(m)[l]}\right)^2 \tag{14}$$

with $\lambda_m = \beta_v/\beta$ except for $\lambda_M = 1$, we can denote Eq. (13) in the typical way of describing the partition function,

$$Z = \int dW\, db\, dh \exp[-\beta H_{\text{new}}]. \tag{15}$$

where the limit over v is omitted for simplicity. The form of this new Hamiltonian is the summation of interactions between the nearest layers [mth layer and $(m+1)$th layer]. In other words, interactions in the new Hamiltonian are restricted to the nearest degrees of freedom following the same pattern, unlike the original Hamiltonian, thereby implying locality.

Physically, this approach can be viewed as a decoupling strategy. Dynamic neurons are introduced between the original degrees of freedom, effectively simplifying the complex interactions between weights and biases. These interactions are decoupled and replaced with simpler interactions between dynamic neurons and their nearest weights and biases, analogous to using power strips to organize and untangle cables. The structure of the new Hamiltonian is illustrated in the lower part of Fig. 2, where the original Hamiltonian's complex structure shown in the upper figure is decoupled.

Although we introduced the infinite β_v in the previous discussion, we will now study a more general system with an arbitrary positive β_v. When β_v is sufficiently large, the system is expected to behave similarly to the infinite β_v case. In other words, we are considering a generalization of deep learning in which the limit of large β_v corresponds to the typical DNN.

Since Eq. (14) is written as a sum of terms with the same form, the Hamiltonian is invariant under translation,

$$h_i^{(m)[l]} \to h_i^{(m+1)[l]}, \quad \sigma_i^{(m)[l]} \to \sigma_i^{(m+1)[l]} \tag{16}$$

except at the boundaries (the first and last layers). This approximate symmetry is not easily identifiable in the original Hamiltonian given by Eqs. (8) and (9).

The emergence of this type of translation symmetry can simplify calculations and lead to interesting consequences. For example, in the one-dimensional and two-dimensional Ising models, the transfer matrix can be calculated explicitly due to translational symmetry. Therefore, it is expected that the new Hamiltonian can be analyzed by utilizing this symmetry. As an example, we perform the RG transformation in Sec. IV.

This symmetry is approximate due to the system's boundaries. However, it is generally assumed that as the system size increases, the symmetry becomes more precise. Similarly, in a DNN with considerable depth, the symmetry becomes nearly exact. Since depth is regarded as a key factor in the success of deep learning [56–60], this approximate symmetry may offer insights into the role of depth in understanding neural networks.

The new Hamiltonian in Eq. (14) is expressed as the sum of the original cost function (for $m = M$) and additional constraint terms, each multiplied by a constant λ_m (for $m \neq M$).This approach is analogous to the use of Lagrange multipliers in Lagrangian mechanics. Notably, similar terms arise when the Yang-Mills Lagrangian is quantized using the Fadeev-Popov procedure in quantum field theory [61], although we will not discuss this further here.

Moreover, this system can be effectively described using principles from statistical physics, which are often employed to uncover the macroscopic behavior of complex systems composed of similarly interacting units.

With certain simplifying assumptions, this approach yields a Hamiltonian that closely resembles the spin glass model,

[6]Although distinct values of v could be assigned to each term, we assume v remains the same for all terms. If a limit exists as multiple distinct values of v approach 0, the same limit will be achieved when the values of v are set identically and approach 0. This can also be understood through the concept of a Lagrange multiplier. If we interpret β_v as the Lagrange multiplier, it imposes the constraint that the sum of the absolute square terms must vanish. For this to occur, each individual term must also vanish.

[7]If the cost function is defined using an L^p-norm, we can define the new Hamiltonian as $H_{\text{new}} = \sum \lambda |h - \sigma|^p$.

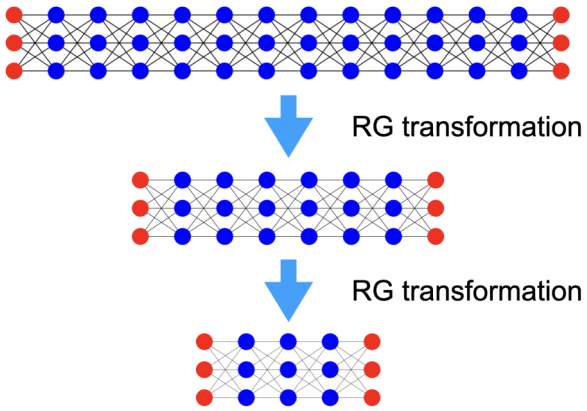


FIG. 3. RG transformation applied to the DNN using the DNA, resulting in reduced network depth. A notable similarity with the Ising model is apparent.

a well-established framework in statistical physics.[8] (See Appendix B.) This approach offers a new method for applying the spin glass model to gradient descent-based training in DNNs. Also, since we represented the training process of a general DNN using tools from statistical physics, we can inversely implement a system governed by such a Hamiltonian through the training of a DNN. (See Appendix A.)

IV. RENORMALIZATION GROUP TRANSFORMATION

Since the new Hamiltonian resembles well-known statistical models, we can apply statistical physics tools to analyze the training of DNNs. A promising method in statistical physics is the RG transformation, which is widely used in fields such as condensed matter and high-energy physics. This technique systematically simplifies a system by progressively integrating out smaller-scale fluctuations, leading to a renormalized system with fewer degrees of freedom.

To understand how the components of the DNN function, we apply the RG transformation with respect to depth using our new Hamiltonian, as shown in Fig. 3.[9] We begin by examining a simple scenario under specific assumptions. While these assumptions may oversimplify the system, potentially differing from actual DNNs, this example is designed to illustrate how the RG transformation can be applied to DNN within the context of the new Hamiltonian.

(i) The system consists of one neuron per layer ($N = 1$) with a large depth (M is large), and the training dataset contains a single data point ($L = 1$). This assumption may be oversimplification and the example could be quite far from actual DNNs. In fact, the Modified National Institute of Standards and Technology (MNIST) [62] data consist of 60 000 training examples ($L = 60\,000$), and the universal approximation theorem prefers a sizable width with $N > 1$ [63–67]. A lack of data can lead to insufficient training and overfitting, while insufficient depth may undermine the validity of the universal approximation theorem. It should be highlighted that our approach can be generally applied to DNNs with arbitrary N, M, L. Nevertheless, we make this assumption because our primary interest here lies in analytically demonstrating the statistical physics properties.

(ii) The activation function is the Heaviside step function:

$$\sigma(x) = \begin{cases} 0, & x < 0, \\ 1, & \text{otherwise.} \end{cases} \tag{17}$$

Although the Heaviside step function is rarely used in modern DNNs due to its zero gradient, it was employed in earlier models and applications [46,68].

(iii) The weights W and biases b are restricted to values of -1 or 1, and the neurons h are limited to values of 0 or 1. While, in theory, h should be integrated over all possible values, we expect the Gaussian-like behavior of the partition function to constrain h to values near 0 or 1. Although in general DNNs W and b can take arbitrary real numbers, we restrict their values here for simplicity. Interestingly, recent research has explored neural networks where parameters are similarly restricted to discrete values, such as $-1, 1$ or $-1, 0, 1$. Despite these limitations, that model appears to achieve good performance [69,70].

Given the restrictions on the available values, integration in the partition function is replaced by summation. Thus, the partition function for this example is written as

$$Z = \sum_{\{W,b,h\}} \exp[-\beta H], \tag{18}$$

where the summation is over all possible values of W, b, and h. Since we are considering the case with $N = 1$ and $L = 1$, the new Hamiltonian in Eq. (14) can be expressed as a sum:

$$H = \sum_m \lambda_m H_m, \tag{19}$$

where each H_m has the common form

$$H_m = [h^{(m+1)} - \sigma(W^{(m)} h^{(m)} + b^{(m)})]^2. \tag{20}$$

As discussed in Sec. III, the Hamiltonian exhibits translational symmetry similar to that of the Ising model. Consequently, we can apply the RG transformation in a manner similar to its application in the Ising model. One approach involves summing over $W^{(m)}$, $b^{(m)}$, and $h^{(m)}$ for even-numbered layers in the partition function. After performing this summation, the partition function describes a system with fewer degrees of freedom. This real-space RG transformation method is known as decimation [71,72], and the translational invariance or periodicity plays a crucial role in performing the decimation.[10]

[8] Several approaches have already introduced spin glass models to analyze DNNs under various assumptions [10–17].

[9] After an RG transformation, the system's degrees of freedom are reduced, making the Hamiltonian primarily distance-dependent along the depth direction. This complicates applying the RG transformation with respect to the width, as maintaining the Hamiltonian's form during such a transformation is challenging.

[10] Decimation is generally avoided for systems with dimensions higher than one because it fails to account for the renormalization of the magnitude of the degrees of freedom, preventing it from achieving a true fixed point in higher dimensions.

Extracting the terms related to $W^{(m)}$, $b^{(m)}$, and $h^{(m)}$ from the partition function, we have

$$Z_m = \sum_{\{W^{(m)},b^{(m)},h^{(m)}\}} \exp[-\beta H_{m+1} - \beta H_m]. \tag{21}$$

This summation is finite and can be calculated without difficulty:

$$Z_m = 3u^{-2}u^{2h+2\sigma} + u^{-1}(u^{2\sigma} + 2u^{2h}) + 2, \tag{22}$$

where

$$\begin{aligned} h &= h^{(m+1)}, \\ \sigma &= \sigma(W^{(m-1)}h^{(m-1)} + b^{(m-1)}), \\ u &= e^{\beta_v}. \end{aligned} \tag{23}$$

By taking the logarithm, we obtain the Hamiltonian of the renormalized system, denoted as H':

$$-\beta H'_m = \ln[3u^{-2}u^{2h+2\sigma} + u^{-1}(u^{2\sigma} + 2u^{2h}) + 2]. \tag{24}$$

By making use of the restricted configuration, where $h^2 = h$ and $\sigma^2 = \sigma$, we can express the renormalized Hamiltonian in polynomial form:[11]

$$\begin{aligned} -\beta H'_m &= h\sigma \ln(3u^2 + 3u + 2) \\ &\quad + (1-h)\sigma \ln(u + 2u^{-1} + 5) \\ &\quad + h(1-\sigma)\ln(2u + u^{-1} + 5) \\ &\quad + (1-h)(1-\sigma)\ln(3u^{-2} + 3u^{-1} + 2) \\ &= h\sigma \ln\frac{(3u^2+3u+2)(2u^2+3u+3)}{(u^2+5u+2)(2u^2+5u+1)} \\ &\quad + h^2 \ln\frac{u(2u^2+5u+1)}{2u^2+3u+3} \\ &\quad + \sigma^2 \ln\frac{u(u^2+5u+2)}{2u^2+3u+3} \\ &\quad + (\text{constant}). \end{aligned} \tag{25}$$

In general, suppose that our Hamiltonian takes the following form:

$$-\beta H = \sum_m (Ah\sigma + Bh + C\sigma + D). \tag{26}$$

The coupling constant A has two critical aspects worth emphasizing. First, it introduces the interaction between neurons across different layers, effectively encoding the mechanism of information propagation within the system. Consequently, any changes in A under RG transformation are expected to significantly influence the learning dynamics and overall behavior of the deep learning model. Second, this coupling term bears resemblance to the spin-spin interaction term in the Ising model. In the Ising model, the spin-spin interaction is the primary driver behind phase transitions, reflecting collective behavior changes at critical points. Similarly, in the context of the dynamic neuron system, A is likely to play a pivotal role in phase transitions, potentially governing shifts in the system's operational regimes or learning efficiency. Understanding this analogy provides a deeper insight into how critical phenomena might emerge in such neural systems.

[11]This approach of representing complex functions as polynomials is also observed in the RG transformation of the Ising model.

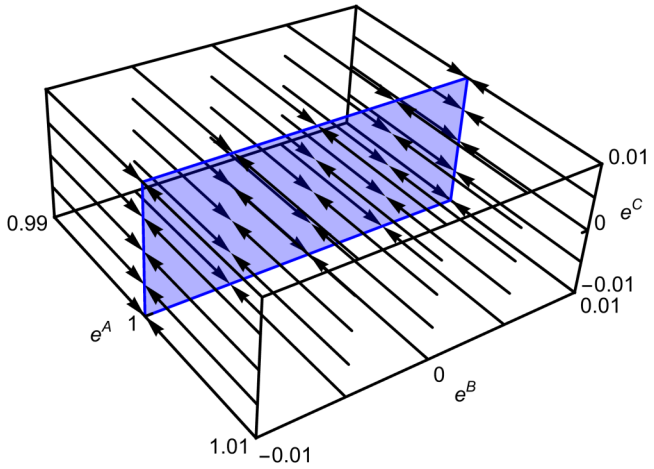


FIG. 4. RG flow near the fixed points $e^{A^*} = 1$, demonstrating that RG flows converge to the plane composed of fixed points.

Under the RG transformation, the constant term in the Hamiltonian only affects the non-singular part of the free energy and is typically ignored when calculating the critical properties of the system.

$$\begin{aligned} A' &= \ln\left[\frac{2+2e^{A+C}+e^{A+B}+3e^{2A+B+C}}{2+2e^{A+C}+e^{B}+3e^{A+B+C}} \right. \\ &\quad \left. \times \frac{2+2e^{C}+e^{B}+3e^{B+C}}{2+2e^{C}+e^{A+B}+3e^{A+B+C}}\right], \\ B' &= B + \ln\frac{2+2e^{A+C}+e^{B}+3e^{A+B+C}}{2+2e^{C}+e^{B}+3e^{B+C}}, \\ C' &= C + \ln\frac{2+2e^{C}+e^{A+B}+3e^{A+B+C}}{2+2e^{C}+e^{B}+3e^{B+C}}. \end{aligned} \tag{27}$$

The fixed points are the points that the renormalized system is equal to itself. By obtaining the solutions $(A,\ B,\ C)$ of the equation

$$(e^{A'}, e^{B'}, e^{C'}) = (e^A, e^B, e^C), \tag{28}$$

the fixed points (A^*, B^*, C^*) of this system can be found. The exponents are used to simplify the calculations. The fixed points are given as

$$e^{A^*} = 1 \tag{29}$$

for arbitrary values of e^{B^*} and e^{C^*}. Figure 4 illustrates the RG flows near these fixed points. The RG flows converge to the fixed points, indicating that there is no phase transition.

Since the couplings are real, there are no fixed points other than those given in Eq. (29). However, if we assume that e^{A^*}, e^{B^*}, and e^{C^*} can be negative, we can find another fixed point at

$$e^{A^*} = 3, \quad e^{B^*} = -\tfrac{2}{3}, \quad e^{C^*} = -\tfrac{1}{3}. \tag{30}$$

The RG flow near this fictitious fixed point is shown in Fig. 5, where we observe that the RG flows diverge from this point.

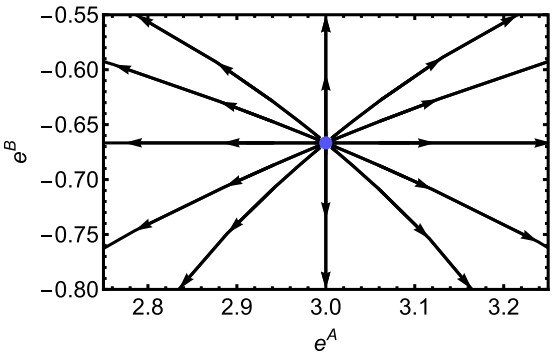


FIG. 5. Diverging RG flow near the fictitious fixed point $(e^{A^*}, e^{B^*}, e^{C^*}) = (3, -2/3, -1/3)$. For visibility, $e^C = -1/3$ is fixed for all flows.

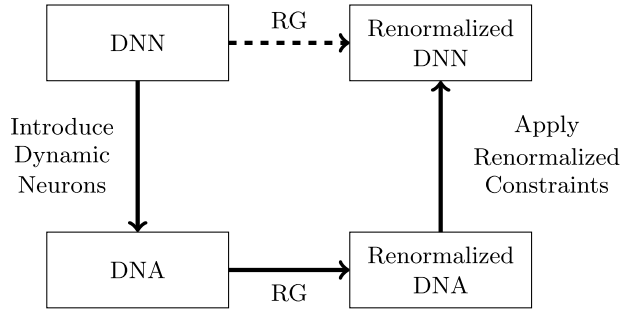


FIG. 6. This diagram illustrates how to develop a renormalized DNN using the DNA. Although not a rigorous explanation, it effectively conveys the essential concept. By introducing dynamic neurons, we obtain a new description of the DNN, and from the advantages of this new description we derive a renormalized DNN with DNA. Finally, by applying renormalized constraints, the dynamic neurons can be integrated out, resulting in a renormalized DNN.

This fixed point appears unphysical since it can only be achieved when the couplings are imaginary. Nevertheless, we anticipate that more general and complex deep learning models, beyond our simplifying assumptions (i)–(iii), will exhibit unstable fixed points, or critical points, where RG flows diverge within realistic coupling regions.[12] Near unstable fixed points, critical phenomena can be discussed. The critical phenomena and scaling laws near this fictitious fixed point are studied in Sec. V.

Let us discuss the physical meaning of the RG transformation in the DNA. As discussed earlier, the introduction of dynamic neurons results in additional terms, $\lambda_m H_m$, which reflect the constraints

$$H_m = [h^{(m+1)} - \sigma(W^{(m)}h^{(m)} + b^{(m)})]^2 = 0, \tag{31}$$

equivalent to Eq. (1). This constraint explains the relationship between the nearest degrees of freedom. After the RG transformation, the degrees of freedom in the mth layer are integrated out, and the renormalized Hamiltonian H'_m is composed of the nearest degrees of freedom.[13] As Eq. (31) reflects the constraints between the nearest degrees of freedom, the vanishing conditions of the renormalized Hamiltonian

$$H'_m = 0 \tag{32}$$

can be interpreted as new constraints between the nearest degrees of freedom in the renormalized system. As the depth of the DNN decreases by applying the RG transformation, new constraints for the system with reduced depth emerge. Therefore, the RG transformation suggests statistically equivalent constraints for the system with reduced depth. Theoretically, by applying these new constraints to the partition function, the introduced dynamic neurons can be integrated out. Subsequently, a renormalized DNN system without dynamic neurons can be obtained. In general, after applying the RG transformation, the constraints often become more complex, leading to multiple solutions or analytically unsolvable equations. However, even if we cannot solve them explicitly, identifying these constraints can aid in understanding the structure of DNNs. In this sense, the DNA is a valuable tool for comprehending the nature of DNNs. This argument can be generalized without relying on the simplifying assumptions (i)–(iii).

The discussions in this section are schematically summarized in Fig. 6. Because the RG transformation is a powerful technique in statistical physics, we aim to use it to analyze DNNs. However, directly applying the RG transformation to DNNs is challenging, so we employ the DNA. Due to the translational symmetry that emerges from the DNA, the RG transformation becomes much easier to apply. When the DNN undergoes an RG transformation using the DNA, it suggests new constraints for the system with reduced depth. By applying these constraints, one can obtain a renormalized DNN system.

Before concluding this section, let us discuss how to relax the assumptions we introduced earlier in this section for the sake of demonstration. Using the decimation-based RG transformation may not be suitable when relaxing these assumptions. If assumption (i) is relaxed by increasing the sizes of L and N, decimation can still be applied, but the number of couplings increases, making it challenging to analytically determine the fixed points. If assumption (ii) is generalized to allow different activation functions, assumption (iii) concerning the available values of the dynamic neuron must also be extended accordingly. If assumption (iii) is relaxed, expanding the available values of synaptic weights, biases, and dynamic neurons could result in an infinite number of possible values. In this case, polynomial expansion in decimation would no longer be feasible.

Nevertheless, these challenges arise from the difficulty of applying decimation, not from the limitations of the DNA itself. DNA can be applied regardless of assumptions (i), (ii), and (iii). Referring to the equations we presented in the previous section, extending the DNA to more general scenarios is straightforward. Statistical physics offers numerous RG transformation methods that do not rely on decimation. Applying

[12]This reasoning is grounded in insights from statistical physics. For instance, while the one-dimensional Ising model lacks a phase transition at any finite temperature, the two-dimensional Ising model does undergo such a transition.

[13]Since the mth layer is integrated out, the $(m-1)$th and $(m+1)$th layers become the nearest layers.

these methods could enable further research on DNNs based on the DNA without these assumptions.

V. CRITICAL PHENOMENA

Critical phenomena are dramatic behaviors that occur near critical points and are crucial in understanding phase transition physics. One key feature of critical phenomena is the scaling law, where physical quantities exhibit power-law behavior. A remarkable aspect of critical phenomena is universality, which means that many systems can exhibit the same scaling laws despite microscopic differences, provided they share key characteristics.

To make predictions about the scaling properties of general deep learning, we further study the fictitious fixed point from the example in the previous section, even though it is not physical. By considering the linear approximation of the RG transformation near the critical point, we can analyze the critical properties of the system:

$$\begin{aligned}\vec{K}' - \vec{K}^* &= \mathcal{R}(\vec{K}) - \mathcal{R}(\vec{K}^*) \\ &\approx \left.\frac{d\mathcal{R}}{dK}\right|_{K=K^*} (K - K^*),\end{aligned} \tag{33}$$

where $\vec{K} = (e^A, e^B, e^C)$, $\vec{K}' = (e^{A'}, e^{B'}, e^{C'})$, $\vec{K}^* = (3, -2/3, -1/3)$, and $\mathcal{R}(\vec{K}) = \vec{K}'$ denotes the function of RG transformation. Then, near $\vec{K}^*$, the RG transformation can be approximated as follows:

$$\begin{aligned}\begin{bmatrix} e^{A'} - 3 \\ e^{B'} + 2/3 \\ e^{C'} + 1/3 \end{bmatrix} &\approx \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix} \begin{bmatrix} e^{A} - 3 \\ e^{B} + 2/3 \\ e^{C} + 1/3 \end{bmatrix} \\ &= \begin{bmatrix} p^{y_A}(e^{A} - 3) \\ p^{y_B}(e^{B} + 2/3) \\ p^{y_C}(e^{C} + 1/3) \end{bmatrix},\end{aligned} \tag{34}$$

where $p = 2$ denotes the ratio between the depth of the original and renormalized systems.[14] The values $y_A = 1$, $y_B = 1$, and $y_C = 1$ determine the universality class of the system. The values may seem trivial here, but this is due to the simplicity of the system we are considering.

From Eq. (34), we can find the scaling laws of observables near the fixed point. As an example, let us introduce the scaling of the correlation function. For simplicity, we explore the scaling laws under the conditions $B = B^*$ and $C = C^*$, though this argument can easily be generalized.

The correlation function of neurons is defined as

$$\begin{aligned}G_h(h^{(m)}, h^{(n)}, A - A^*) &= \langle h^{(m)} h^{(n)} \rangle_H - \langle h^{(m)} \rangle_H \langle h^{(n)} \rangle_H \\ &= G_h(m - n, A - A^*),\end{aligned} \tag{35}$$

where $\langle \cdot \rangle_H$ denotes ensemble average with respect to the Hamiltonian H.[15] Note that the last equality holds due to translational invariance. As stated earlier, translational invariance was not readily apparent in the original cost function. This implies that our approach reveals the underlying correlating properties of deep learning.

The correlation length ξ is defined from the behavior of the correlation function, which means that

$$G_h(m - n, A - A^*) \propto e^{-|m-n|/\xi} \tag{36}$$

for sufficiently large $|m - n|$.

After the RG transformation, the difference in depth between two neurons decreases by the ratio p. Since we are calculating the correlation function of neurons h, the coupling constants associated with the neurons, B, affect the scaling of the correlation function. Furthermore, the dimension of the system d, along with the degree of freedom we are reducing, affects the RG transformation. In our example, we consider the RG transformation with respect to depth, so $d = 1$.

In general, the following scaling behavior can be found near the critical point:

$$G_h\left(\frac{m - n}{p}, A' - A^*\right) = p^{2d-2y_B} G_h(m - n, A - A^*). \tag{37}$$

Equation (37) is the core of the scaling law for the correlation function and correlation length.

Considering $p|A - A^*| = A_0$ for some positive constant A_0, and substituting $p = (|A - A^*|/A_0)^{-1/y_A}$, one can conclude that

$$G_h(m - n, A - A^*) \propto \Psi\left((m - n)\left|\frac{A - A^*}{A_0}\right|^{1/y_A}\right) \tag{38}$$

for some function Ψ. For large $m - n$, we expect the correlation function to follow Eq. (36), so we find the scaling law of the correlation length:

$$\xi \propto |A - A^*|^{-1/y_A}. \tag{39}$$

Now, considering that $|m - n|/p = r_0$ for some fixed distance r_0, and repeating the same argument, one can conclude that

$$G_h(m - n) \propto |m - n|^{-2d+2y_B} \tag{40}$$

near the critical point.[16] These scaling laws are frequently observed in various examples of critical phenomena in statistical physics. Further research from the statistical physics perspective could provide deeper insights into these phenomena.

Equation (13) implies that, in the limit of large β_v, the DNA becomes equivalent to the DNN. To examine this correspondence, we performed a Monte Carlo simulation of the simplified neural network introduced in the previous section. The network depth was set to 10^4, and we used $\beta = \beta_v = 1$. As shown in Fig. 7, the time-averaged correlation functions of both systems exhibit a close resemblance. This observation suggests that, under suitable conditions, the DNA can serve as a reasonable theoretical approximation of the DNN. Owing to

[14] In general, by taking the continuum limit of the system, the parameter p can be extended to real numbers. This enables the definition of an infinitesimal RG transformation with parameter $p = 1 + \epsilon$, where ϵ is an infinitesimally small quantity.

[15] We can also define correlation functions for different variables, such as $G_\sigma(\sigma^{(m)}, \sigma^{(n)}, A - A^*)$.

[16] Here, we define "distance" as the number of layers separating two neurons. In more complex deep neural network architectures, this concept may need a more refined definition.

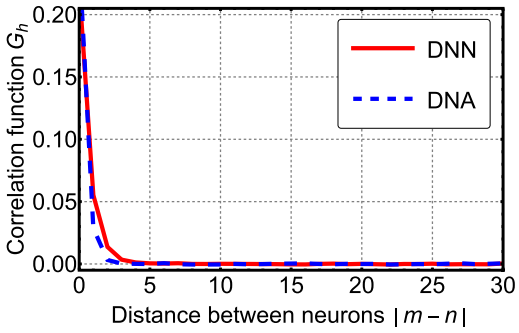


FIG. 7. Results of a Monte Carlo simulation for a simple neural network (red solid) and its dynamic neuron counterpart (blue dashed). We computed the time-averaged correlation function of neuron activations as a function of spatial separation, averaged over all positions. The two systems exhibit a close resemblance.

its analytical tractability, the DNA provides a useful perspective for further investigations into the structure and dynamics of deep neural networks.

VI. SUMMARY AND DISCUSSION

The statistical physical approach to deep learning has been explored in many studies, but there is no agreed-upon description. Additionally, the dynamics of neurons have only been discussed in specific models, such as the Hopfield network and the Boltzmann machine, which differ fundamentally from typical deep learning systems [14,15,18].

Our study introduced the DNA. In this approach, neurons are treated as dynamic variables of the system. We transformed the original Hamiltonian, where the degrees of freedom are synaptic weights and biases, into a new Hamiltonian with additional degrees of freedom: dynamic neurons. The DNA with such a transformed system could be understood as a generalization of deep learning, where the limit of large β_v corresponds to the typical DNN.

The new Hamiltonian simplifies the interactions and reveals translational invariance. This symmetry, induced by the DNA, is approximate because the first and last layers are ignored. However, as the number of layers increases, the relative effects of the first and last layers diminish, and the symmetry becomes better preserved. This observation could be related to a fundamental question about DNNs: why do they perform so well when their depth is large? Our approach might offer insights into many fundamental questions about DNNs [73].

Additionally, the DNA highlights that the degrees of freedom—dynamic neurons, synaptic weights, and synaptic biases—interact primarily with their nearest neighbors. This interaction is intuitive, as neurons in a network communicate only with adjacent neurons to produce the final output. Consequently, the entire system can be understood by examining its individual components and their interactions.

We discussed the RG transformation and critical phenomena using a simple example of a DNN with dynamic neurons. Although it was a straightforward case, we demonstrated that well-known statistical approaches can be easily applied to DNNs using the DNA. RG transformations and critical phenomena are deeply studied topics in statistical physics with many physical implications. Finding connections between these implications and the properties of DNNs is very intriguing and promises rich outcomes.

Although our study explains the statistical property of the specific example, it should be emphasized that the DNA can be applied to any DNN with arbitrary width, depth, and the number of data points. Many studies on DNNs concentrate on factors such as network size and the amount of training data. We anticipate that our DNA can become a useful tool to investigate those subjects.

In the present paper, given the limited theoretical development surrounding inference, we have focused on building a model that analytically explains the statistical properties of the training process. Addressing inference from a theoretical perspective remains challenging, as it involves understanding how neural networks generalize to unseen data sets. Future work should therefore develop theoretical tools for inference that would enable investigations into phenomena such as the neural scaling law [33–42] and its relevance to our theory. These advances will help bridge the gap between the statistical properties of training and predictive performance in practical applications.

ACKNOWLEDGMENTS

This work was supported in part by the National Research Foundation of Korea (Grant No. RS-2024-00352537). Author names in this work are in alphabetical order.

APPENDIX A: TECHNICAL APPLICATION

This appendix delves into the technical implementation of the dynamic neuron system. Since the DNA modifies the Hamiltonian and introduces new degrees of freedom, referred to as "dynamic neurons," it is crucial to analyze both the altered system dynamics and the behavior of these newly introduced components.

The new Hamiltonian described in Eq. (14) emerges from the dynamic neuron framework and governs the system's behavior. Consequently, this introduces additional equations of motion for dynamic neurons and modifies the equations of motion for synaptic weights accordingly. To implement such a system, we can utilize the gradient descent method, as outlined in Eq. (3), in the following manner:

$$\Delta W_{ij}^{(m)} = -\eta \frac{\partial H_{\text{new}}}{\partial W_{ij}^{(m)}},$$
$$\Delta b_i^{(m)} = -\eta \frac{\partial H_{\text{new}}}{\partial b_i^{(m)}},$$
$$\Delta h_i^{(m)[l]} = -\eta \frac{\partial H_{\text{new}}}{\partial h_i^{(m)[l]}}. \tag{A1}$$

We need to implement changes for the dynamic neurons for each data represented by l. Under the lens of our theoretical framework, we anticipate that when β_v is large enough the conventional DNN and the dynamic neuronal DNN will exhibit only minor differences in their statistical properties over the same training epochs.

However, it seems that constructing a DNN using the DNA is impractical at this point. In this framework, the number of dynamic neurons introduced is LMN, corresponding to the product of the number of data points, the width of the network, and its depth. This implies that additional LMN equations of motion must be solved to implement the system. Given the large datasets typically used in training, this approach appears computationally infeasible for practical applications.

Nonetheless, we remain open to the potential technical applications of this approach. Despite its current inefficiency, the development of better optimization strategies could enable the practical realization of the dynamic neuron framework. In many deep learning implementations, Eqs. (3) are not solved simultaneously; instead, more efficient methods like backpropagation are commonly used for training. A similar method could be designed specifically for DNNs based on the DNA. The inherent homogeneity in the equations of motion, as highlighted in our main text, could serve as a basis for innovative optimization techniques. Moreover, since interactions in the DNA are restricted to nearest neighbors, the right-hand sides of Eqs. (A1) are expected to be simpler than those of Eqs. (3). With further research into such optimization methods, we expect that the practical realization of the DNA could become feasible.

APPENDIX B: SPIN GLASS

In this appendix, we illustrate how the new Hamiltonian in the DNA replicates key characteristics of spin glass models, a class of statistical models renowned for their complex energy landscapes and disordered states. While there are several formulations of spin glass models, we focus on the Edwards-Anderson model, a canonical example widely used in statistical physics [74,75]:

$$H = \sum_{i,j} J_{ij} s_i s_j + \sum_i K_i s_i, \tag{B1}$$

where s_i are spin degrees of freedom, which take values of 1 or -1, and the coupling constants between spins J_{ij} are typically assumed to follow independent probability distributions, such as Gaussian or Bernoulli distributions. K_i is the coupling to magnetic fields, which depends on an external magnetic field.

We apply certain conditions and assumptions to the new Hamiltonian presented in Eq. (14). We suppose the activation function is the sigmoid function, so $|\sigma|$ and $|h|$ are likely smaller than 1. Also, we introduce the regularization terms to restrict W and b from becoming too large. Furthermore, we expect that the linear order expansion of the sigmoid function is a valid approximation.

The new Hamiltonian, without the constant term, is then given as

$$H = \sum_{i,j,l,m} \lambda_m W_{ij}^{(m)} h_i^{(m+1)[l]} h_j^{(m)[l]} + \sum_{i,l,m} \lambda_m b_i^{(m)} h_i^{(m+1)[l]} + \sum_{i,j,m} c_W \left(W_{ij}^{(m)}\right)^2 + \sum_{i,m} c_b \left(b_i^{(m)}\right)^2. \tag{B2}$$

where the last two terms are regularization terms. These regularization terms provide the Gaussian probability distribution for W and b and result in the Gaussian distributed coupling constants for the first two terms in the Hamiltonian. Then the first two terms of Hamiltonian are similar to the spin glass model with a space-varying external magnetic field with degrees of freedom h and coupling constants determined by W and b.[17]

[17] One should note that, despite these similarities, there are differences between spin glass models and the new Hamiltonian: (1) The couplings can vary in time, and (2) The spin degrees of freedom can take arbitrary values between -1 and 1.

[1] F.-L. Fan, J. Xiong, M. Li, and G. Wang, On interpretability of artificial neural networks: A survey, IEEE Trans. Radiat. Plasma Med. Sci. **5**, 741 (2021).

[2] X. Li, H. Xiong, X. Li, X. Wu, X. Zhang, J. Liu, J. Bian, and D. Dou, Interpretable deep learning: Interpretation, interpretability, trustworthiness, and beyond, Knowl. Inf. Syst. **64**, 3197 (2022).

[3] Y. Bahri, J. Kadmon, J. Pennington, S. S. Schoenholz, J. Sohl-Dickstein, and S. Ganguli, Statistical mechanics of deep learning, Annu. Rev. Condens. Matter Phys. **11**, 501 (2020).

[4] P. Carnevali and S. Patarnello, Exhaustive thermodynamical analysis of Boolean learning networks, Europhys. Lett. **4**, 1199 (1987).

[5] N. Tishby, E. Levin, and S. A. Solla, Consistent inference of probabilities in layered networks: Predictions and generalizations, in *Proceedings IEEE International Conference on Neural Networks* (IEEE, Piscataway, NJ, 1989), pp. 403–409.

[6] H. Sompolinsky, N. Tishby, and H. S. Seung, Learning from examples in large neural networks, Phys. Rev. Lett. **65**, 1683 (1990).

[7] E. Levin, N. Tishby, and S. A. Solla, A statistical approach to learning and generalization in layered neural networks, Proc. IEEE **78**, 1568 (1990).

[8] H. S. Seung, H. Sompolinsky, and N. Tishby, Statistical mechanics of learning from examples, Phys. Rev. A **45**, 6056 (1992).

[9] A. Engel, *Statistical Mechanics of Learning* (Cambridge University Press, Cambridge, 2001).

[10] M. Mezard and A. Montanari, *Information, Physics, and Computation* (Oxford University Press, Oxford, 2009).

[11] G. Györgyi, First-order transition to perfect generalization in a neural network with binary synapses, Phys. Rev. A **41**, 7097 (1990).

[12] R. Koebarle and W. K. Theumann, *Neural Networks and Spin Glasses—Proceedings of the Statphys 17 Workshop* (World Scientific, Singapore, 1990).

[13] M. Advani, S. Lahiri, and S. Ganguli, Statistical mechanics of complex neural systems and high dimensional data, J. Stat. Mech. (2013) P03014.

[14] J. J. Hopfield, Neural networks and physical systems with emergent collective computational abilities, Proc. Natl. Acad. Sci. USA **79**, 2554 (1982).

[15] W. A. Little, The existence of persistent states in the brain, Math. Biosci. **19**, 101 (1974).

[16] D. J. Amit, H. Gutfreund, and H. Sompolinsky, Spin-glass models of neural networks, Phys. Rev. A **32**, 1007 (1985).

[17] D. J. Amit, H. Gutfreund, and H. Sompolinsky, Statistical mechanics of neural networks near saturation, Ann. Phys. (NY) **173**, 30 (1987).

[18] G. Montúfar, Restricted Boltzmann machines: Introduction and review, in *Information Geometry and Its Applications: On the Occasion of Shun-ichi Amari's 80th Birthday, IGAIA IV Liblice, Czech Republic, June 2016* (Springer, Berlin, 2018), pp. 75–115.

[19] C. Bény, Deep learning and the renormalization group, arXiv:1301.3124.

[20] P. Mehta and D. J. Schwab, An exact mapping between the variational renormalization group and deep learning, arXiv:1410.3831.

[21] M. D. Zeiler and R. Fergus, Visualizing and understanding convolutional networks, in *Computer Vision–ECCV 2014*, Lecture Notes in Computer Science Vol. 8689 (Springer, Cham, 2014).

[22] H. W. Lin, M. Tegmark, and D. Rolnick, Why does deep and cheap learning work so well? J. Stat. Phys. **168**, 1223 (2017).

[23] S. Iso, S. Shiba, and S. Yokoo, Scale-invariant feature extraction of neural network and renormalization group flow, Phys. Rev. E **97**, 053304 (2018).

[24] E. de Mello Koch, R. de M. Koch, and L. Cheng, Is deep learning a renormalization group flow? IEEE Access **8**, 106487 (2020).

[25] S. S. Funai and D. Giataganas, Thermodynamics and feature extraction by machine learning, Phys. Rev. Res. **2**, 033415 (2020).

[26] F. Gong and Z. Xia, Interpreting deep learning by establishing a rigorous corresponding relationship with renormalization group, arXiv:2212.00005.

[27] D. A. Roberts, S. Yaida, and B. Hanin, *The Principles of Deep Learning Theory* (Cambridge University Press, Cambridge, 2022).

[28] A. Choromanska, M. Henaff, M. Mathieu, G. B. Arous, and Y. LeCun, The loss surfaces of multilayer networks, in *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*, Proceedings of Machine Learning Research Vol. 38 (PMLR, Brookline, MA, USA, 2015), pp. 192–204.

[29] S. S. Schoenholz, J. Gilmer, S. Ganguli, and J. Sohl-Dickstein, Deep information propagation, arXiv:1611.01232.

[30] B. Poole, S. Lahiri, M. Raghu, J. Sohl-Dickstein, and S. Ganguli, Exponential expressivity in deep neural networks through transient chaos, in *Advances in Neural Information Processing Systems (NeurIPS)* (Curran Associates, Inc., Red Hook, NY, USA, 2016), Vol. 29, pp. 3360–3368.

[31] D. Oprisa and P. Toth, Criticality & deep learning II: Momentum renormalisation group, arXiv:1705.11023.

[32] D. Oprisa and P. Toth, Criticality and deep learning I: Theory vs empirics, arXiv:1702.08039.

[33] Z. Liao, R. Couillet, and M. W. Mahoney, A random matrix analysis of random Fourier features: Beyond the Gaussian kernel, a precise phase transition, and the corresponding double descent, in *Advances in Neural Information Processing Systems (NeurIPS)* (Curran Associates, Inc., Red Hook, NY, USA, 2020), Vol. 33, pp.–13939–13950.

[34] J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, and D. Amodei, Scaling laws for neural language models, arXiv:2001.08361.

[35] E. Caballero, K. Gupta, I. Rish, and D. Krueger, Broken neural scaling laws, arXiv:2210.14891.

[36] Y. Bahri, E. Dyer, J. Kaplan, J. Lee, and U. Sharma, Explaining neural scaling laws, Proc. Natl. Acad. Sci. USA **121**, e2311878121 (2024).

[37] B. Sorscher, R. Geirhos, S. Shekhar, S. Ganguli, and A. Morcos, Beyond neural scaling laws: Beating power law scaling via data pruning, in *Advances in Neural Information Processing Systems (NeurIPS)* (Curran Associates, Inc., Red Hook, NY, USA, 2022), Vol. 35, pp. 19523–19536.

[38] T. Hastie, A. Montanari, S. Rosset, and R. J. Tibshirani, Surprises in high-dimensional ridgeless least squares interpolation, Ann. Statist. **50**, 949 (2022).

[39] L. Ziyin and M. Ueda, Exact phase transitions in deep learning, arXiv:2205.12510.

[40] B. Bordelon, A. Atanasov, and C. Pehlevan, A dynamical model of neural scaling laws, arXiv:2402.01092.

[41] Q. Ma, H. Mao, J. Liu, Z. Zhang, C. Feng, Y. Song, Y. Shao, and Y. Ma, Do neural scaling laws exist on graph self-supervised learning? arXiv:2408.11243.

[42] Z. Du, A. Zeng, Y. Dong, and J. Tang, Understanding emergent abilities of language models from the loss perspective, arXiv:2403.15796.

[43] I. Ernst, Beitrag zur Theorie des Ferromagnetismus, Z. Phys. A: Hadrons Nucl. **31**, 253 (1925).

[44] S. Ruder, An overview of gradient descent optimization algorithms, arXiv:1609.04747.

[45] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-Dujaili, Y. Duan, O. Al-Shamma, J. Santamaría, M. A. Fadhel, M. Al-Amidie, and L. Farhan, Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions, J. Big. Data. **8**, 53 (2021).

[46] W. S. McCulloch and W. Pitts, A logical calculus of the ideas immanent in nervous activity, Bull. Math. Biophys. **5**, 115 (1943).

[47] A. Dongare, R. Kharde, A. D. Kachare *et al.*, Introduction to artificial neural network, Int. J. Eng. Innov. Technol. **2**, 189 (2012).

[48] S. Amato, B. Apolloni, G. Caporali, U. Madesani, and A. Zanaboni, Simulated annealing approach in backpropagation, Neurocomputing **3**, 207 (1991).

[49] R. M. Burton and G. J. Mpitsos, Event-dependent control of noise enhances learning in neural networks, Neural Netw. **5**, 627 (1992).

[50] M. Welling and Y. Teh, Bayesian learning via stochastic gradient Langevin dynamics, in *Proceedings of the 28th International Conference on Machine Learning, ICML 2011* (ACM, New York, 2011), p. 681.

[51] T. Chen, E. Fox, and C. Guestrin, Stochastic gradient Hamiltonian Monte Carlo, in *International Conference on Machine Learning*, Proceedings of Machine Learning Research Vol. 32 (PMLR, Brookline, MA, USA, 2014), pp. 1683–1691.

[52] C. Li, C. Chen, D. Carlson, and L. Carin, Preconditioned stochastic gradient Langevin dynamics for deep neural networks, in *Proceedings of the AAAI Conference on Artificial Intelligence* (AAAI, Washington, 2016), Vol. 30.

[53] H. Risken, Langevin equations, *The Fokker-Planck Equation: Methods of Solution and Applications* (Springer, Berlin, 1996), pp. 32–62.

[54] Q. Li and H. Sompolinsky, Statistical mechanics of deep linear neural networks: The backpropagating kernel renormalization, Phys. Rev. X **11**, 031059 (2021).

[55] G. A. Pavliotis, The Langevin equation, *Stochastic Processes and Applications: Diffusion Processes, the Fokker-Planck and Langevin Equations* (Springer, New York, 2014), pp. 181–233.

[56] G. E. Hinton and R. R. Salakhutdinov, Reducing the dimensionality of data with neural networks, Science **313**, 504 (2006).

[57] A. Krizhevsky, I. Sutskever, and G. E. Hinton, Imagenet classification with deep convolutional neural networks, in *Advances in Neural Information Processing Systems (NeurIPS)* (Curran Associates, Inc., Red Hook, NY, USA, 2012), Vol. 25, pp. 1097–1105.

[58] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath *et al.*, Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups, IEEE Signal Process. Mag. **29**, 82 (2012).

[59] D. Ciregan, U. Meier, and J. Schmidhuber, Multi-column deep neural networks for image classification, in *2012 IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, Piscataway, NJ, 2012), pp. 3642–3649.

[60] S. Sun, W. Chen, L. Wang, X. Liu, and T.-Y. Liu, On the depth of deep neural networks: A theoretical view, in *Proceedings of the AAAI Conference on Artificial Intelligence* (AAAI, Washington, 2016), Vol. 30.

[61] L. D. Faddeev and V. N. Popov, Feynman diagrams for the Yang-Mills field, Phys. Lett. B **25**, 29 (1967).

[62] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, Gradient-based learning applied to document recognition, Proc. IEEE **86**, 2278 (1998).

[63] Z. Lu, H. Pu, F. Wang, Z. Hu, and L. Wang, The expressive power of neural networks: A view from the width, in *Advances in Neural Information Processing Systems (NeurIPS)* (Curran Associates, Inc., Red Hook, NY, USA, 2017), Vol. 30, pp. 6231–6239.

[64] B. Hanin and M. Sellke, Approximating continuous functions by ReLU nets of minimal width, arXiv:1710.11278.

[65] J. Johnson, Deep, skinny neural networks are not universal approximators, arXiv:1810.00393.

[66] P. Kidger and T. Lyons, Universal approximation with deep narrow networks, in *Proceedings of Thirty Third Conference on Learning Theory*, Proceedings of Machine Learning Research Vol. 125 (PMLR, Brookline, MA, USA, 2020), pp. 2306–2327.

[67] S. Park, C. Yun, J. Lee, and J. Shin, Minimum width for universal approximation, arXiv:2006.08859.

[68] S. Sharma, S. Sharma, and A. Athaiya, Activation functions in neural networks, Int. J. Eng. Appl. Sci. Technol. **4**, 310 (2020).

[69] H. Wang, S. Ma, L. Dong, S. Huang, H. Wang, L. Ma, F. Yang, R. Wang, Y. Wu, and F. Wei, BitNet: Scaling 1-bit transformers for large language models, arXiv:2310.11453.

[70] S. Ma, H. Wang, L. Ma, L. Wang, W. Wang, S. Huang, L. Dong, R. Wang, J. Xue, and F. Wei, The era of 1-bit LLMs: All large language models are in 1.58 bits, arXiv:2402.17764.

[71] B. Hu, Introduction to real-space renormalization-group methods in critical and chaotic phenomena, Phys. Rep. **91**, 233 (1982).

[72] J. Cardy, *Scaling and Renormalization in Statistical Physics*, Cambridge Lecture Notes in Physics Vol. 5 (Cambridge University Press, Cambridge, 1996).

[73] D. Lee, H.-S. Lee, and J. Yi (unpublished).

[74] S. F. Edwards and P. W. Anderson, Theory of spin glasses, J. Phys. F **5**, 965 (1975).

[75] H. Nishimori, *Statistical Physics of Spin Glasses and Information Processing: An Introduction*, Mathematical and Statistical Physics Vol. 111 (Clarendon, Oxford, 2001).