Bridging Many-Body Quantum Physics and Deep Learning via Tensor Networks

Yoav Levine,[1,*] Or Sharir,[1,†] Nadav Cohen,[2,‡] and Amnon Shashua[1,§]

[1] *The Hebrew University of Jerusalem, Israel*

[2] *School of Mathematics, Institute for Advanced Study, Princeton, NJ, USA*

The harnessing of modern computational abilities for many-body wave-function representations is naturally placed as a prominent avenue in contemporary condensed matter physics. Specifically, highly expressive computational schemes that are able to efficiently represent the entanglement properties which characterize many-particle quantum systems are of interest. In the seemingly unrelated field of machine learning, deep network architectures have exhibited an unprecedented ability to tractably encompass the convoluted dependencies which characterize hard learning tasks such as image classification or speech recognition. However, theory is still lagging behind these rapid empirical advancements, and key questions regarding deep learning architecture design have no adequate theoretical answers. In this paper, we establish a Tensor Network (TN) based common language between the two disciplines, which allows us to offer bidirectional contributions. By showing that many-body wave-functions are structurally equivalent to mappings of convolutional and recurrent networks, we construct their TN descriptions in the form of Tree and Matrix Product State TNs, respectively, and bring forth quantum entanglement measures as natural quantifiers of dependencies modeled by such networks. Accordingly, we propose a novel entanglement based deep learning design scheme that sheds light on the success of popular architectural choices made by deep learning practitioners, and suggests new practical prescriptions. In the other direction, we identify that an inherent re-use of information in state-of-the-art deep learning architectures is a key trait that distinguishes them from TN based representations. Therefore, we suggest a new TN manifestation of information re-use, which enables TN constructs of powerful architectures such as deep recurrent networks and overlapping convolutional networks. This allows us to theoretically demonstrate that the entanglement scaling supported by state-of-the-art deep learning architectures can surpass that of commonly used expressive TNs such as the Multiscale Entanglement Renormalization Ansatz in one dimension, and can support volume law entanglement scaling in two dimensions with an amount of parameters that is a square root of that required by Restricted Boltzmann Machines. We thus provide theoretical motivation to shift trending neural-network based wave-function representations closer to state-of-the-art deep learning architectures.

Introduction.– Many-body physics and machine learning are distinct scientific disciplines, however they share a common need for efficient representations of highly expressive multivariate function classes. In the former, the function class of interest captures the entanglement properties of an examined many-body quantum system, and in the latter, it describes the dependencies required for performing a modern machine learning task.

In the physics domain, a prominent approach for classically simulating many-body wave-functions makes use of their entanglement properties in order to construct Tensor Network (TN) architectures that aptly model them in the thermodynamic limit. For example, in one dimension (1D), systems that obey area law entanglement scaling [1] are efficiently represented by a Matrix Product State (MPS) TN [2, 3], while systems with logarithmic corrections to this entanglement scaling are efficiently represented by a Multiscale Entanglement Renormalization Ansatz (MERA) TN [4]. An ongoing development of TN architectures [5–9] is intended to meet the need for function classes that are expressive enough to model systems of interest, yet are still susceptible to the rich algorithmic toolbox that accompanies TNs [10–17].

In the machine learning domain, deep network architectures have enabled unprecedented results in recent years [18–25], due to their ability to successfully capture intricate dependencies in complex data sets. However, despite their popularity in science and industry, formal understanding of these architectures is limited. Specifically, the question of why the multivariate function families induced by common deep learning architectures successfully capture the dependencies brought forth by challenging machine learning tasks, is largely open.

TN applications in machine learning include optimizations of an MPS to preform learning tasks [26, 27] and unsupervised preprocessing of the data set via tree TNs [28]. Additionally, a theoretical mapping between Restricted Boltzmann Machines (RBMs) and TNs was proposed [29]. Inspired by recent achievements in machine learning, wave function-representations based on fully-connected neural networks and RBMs, which represent relatively veteran deep learning constructs, have recently been suggested [30–35]. Consequently, RBMs were shown to support volume law entanglement scaling with an amount of parameters that is quadratic in the linear dimension of the represented system (its characteristic size along one dimension) in 2D [36], versus a quartic dependence required for volume law scaling in 2D fully-connected networks.

In this paper, we propose a TN based approach for describing deep learning architectures which are at the forefront of recent empirical successes. Thus, we estab-

lish a bridge that facilitates an interdisciplinary transfer of results and tools, and allows us to address above presented needs of both fields. First, we import concepts from quantum physics that enable us obtain new results in the rapidly evolving field of deep learning theory. In the opposite direction, we attain TN manifestations of provably powerful deep learning principles, which help us establish the benefits of employing such principles for the representation of highly entangled wave-functions.

We begin by identifying an equivalence between the tensor based form of a many-body wave-function and the function realized by Convolutional Arithmetic Circuits (ConvACs) [37–39] and single-layered Recurrent Arithmetic Circuits (RACs) [40, 41]. These are representatives of two prominent deep learning architecture classes: convolutional networks, commonly operating over spatial inputs, used for tasks such as image classification [18]; and recurrent networks, commonly operating over temporal inputs, used for tasks such as speech recognition [23]. Given the above equivalence, we construct a Tree TN representation of the ConvAC and an MPS representation of the RAC (Fig. 1), and show how entanglement measures [42] naturally quantify the ability of the multivariate function realized by a deep network to model dependencies. Consequently, we are able to demonstrate how the common practice of entanglement based TN architecture selection, can be readily converted into a methodological approach for matching the architecture of a deep network to a given task. Specifically, our construction allows translation of recently derived bounds on the maximal entanglement represented by an arbitrary TN [43] into machine learning terms. We thus obtain novel quantum physics inspired practical guidelines for task-tailored architecture design of deep convolutional networks.

The above analysis highlights a key principle separating powerful deep learning architectures from common TN based representations, namely, the re-use of information. Specifically, in the ConvAC, which is shown to be described by a Tree TN, the convolutional windows have 1×1 receptive fields and therefore do not overlap when slid across the feature maps. In contrast, state-of-the-art convolutional networks involve larger convolution kernels (*e.g.* 3×3), which therefore overlap during the calculation of the convolution [18, 19]. Such overlapping architectures inherently involve re-use of information, since the same activation is used for the calculation of several adjacent activations in the subsequent layer. Similarly, unlike the shallow recurrent network, shown to be described by an MPS TN, state-of-the-art deep recurrent networks inherently involve information re-use [23].

Relying on the above observation, we propose a method for representing information re-use within the standard TN framework. Accordingly, we are able to present new TN constructs that correspond to deep recurrent and overlapping convolutional architectures (Figs. 2 and 3). We thus obtain the tools to directly compare the entanglement scaling of powerful deep networks to that of traditionally used TNs. Specifically, we prove that while the MERA TN efficiently supports logarithmic corrections to the area law entanglement scaling with sub-system size in 1D, deep recurrent networks support super-logarithmic corrections to the area law in 1D, and overlapping convolutional networks support volume law entanglement scaling in 1D and 2D up to system sizes smaller than a threshold related to the amount of overlaps. Our analysis shows that the amount of parameters required for supporting volume entanglement scaling is linear in the linear dimension of the represented system in 1D and in 2D. Therefore, in comparison with previous neural network based wave-function representations, which allow tractable access to 2D systems of sizes unattainable by current TN based approaches, we demonstrate that overlapping convolutional networks are polynomially more efficient in representing volume law entanglement scaling in 2D. Thus, we establish formal advantages of employing state-of-the-art deep learning principles for many-body wave-function representation, and suggest a practical framework for implementing and investigating these architectures within the standard TN platform.

Quantum wave-functions and deep learning architectures.– We show below a structural equivalence between a many-body wave-function and the function which a deep learning architecture implements over its inputs. The convolutional and recurrent networks described below have been analyzed to date via tensor decompositions [44, 45], which are compact high-order tensor representations based on linear combinations of *outer-products* between low-order tensors. The presented equivalence to wave-functions, suggests the slightly different algebraic approach manifested by TNs – a compact representation of a high-order tensor through contractions (or *inner-products*) among lower-order tensors. Accordingly, we provide TN constructions of the examined deep learning architectures.

We consider a convolutional network referred to as a Convolutional Arithmetic Circuit (ConvAC) [37–39]. This deep convolutional network operates similarly to ConvNets typically employed in practice [18, 19], only with linear activations and product decimation instead of more common non-linear activations and max-out decimation, see Fig. 1a. Proof methodologies related to ConvACs have been extended to common ConvNets [38], and from an empirical perspective ConvACs work well in many practical settings [46, 47]. Analogously, we examine a class of recurrent networks referred to as Recurrent Arithmetic Circuits (RACs) [40, 41]. RACs share the architectural features of standard recurrent networks, where information from previous time-steps is mixed with current incoming data via the Multiplicative Integration operation [22, 48] (see Fig. 1b). It has been experimentally demonstrated that conclusions attained by analyses

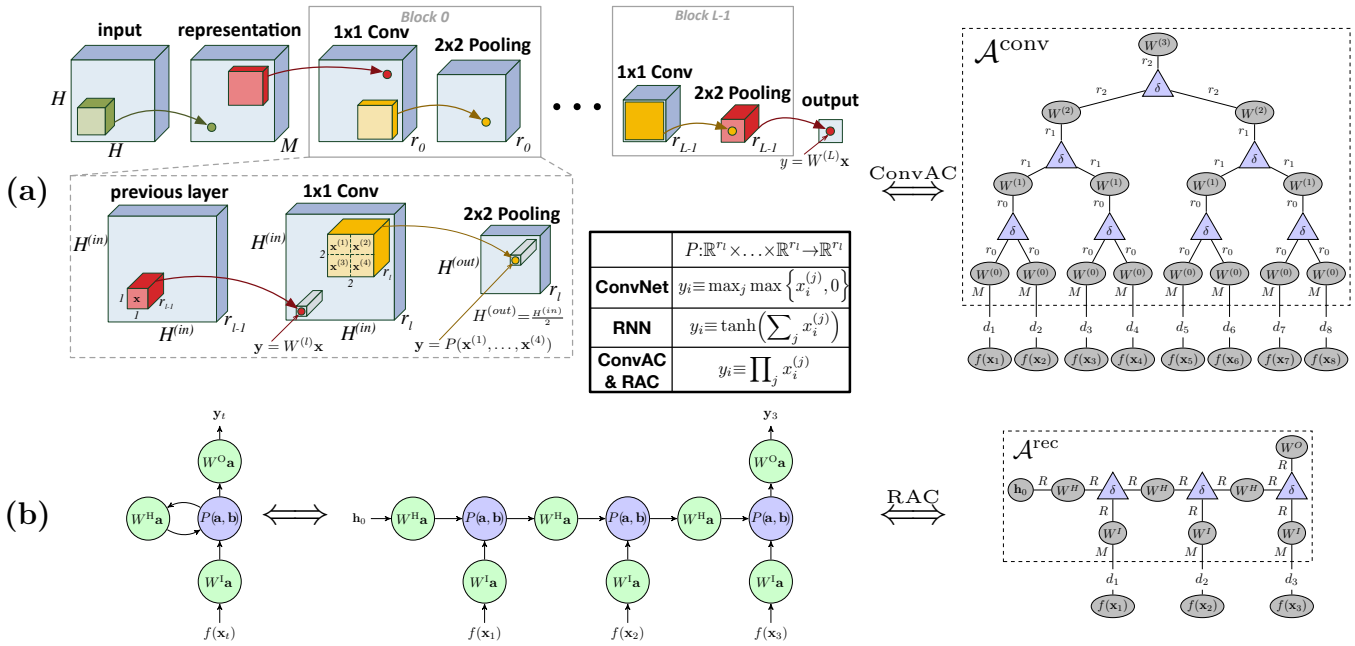

FIG. 1. **(a)** The ConvAC [37], which has convolution kernels of size 1×1, linear activations and product pooling operations, is equivalent to a Tree TN (presented for the case of $N = 8$ in its 1D form for clarity). The triangular δ-tensors impose the 'same channel pooling' trait of the ConvAC. The matrices in the Tree TN host the ConvAC's convolutional weights, and the bond dimensions at each tree level $l \in [L]$ are equal to the number of channels in the corresponding layer of the ConvAC, also referred to as its width, denoted r_l. This fact allows us to translate a min-cut result on the TN into practical conclusions regarding pooling geometry and layer widths in convolutional networks. **(b)** The shallow RAC [40], which merges the hidden state of the previous time-step with new incoming data via the Multiplicative Integration operation, is equivalent to an MPS TN (presented for the case of $N = 3$). The matrices W^{I}, W^{H}, and W^{O}, are the RAC's input, hidden, and output weight matrices, respectively. The δ-tensors correspond the Multiplicative Integration. In Eq. (1), we consider the output of the RAC after N time-steps.

of the above arithmetic circuits extend to commonly used networks [39, 41, 49, 50]. In the convolutional case, we consider tasks over images in which the network is given an N-pixel input image, $X = (\mathbf{x}_1, ..., \mathbf{x}_N)$. In the recurrent case, we focus on a task in which the network is given a sequential input $\{\mathbf{x}_t\}_{t=1}^N$. The output of a ConvAC/single layered RAC obeys:

$$y(\mathbf{x}_1, .., \mathbf{x}_N) = \sum_{d_1,..,d_N=1}^{M} \mathcal{A}^{\mathrm{conv/rec}}_{d_1..d_N} \prod_{j=1}^{N} f_{d_j}(\mathbf{x}_j), \qquad (1)$$

where $\{f_d\}_{d=1}^M$ are linearly independent representation functions, which form an initial mapping of each input $\mathbf{x}_j$ to a vector $(f_1(\mathbf{x}_j), ..., f_M(\mathbf{x}_j)) \in \mathbb{R}^M$. The tensors $\mathcal{A}^{\mathrm{conv}}$ and $\mathcal{A}^{\mathrm{rec}}$ that define the computation of the ConvAC and RAC, have been analyzed to date via the Hierarchical Tucker [51] and the Tensor Train [52] decompositions, respectively. Their entries are polynomials in the appropriate network's convolutional weights [37] or recurrent weights [40, 41], and their N indices respectively correspond to the N spatial or temporal inputs.

Considering N-particle quantum states with a local Hilbert space $\mathcal{H}$ of dimension M, Eq. (1) is equivalent to the inner-product:

$$y(\mathbf{x}_1, .., \mathbf{x}_N) = \left\langle \psi^{\ \mathrm{ps}}(\mathbf{x}_1, .., \mathbf{x}_N) | \psi^{\ \mathrm{conv/rec}} \right\rangle, \qquad (2)$$

where $\left|\psi^{\ \mathrm{conv/rec}}\right\rangle = \sum_{d_1,..,d_N=1}^{M} \mathcal{A}^{\mathrm{conv/rec}}_{d_1..d_N} \left|\hat{\psi}_{d_1..d_N}\right\rangle$, and $\left|\psi^{\ \mathrm{ps}}\right\rangle = \sum_{d_1,..,d_N=1}^{M} \prod_{j=1}^{N} f_{d_j}(\mathbf{x}_j) \left|\hat{\psi}_{d_1..d_N}\right\rangle$ is a product state, for $\left|\hat{\psi}_{d_1..d_N}\right\rangle = \left|\hat{\psi}_{d_1}\right\rangle \otimes \cdots \otimes \left|\hat{\psi}_{d_N}\right\rangle$, where $\left\{\left|\hat{\psi}_d\right\rangle\right\}_{d=1}^M$ is some orthonormal basis of $\mathcal{H}$. In this structural equivalence, the N-inputs to the deep learning architecture (*e.g.* pixels in an input image or syllables in an input sentence) are analogous to the N-particles. Since the product state can be associated with some local preprocessing of the inputs, all the information regarding dependencies between input elements that the network is able to model is effectively encapsulated in $\mathcal{A}^{\mathrm{conv/rec}}$, which by definition also holds the entanglement structure of the state $\left|\psi^{\ \mathrm{conv/rec}}\right\rangle$.

Therefore, the TN form of the weights tensor is of interest. In Fig. 1a, we present the TN corresponding to $\mathcal{A}^{\mathrm{conv}}$. The depth of this Tree TN is equal to the depth of the convolutional network, L, and the circular 2-legged tensors represent matrices holding the convolu-

tional weights of each layer $l \in [L] := \{1, ..., L\}$, denoted $W^{(l)}$. Accordingly, the bond dimension of the TN edges comprising each tree level $l \in [L]$ is equal to the number of channels in the corresponding layer of the convolutional network, r_l, referred to as the layer's width. The 3-legged triangles in the Tree TN represent δ_{ijk} tensors (equal to 1 if $i = j = k$ and 0 otherwise), which correspond to the decimation procedure, referred to as pooling in deep learning nomenclature. In Fig. 1b we present the TN corresponding to $\mathcal{A}^{\text{rec}}$. In this MPS shaped TN, the circular 2-legged tensors represent matrices holding the input, hidden, and output weights, respectively denoted W^{I}, W^{H}, and W^{O}. The δ-tensors correspond to the Multiplicative Integration trait of the RAC.

Deep learning architecture design via entanglement measures.– The structural connection between many-body wave-functions and functions realized by convolutional and recurrent networks, creates an opportunity to employ well-established tools and insights from many-body physics for the design of these deep learning architectures. We begin this section by showing that quantum entanglement measures extend previously used means for quantifying dependencies modeled by deep learning architectures. Then, inspired by common condensed matter physics practice, we propose a novel methodology for principled deep network design.

In [39], the algebraic notion of separation-rank is used as a tool for measuring dependencies modeled between two disjoint parts of a deep convolutional network's input. Let (A, B) be a partition of $[N]$. The separation-rank of $y(\mathbf{x}_1, .., \mathbf{x}_N)$ w.r.t. (A, B) is defined as the minimal number of multiplicatively separable [w.r.t. (A, B)] summands that together give y. For example, if y is separable w.r.t. (A, B), then its separation rank is 1 and it models no dependency between the inputs of A and those of B [53]. The higher the separation rank, the stronger the dependency modeled between sides of the partition [54]. Remarkably, due to the equivalence in Eq. (2), the separation rank of $y(\mathbf{x}_1, ..., \mathbf{x}_N)$ w.r.t. a partition (A, B) is equal to the Schmidt entanglement measure of $\left|\psi^{\text{conv/rec}}\right\rangle$ w.r.t. (A, B) [55]. The logarithm of the Schmidt number upper bounds the state's entanglement entropy w.r.t. (A, B).

The analysis of separation ranks, now extended to entanglement measures, brings forth a principle for designing a deep learning architecture intended to perform a task specified by certain dependencies – the network should be designed such that these dependencies can be modeled, *i.e.* partitions that split dependent regions should have high entanglement entropy. For example, for tasks over symmetric face images, a convolutional network should support high entanglement entropy w.r.t. the left-right partition in which A and B hold the left and right halves of the image. Conversely, for tasks over natural images with high dependency between adjacent pixels, the interleaved partition for which A and B hold the odd and even input pixels, should be given higher entanglement entropy (partitions of interest in this machine learning context are not necessarily contiguous). As for recurrent networks, they should be able to integrate data from different time steps, and specifically to model long-term dependencies between the beginning and ending of the input sequence. Therefore, the recurrent network should ideally support high entanglement w.r.t. the partition separating the first inputs from the latter ones.

Given the TN constructions in Fig. 1, we translate a known result on the quantitative connection between quantum entanglement and TNs [43] into bounds on the dependencies supported by the above deep learning architectures:

Theorem 1 *(proof in [49]) Let y be the function computed by a ConvAC/RAC [Eq. (1)] with $\mathcal{A}^{conv/rec}$ represented by the TNs in Fig. 1. Let (A, B) be any partition of $[N]$. Assume that the channel numbers (equivalently, bond dimensions) are all powers of the same integer [56]. Then, the maximal entanglement entropy w.r.t. (A, B) supported by $\mathcal{A}^{conv/rec}$ is equal to the minimal cut separating A from B in the respective TN, where the weight of each edge is the log of its bond dimension.*

Theorem 1 leads to practical implications when there is prior knowledge regarding the task at hand. If one wishes to construct a deep learning architecture that is expressive enough to model intricate dependencies according to some partition (A, B), it is advisable to design the network such that all the cuts separating A from B in the corresponding TN have high weights. Two practical principles for deep convolutional network design emerge. Firstly, as established in [39] by using a separation rank based approach, pooling windows should combine dependent inputs earlier along the network calculation. This explains the success of commonly employed networks with contiguous pooling windows – they are able to model short-range dependencies in natural data sets. A new prescription obtained from the above theorem [49] advises that layer widths are to be set according to spatial scale of dependencies – deeper layers should be wide in order to model long-range input dependencies (present *e.g.* in face images), and early layers should be wide in order to model short-range dependencies (present *e.g.* in natural images). Thus, inspired by entanglement based TN architecture selection for many-body wave-functions, practical insights regarding deep learning architecture design can be attained.

Power of deep learning for wave-function representations.– In this section, we consider successful extensions to the above presented architectures, in the form of deep recurrent networks and overlapping convolutional networks. These extensions, presented below (Figs. 2 and 3), are seemingly 'innocent' - they

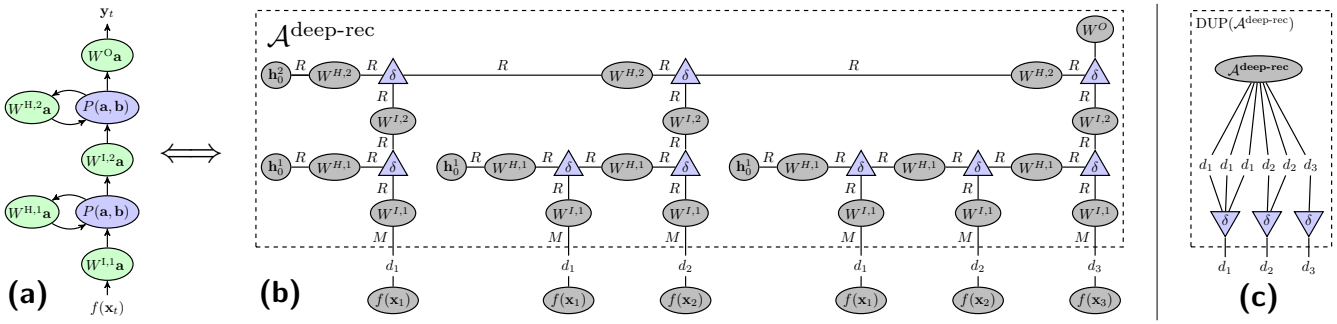

FIG. 2. **(a)** A deep recurrent network is represented by a concise and tractable computation graph, which employs information re-use (two edges emanating out of a single node). **(b)** In TN language, deep RACs [40] are represented by a recursive TN structure (presented for the case of $N = 3$), which makes use of input duplication to circumvent an inherent inability of TNs to model information re-use (see supplementary material for formalization of this argument). This novel tractable extension to an MPS TN supports entanglement scaling that surpasses that of a MERA TN in 1D (theorem 2). **(c)** Given the high-order tensor $\mathcal{A}^{\text{deep-rec}}$ with duplicated external indices [presented in (b)], the process of obtaining a dup-tensor $DUP(\mathcal{A}^{\text{deep-rec}})$ that corresponds to $|\psi^{\text{deep-rec}}\rangle$ [Eq. (4)], involves a single δ-tensor per unique external index.

introduce a linear growth in the amount of parameters and their computation remains tractable. Despite this fact, both are empirically known to enhance performance [18, 23], and have been theoretically shown to introduce an exponential boost in network expressivity [40, 50]. As demonstrated below, both deep recurrent and overlapping convolutional architectures inherently involve information re-use, where a single activation is duplicated and sent to several subsequent calculations along the network computation. We pinpoint this re-use of information along the network as a key element differentiating powerful deep learning architectures from standard TN representations. Though data duplication is generally unachievable in the language of TNs, we are able to circumvent this restriction and construct TN equivalents of the above networks, which may be viewed as deep learning inspired enhancements of MPS and Tree TNs. We are thus able to translate super-polynomial expressivity results on the above networks into super-area-law lower bounds on the entanglement scaling that can be supported by them. Our results indicate these successful deep learning architectures as natural candidates for joining the recent effort of neural network based wave function representations, currently focused mainly on RBMs. As a by-product, our construction suggests new TN mechanisms, which enjoy an equivalence to tractable deep learning computation schemes and surpass traditionally used expressive TNs in representable entanglement scaling.

Beginning with recurrent networks, an architectural choice that has been empirically shown to yield enhanced performance in sequential tasks [23, 57] and recently proven to bring forth a super-polynomial advantage in network long-term memory capacity [40], involves adding more layers, *i.e.* deepening (see Fig. 2a). The construction of a TN which matches the calculation of a deep RAC is less trivial than that of the shallow case, since the output vector of each layer at every time-step is re-used and sent to two different calculations - as an input of the next layer up and as a hidden vector for the next time-step. This operation of duplicating data, which is simply achieved in any practical setting, is actually impossible to represent in the framework of TNs (see claim 1 in the supplementary material). However, the form of a TN representing the deep recurrent network may be attained by a simple 'trick' – duplication of the input data itself, such that each instance of a duplicated intermediate vector is generated by a separate TN branch. This technique yields the 'recursive-MPS' TN construction of deep recurrent networks, depicted in Fig. 2b.

It is noteworthy that due to these external duplications, the tensor represented by a deep RAC TN, denoted $\mathcal{A}^{\text{deep-rec}}$, does not immediately correspond to an N-particle wave-function, as the TN has more than N external edges. However, when considering the operation of the deep recurrent network over inputs comprised solely of standard basis vectors, $\{\hat{e}^{(i_j)}\}_{j=1}^{N}$ ($i_j \in [M]$), with identity representation functions [58], we may write the function realized by the network in a form analogous to that of Eq. (2):

$$y\left(\hat{e}^{(i_1)},..,\hat{e}^{(i_N)}\right) = \left\langle \psi^{\text{ps}}\left(\hat{e}^{(i_1)},..,\hat{e}^{(i_N)}\right)|\psi^{\text{deep-rec}}\right\rangle, \tag{3}$$

where in this case the product state simply upholds $\left|\psi^{\text{ps}}\left(\hat{e}^{(i_1)},..,\hat{e}^{(i_N)}\right)\right\rangle = \left|\hat{\psi}_{i_1}\right\rangle \otimes \cdots \otimes \left|\hat{\psi}_{i_N}\right\rangle$, *i.e.* some orthonormal basis element of $\mathcal{H}^{\otimes N}$, and:

$$|\psi^{\text{deep-rec}}\rangle := \sum_{d_1,..,d_N=1}^{M} DUP(\mathcal{A}^{\text{deep-rec}})_{d_1..d_N}\left|\hat{\psi}_{d_1..d_N}\right\rangle, \tag{4}$$

where $DUP(\mathcal{A}^{\text{deep-rec}})$ is the N-indexed sub-tensor of

$\mathcal{A}^{\text{deep-rec}}$ holding its values when duplicated external indices are equal, referred to as the dup-tensor. Fig. 2c shows the TN calculation of $DUP(\mathcal{A}^{\text{deep-rec}})$. Effectively, since Eqs. (3) and (4) dictate: $y\left(\hat{e}^{(i_1)},..,\hat{e}^{(i_N)}\right) = DUP(\mathcal{A}^{\text{deep-rec}})_{i_1..i_N}$, under the above conditions the deep recurrent network represents the N-particle wave function $\left|\psi^{\text{deep-rec}}\right\rangle$. In the following theorem, we show that the maximum entanglement entropy of a state $\left|\psi^{\text{deep-rec}}\right\rangle$ modeled by such a deep recurrent network increases super-logarithmically with sub-system size:

Theorem 2 *(proof in supplementary material) Let y be the function computing the output after N time-steps of an RAC with 2 layers and R hidden channels per layer (Fig. 2a), with 'one-hot' inputs and identity representation functions [Eq. (3)]. Let $\mathcal{A}^{deep\text{-}rec}$ be the tensor represented by the TN corresponding to y (Fig. 2b) and* DUP$(\mathcal{A}^{deep\text{-}rec})$ *the matching N-indexed* dup-tensor *(Fig. 2c). Let (A,B) be a partition of $[N]$ such that $|A| \leq |B|$ and $B = \{1,\ldots,|B|\}$ [59]. Then, the maximal entanglement entropy w.r.t. (A,B) supported by* DUP$(\mathcal{A}^{deep\text{-}rec})$ *is lower-bounded by:*

$$log\left\{\binom{\min\{R,M\}+|A|-1}{|A|}\right\} = log\left\{super\text{-}poly(|A|)\right\}.$$

The above theorem brings about noteworthy consequences when comparing the entanglement scaling representable by a deep RAC with that of prevalent TNs. Specifically, in order to model wave-functions with super-area-law entanglement scaling in 1D, which cannot be efficiently represented by an MPS TN (*e.g.* ground states of critical systems), a common approach is to employ the MERA TN (Fig. 4). In a MERA representing a 1D system, the maximum entanglement entropy of a representable sub-system A scales no more than linearly with the log of its size: $\mathcal{O}(log\,|A|)$. This can be viewed as a direct corollary of the min-cut result in [43]. In comparison with this logarithmic correction to the area law entanglement scaling in 1D, the bound in theorem 2 may be referred to as a super-logarithmic correction to such entanglement scaling. Notably, while the result in theorem 2 is attained for depth $L = 2$ recurrent networks, empirical evidence suggests that deeper recurrent networks have a better ability to model intricate dependencies [57], so are expected to support even higher entanglement scaling.

Thus, we establish a clear advantage in representable entanglement scaling of the deep recurrent network's tractable computation graph, presented in Fig. 2a, over that of the MERA TN in 1D. This motivates inclusion of deep recurrent networks in the recent effort to achieve wave function representations in neural networks, such as performed by *e.g.* [30] with RBMs. Further development of the 'recursive MPS' TN that aided us in this construction may be of interest. Upon optimization of the deep recurrent network's weights, the above equivalence allows an efficient casting of these weights into the recursive TN. However, the resultant TN would not be normalized, and additional tools are required for any operation of interest to be performed on this TN.

Moving to convolutional networks, state-of-the-art architectures make use of convolution kernels of size $K \times K$, where $K > 1$ [18, 19]. This architectural trait, which implies that the kernels overlap when slid across the feature maps during computation, was shown to yield an exponential enhancement in network expressivity [50]. An example of such an architecture is the overlapping ConvAC depicted in Fig. 3a. It is important to notice that the overlap in convolution kernels automatically results in information re-use, since a single activation is being used for computing several neighboring activations in the following layer. As above, such re-use of information poses a challenge on the straight-forward TN description of overlapping convolutional networks. We employ a similar input duplication technique as that used for deep recurrent networks, in which all instances of a duplicated vector are generated in separate TN branches. This results in the complex looking TN representing the computation of the overlapping ConvAC, presented in Fig. 3b (in 1D form, with $j \in [4]$ representing d_j, for legibility).

Comparing the ConvAC TN in Fig. 3 and the MERA TN in Fig. 4, it is evident that the many-body physics and deep learning communities have effectively elected competing mechanisms in order to enhance the naive decimation/coarse graining scheme represented by a Tree TN. The MERA TN introduces loops via the disentangling operations, which entail intractability yet facilitate operations of interest such as efficient computation of expectation values of local operators. In contrast, overlapping convolutional networks employ information re-use, resulting in a compact and tractable computation, however, as in the aforementioned case of deep recurrent networks, they currently lack an algorithmic toolbox that allows extraction of a represented state's properties of interest.

Due to the inputs duplication, the tensor represented by the TN of the overlapping ConvAC of depth L with kernel size K^d in d spatial dimensions, denoted $\mathcal{A}^{K,L,d}$, has more than N external edges and does not correspond to an N-particle wave-function. When considering the operation of the overlapping ConvAC over standard basis inputs, $\{\hat{e}^{(i_j)}\}_{j=1}^{N}$, with identity representation functions, we attain a form similar to the above:

$$y\left(\hat{e}^{(i_1)},..,\hat{e}^{(i_N)}\right) = \left\langle \psi^{\text{ ps}}\left(\hat{e}^{(i_1)},..,\hat{e}^{(i_N)}\right) | \psi^{\text{ overlap-conv}}\right\rangle, \tag{5}$$

where the product state is a basis element of $\mathcal{H}^{\otimes N}$ as in

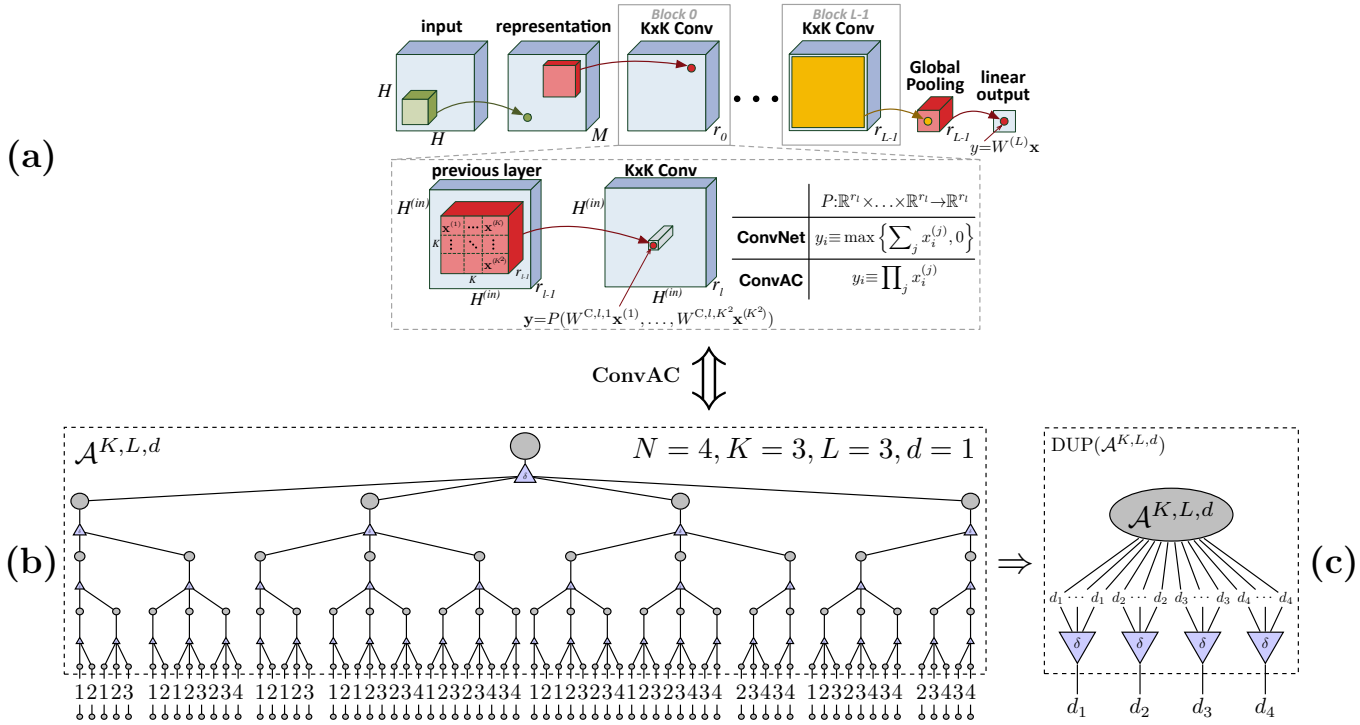

FIG. 3. **(a)** A deep overlapping convolutional network, in which the convolution kernels are of size $K \times K$, with $K > 1$. This results in information re-use as each activation in layer $l \in [L]$ is part of the calculations of several adjacent activations in layer $l+1$. **(b)** The TN corresponding to the calculation of the overlapping ConvAC [50] in the 1D case when the convolution kernel size is $K = 3$, the network depth is $L = 3$, and its spatial extent is $N = 4$. Similarly to the case of deep recurrent networks, the inherent re-use of information in the overlapping convolutional network results in duplication of external indices and a recursive TN structure. This novel tractable extension to a Tree TN supports volume law entanglement scaling until the linear dimension of the represented system exceeds the order of $K \cdot L$ (theorem 3). For legibility, $j \in [4]$ stands for d_j in this figure. **(c)** The TN representing the calculation of the dup-tensor $DUP(\mathcal{A}^{K,L,d})$, which corresponds to $|\psi^{\text{overlap-conv}}\rangle$ [Eq. (6)].

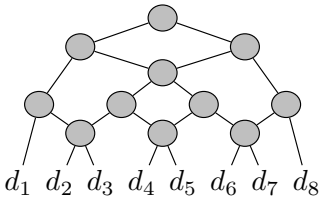

FIG. 4. A 1D MERA TN with open boundary conditions for the case of $N = 8$. The loops in this TN render it intractable for large system sizes.

Eq. (3), and:

$$|\psi^{\text{overlap-conv}}\rangle := \sum_{d_1,..,d_N=1}^{M} DUP(\mathcal{A}^{K,L,d})_{d_1..d_N} \left|\hat{\psi}_{d_1..d_N}\right\rangle . \tag{6}$$

The dup-tensor $DUP(\mathcal{A}^{K,L,d})$ is the N-indexed sub-tensor of $\mathcal{A}^{K,L,d}$ holding its values when duplicated external indices are equal (see its TN calculation in Fig. 3c). Eqs. (5) and (6) imply that under the above conditions, the overlapping ConvAC represents the dup-tensor, which corresponds to the N-particle wave function $|\psi^{\text{overlap-conv}}\rangle$. Relying on results regarding expressiveness of overlapping convolutional architectures [50], we examine the entanglement scaling of a state $|\psi^{\text{overlap-conv}}\rangle$ modeled by such a network:

Theorem 3 *(proof in supplementary material) Let y be the function computing the output of the depth L, d-dimensional overlapping ConvAC with convolution kernels of size K^d (Fig. 3a) for $d = 1, 2$, with 'one-hot' inputs and identity representation functions [Eq. (5)]. Let $\mathcal{A}^{K,L,d}$ be the tensor represented by the TN corresponding to y (Fig. 3b) and* DUP($\mathcal{A}^{K,L,d}$) *the matching N-indexed* dup-tensor *(Fig. 3c). Let (A, B) be a partition of $[N]$ such that A is of size A_{lin} in $d = 1$ and $A_{lin} \times A_{lin}$ in $d = 2$ (A_{lin} is the linear dimension of the sub-system), with $|A| \leq |B|$. Then, the maximal entanglement entropy w.r.t. (A, B) modeled by* DUP($\mathcal{A}^{K,L,d}$) *obeys:*

$$\Omega\left(\min\left\{(A_{lin})^d, KL\,(A_{lin})^{d-1}\right\}\right).$$

Thus, for system sizes limited by the network's architectural parameters (amount of overlapping of convolution kernels and network depth), the tractable overlapping convolutional network in Fig. 3a supports volume law entanglement scaling, while for larger systems

it supports area law entanglement scaling. In 2D, the tractability of the convolutional network gives it an advantage over alternative numerical approaches which are limited to systems of linear size A_{lin} of a few dozens [60–65], (standard overlapping convolutional networks can reach A_{lin} of a few hundreds [18]). Practically, the result in theorem 3 implies that overlapping convolutional networks with common characteristics of *e.g.* kernel size $K = 5$ and depth $L = 20$, can support the entanglement of any 2D system of interest up to sizes unattainable by such approaches. In comparison with previously suggested neural network based representations, the amount of parameters required for modeling volume law entanglement scaling in 2D is linear in A_{lin} for overlapping convolutional networks, versus quartic in A_{lin} for fully connected networks [31, 35] and quadratic in A_{lin} for RBMs [30, 36]. For many interesting physical questions, for example those related to the dynamics of non-equilibrium high energy states in generic clean systems [66], modeling volume law entanglement scaling is a necessity. The tractable representation of such entanglement scaling in large lattices, polynomially more efficient in popular deep learning architectures than in common RBM based approaches in 2D, may bring forth access to new quantum many-body phenomena.

Finally, the result in theorem 3 applies to a network with no spatial decimation (pooling) in its first L layers, such as employed in *e.g.* [67, 68]. In the supplementary material, we prove a result analogous to that of theorem 3 for overlapping networks that integrate pooling layers in between convolution layers. We show that in this case the volume law entanglement scaling is limited to system sizes under a cutoff equal to the convolution kernel size K, which is small in common convolutional network architectures. Practically, this suggests the use of overlapping convolutional networks without pooling operations for modeling highly entangled states, and overlapping convolutional networks that include pooling for modeling states that obey area law entanglement scaling.

Discussion.– The presented TN constructions of prominent deep learning architectures, served as a bidirectional bridge for transfer of concepts and results between the two domains. In one direction, it allowed us to convert well-established tools and approaches from many-body physics into new deep learning insights. An identified structural equivalence between many-body wave-functions and the functions realized by non-overlapping convolutional and shallow recurrent networks, brought forth the use of entanglement measures as well-defined quantifiers of the network's ability to model dependencies in the inputs. Via the TN construction of the above networks, in the form of Tree and MPS TNs (Fig. 1), we made use of a quantum physics result which bounds the entanglement represented by a generic TN [43] in order to propose a novel deep learning architecture design scheme. We were thus able to suggest principles for parameter allocation along the network (layer widths) and choice of network connectivity (pooling geometry), which were shown to correspond to the network's ability to model dependencies of interest.

In the opposite direction, we constructed TNs corresponding to powerful enhancements of the above architectures, in the form of deep recurrent networks and overlapping convolutional networks. These architectures, which stand at the forefront of recent deep learning achievements, inherently involve reuse of information along network computation. In order to construct their TN equivalents, we introduced a method of indices duplication, resulting in recursive MPS (Fig. 2) and Tree (Fig. 3) TNs. This method allowed us to demonstrate how a tensor representing an N-particle wave-function can be represented by the above architectures, and thus made available the investigation of their entanglement scaling properties. Relying on a recent result which establishes that depth has a super-polynomially enhancing effect on long-term memory capacity in recurrent networks [40], we showed that deep recurrent networks supports super-logarithmic corrections to the area law entanglement scaling in 1D. Similarly, by translating a recent result which shows that overlapping convolutional networks are exponentially more expressive than their non-overlapping counterparts [50], we showed that such architectures support volume law entanglement scaling for systems of linear sizes smaller than $K \cdot L$ (K is the convolution kernel size and L is the network depth), and area law entanglement scaling for larger systems. Our results show that overlapping convolutional networks are polynomially more efficient for modeling highly entangled states in 2D than competing neural network based representations.

The expressive power of deep learning architectures has been noticed by the many-body quantum physics community, and neural network representations of wave-functions have recently been suggested and analyzed. The novelty of our construction is twofold. Firstly, we analyze state-of-the-art deep learning approaches while previous suggestions focus on more traditional architectures such as fully-connected networks or RBMs. Secondly, we embed the deep learning apparatus within the TNs framework, which allowed us to compare the entanglement enhancing mechanisms used by deep networks with those employed by common expressive TN architectures such as MERA. In the future, this description may facilitate adaptation of standard TN tools such as *e.g.* [11–13] to comply with deep learning wave-function representations. A first step in this direction might be an adjustment of recently suggested networks (non-overlapping [46] and overlapping [69]) which obey l_1 normalization constraints by design, so that they obey l_2 normalization, and correspond more naturally to quantum wave-functions.

This paper draws similarities between disciplines, how-

ever it is important to note conceptual differences in order to fully exploit the power of deep learning architectures for many-body wave-function representations. The condensed matter physics community is in pursuit of computational schemes which can be as expressive as possible yet still efficient enough in order to be practically feasible. Though may be surprising to some readers, in machine learning excessive expressivity is not merely wasteful in resources but actually harms an architecture's performance (an effect commonly known as 'overfitting'). In some sense, this means that popular and successful deep learning architectures are not as expressive as they can be. The practical message for numerical physicists attempting to harness the power of deep learning, is to explore beyond off-the-shelf architectures that operate well in machine learning tasks, as those may unnecessarily restrict the available expressiveness. As our results indicate, the desirable volume law entanglement scaling supported by overlapping convolutional networks is effectively limited by common deep learning choices of using pooling operations and small convolution kernels.

Our established view of entanglement measures as quantifiers of dependencies supported by deep networks, indicates that this connection may help shed light on the question of characteristic dependencies in machine learning data sets. Physicists often have a clear understanding of the entanglement properties of the many-body system they wish to represent, which assists them in choosing an adequate TN architecture to represent it. In the machine learning domain, dependencies in natural data sets are yet to be adequately characterized. Empirical evidence suggests that the mutual information in various natural data-sets such as English Wikipedia, works of Bach, the human genome *etc.*, decays polynomially with a critical exponent similar in value to that of the critical Ising model [70]. Our results show that deep learning architectures can support the entanglement scaling of such critical systems. A future quantification of the 'characteristic entanglement' in natural data sets may shed light on the empirical success of deep learning architectures, and suggest further task specific design principles such as those brought forth in this work. Overall, we believe that the bidirectional bridge presented in this work can help bring quantum many-body physics research and state-of-the-art machine learning approaches one step closer together.

We thank Guifre Vidal, Thomas Spencer, John Imbrie, Bartlomiej Czech, Eyal Bairey and Eyal Leviatan for helpful discussions. This work is supported by Intel grant ICRI-CI #9- 2012-6133, by ISF Center grant 1790/12 and by the European Research Council (TheoryDL project). Nadav Cohen is supported by Eric and Wendy Schmidt.

* yoavlevine@cs.huji.ac.il
† or.sharir@cs.huji.ac.il
‡ cohennadav@ias.edu
§ shashua@cs.huji.ac.il

[1] Jens Eisert, Marcus Cramer, and Martin B Plenio, "Colloquium: Area laws for the entanglement entropy," Reviews of Modern Physics **82**, 277 (2010).
[2] Mark Fannes, Bruno Nachtergaele, and Reinhard F Werner, "Finitely correlated states on quantum spin chains," Communications in mathematical physics **144**, 443–490 (1992).
[3] David Perez-García, Frank Verstraete, Michael M Wolf, and J Ignacio Cirac, "Matrix product state representations," Quantum Information and Computation **7**, 401–430 (2007).
[4] Guifre Vidal, "Entanglement renormalization," Physical review letters **99**, 220405 (2007).
[5] Frank Verstraete and J Ignacio Cirac, "Renormalization algorithms for quantum-many body systems in two and higher dimensions," arXiv preprint cond-mat/0407066 (2004).
[6] Frank Verstraete, Valentin Murg, and J Ignacio Cirac, "Matrix product states, projected entangled pair states, and variational renormalization group methods for quantum spin systems," Advances in Physics **57**, 143–224 (2008).
[7] Zheng-Cheng Gu and Xiao-Gang Wen, "Tensor-entanglement-filtering renormalization approach and symmetry-protected topological order," Physical Review B **80**, 155131 (2009).
[8] Glen Evenbly and Guifré Vidal, "Tensor network states and geometry," Journal of Statistical Physics **145**, 891–918 (2011).
[9] Glen Evenbly and Guifre Vidal, "Scaling of entanglement entropy in the (branching) multiscale entanglement renormalization ansatz," Physical Review B **89**, 235113 (2014).
[10] Steven R White, "Density matrix formulation for quantum renormalization groups," Physical review letters **69**, 2863 (1992).
[11] Guifré Vidal, "Efficient simulation of one-dimensional quantum many-body systems," Phys. Rev. Lett. **93**, 040502 (2004).
[12] Steven R. White and Adrian E. Feiguin, "Real-time evolution using the density matrix renormalization group," Phys. Rev. Lett. **93**, 076401 (2004).
[13] Roman Orus and Guifre Vidal, "Infinite time-evolving block decimation algorithm beyond unitary evolution," Physical Review B **78**, 155117 (2008).
[14] Glen Evenbly and Guifré Vidal, "Algorithms for entanglement renormalization," Physical Review B **79**, 144108 (2009).
[15] Matteo Rizzi, Simone Montangero, and Guifre Vidal, "Simulation of time evolution with multiscale entanglement renormalization ansatz," Physical Review A **77**, 052328 (2008).
[16] Ulrich Schollwöck, "The density-matrix renormalization group in the age of matrix product states," Annals of Physics **326**, 96–192 (2011).
[17] Naoki Nakatani and Garnet Kin-Lic Chan, "Efficient tree tensor network states (ttns) for quantum chemistry: Generalizations of the density matrix renormalization group algorithm," The Journal of chemical physics **138**, 134113 (2013).
[18] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton,

"Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*, edited by F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger (Curran Associates, Inc., 2012) pp. 1097–1105.
[19] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint arXiv:1409.1556 (2014).
[20] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich, "Going Deeper with Convolutions," CVPR (2015).
[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016) pp. 770–778.
[22] Ilya Sutskever, James Martens, and Geoffrey E Hinton, "Generating text with recurrent neural networks," in *Proceedings of the 28th International Conference on Machine Learning (ICML-11)* (2011) pp. 1017–1024.
[23] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton, "Speech recognition with deep recurrent neural networks," in *Acoustics, speech and signal processing (icassp), 2013 ieee international conference on* (IEEE, 2013) pp. 6645–6649.
[24] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio, "Neural machine translation by jointly learning to align and translate," arXiv preprint arXiv:1409.0473 (2014).
[25] Dario Amodei, Sundaram Ananthanarayanan, Rishita Anubhai, Jingliang Bai, Eric Battenberg, Carl Case, Jared Casper, Bryan Catanzaro, Qiang Cheng, Guoliang Chen, *et al.*, "Deep speech 2: End-to-end speech recognition in english and mandarin," in *International Conference on Machine Learning* (2016) pp. 173–182.
[26] Edwin Stoudenmire and David J Schwab, "Supervised learning with tensor networks," in *Advances in Neural Information Processing Systems 29*, edited by D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett (Curran Associates, Inc., 2016) pp. 4799–4807.
[27] Zhao-Yu Han, Jun Wang, Heng Fan, Lei Wang, and Pan Zhang, "Unsupervised generative modeling using matrix product states," arXiv preprint arXiv:1709.01662 (2017).
[28] E Miles Stoudenmire, "Learning relevant features of data with multi-scale tensor networks," arXiv preprint arXiv:1801.00315 (2017).
[29] Jing Chen, Song Cheng, Haidong Xie, Lei Wang, and Tao Xiang, "Equivalence of restricted boltzmann machines and tensor network states," Phys. Rev. B **97**, 085104 (2018).
[30] Giuseppe Carleo and Matthias Troyer, "Solving the quantum many-body problem with artificial neural networks," Science **355**, 602–606 (2017).
[31] Hiroki Saito, "Solving the bose–hubbard model with machine learning," Journal of the Physical Society of Japan **86**, 093001 (2017).
[32] Dong-Ling Deng, Xiaopeng Li, and S. Das Sarma, "Machine learning topological states," Phys. Rev. B **96**, 195145 (2017).
[33] Xun Gao and Lu-Ming Duan, "Efficient representation of quantum many-body states with deep neural networks," Nature communications **8**, 662 (2017).
[34] Giuseppe Carleo, Yusuke Nomura, and Masatoshi Imada, "Constructing exact representations of quantum many-body systems with deep neural networks," arXiv preprint arXiv:1802.09558 (2018).
[35] Zi Cai and Jinguo Liu, "Approximating quantum many-body wave functions using artificial neural networks," Physical Review B **97**, 035116 (2018).
[36] Dong-Ling Deng, Xiaopeng Li, and S. Das Sarma, "Quantum entanglement in neural network states," Phys. Rev. X **7**, 021021 (2017).
[37] Nadav Cohen, Or Sharir, and Amnon Shashua, "On the expressive power of deep learning: A tensor analysis," Conference On Learning Theory (COLT) (2016).
[38] Nadav Cohen and Amnon Shashua, "Convolutional rectifier networks as generalized tensor decompositions," International Conference on Machine Learning (ICML) (2016).
[39] Nadav Cohen and Amnon Shashua, "Inductive bias of deep convolutional networks through pooling geometry," in *5th International Conference on Learning Representations (ICLR)* (2017).
[40] Yoav Levine, Or Sharir, and Amnon Shashua, "Benefits of depth for long-term memory of recurrent networks," arXiv preprint arXiv:1710.09431 (2017).
[41] Valentin Khrulkov, Alexander Novikov, and Ivan Oseledets, "Expressive power of recurrent neural networks," in *6th International Conference on Learning Representations (ICLR)* (2018).
[42] Martin B Plenio and Shashank Virmani, "An introduction to entanglement measures," Quantum Information and Computation **7**, 001–051 (2007).
[43] Shawn X Cui, Michael H Freedman, Or Sattath, Richard Stong, and Greg Minton, "Quantum max-flow/min-cut," Journal of Mathematical Physics **57**, 062206 (2016).
[44] Wolfgang Hackbusch, *Tensor spaces and numerical tensor calculus*, Vol. 42 (Springer Science & Business Media, 2012).
[45] Tamara G Kolda and Brett W Bader, "Tensor decompositions and applications," SIAM review **51**, 455–500 (2009).
[46] Or Sharir, Ronen Tamari, Nadav Cohen, and Amnon Shashua, "Tractable generative convolutional arithmetic circuits," arXiv preprint arXiv:1610.04167 (2016).
[47] Nadav Cohen, Or Sharir, and Amnon Shashua, "Deep simnets," IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016).
[48] Yuhuai Wu, Saizheng Zhang, Ying Zhang, Yoshua Bengio, and Ruslan R Salakhutdinov, "On multiplicative integration with recurrent neural networks," in *Advances in Neural Information Processing Systems* (2016) pp. 2856–2864.
[49] Yoav Levine, David Yakira, Nadav Cohen, and Amnon Shashua, "Deep learning and quantum entanglement: Fundamental connections with implications to network design," in *6th International Conference on Learning Representations (ICLR)* (2018).
[50] Or Sharir and Amnon Shashua, "On the expressive power of overlapping architectures of deep learning," in *6th International Conference on Learning Representations (ICLR)* (2018).
[51] Wolfgang Hackbusch and Stefan Kühn, "A new scheme for the tensor representation," Journal of Fourier analysis and applications **15**, 706–722 (2009).
[52] Ivan V Oseledets, "Tensor-train decomposition," SIAM Journal on Scientific Computing **33**, 2295–2317 (2011).
[53] In a statistical setting, where $f(\cdot)$ is a probability density

function, separability w.r.t. (A, B) corresponds to statistical independence between inputs from A and B.
[54] Gregory Beylkin and Martin J Mohlenkamp, "Numerical operator calculus in higher dimensions," Proceedings of the National Academy of Sciences **99**, 10246–10251 (2002).
[55] The equivalence of the Schmidt number and the separation rank follows from the linear independence of the representation functions.
[56] See [49] for treatment of a general channel numbers setting.
[57] Michiel Hermans and Benjamin Schrauwen, "Training and analysing deep recurrent neural networks," in *Advances in Neural Information Processing Systems* (2013) pp. 190–198.
[58] This scenario of representing inputs to an RNN as 'one-hot' vectors is actually quite common in sequential tasks, see *e.g.* [71].
[59] We focus on the case where A is located to the right B for proof simplicity, numerical simulations of the network in Fig. 2 with randomized weight matrices indicate that the lower bound in theorem 2 holds for all other locations of A.
[60] Emanuel Gull, Olivier Parcollet, and Andrew J Millis, "Superconductivity and the pseudogap in the two-dimensional hubbard model," Physical review letters **110**, 216405 (2013).
[61] K-S Chen, Zi Yang Meng, S-X Yang, Thomas Pruschke, Juana Moreno, and Mark Jarrell, "Evolution of the superconductivity dome in the two-dimensional hubbard model," Physical Review B **88**, 245110 (2013).
[62] Michael Lubasch, J Ignacio Cirac, and Mari-Carmen Banuls, "Algorithms for finite projected entangled pair states," Physical Review B **90**, 064425 (2014).
[63] Bo-Xiao Zheng and Garnet Kin-Lic Chan, "Ground-state phase diagram of the square lattice hubbard model from density matrix embedding theory," Physical Review B **93**, 035126 (2016).
[64] Wen-Yuan Liu, Shao-Jun Dong, Yong-Jian Han, Guang-Can Guo, and Lixin He, "Gradient optimization of finite projected entangled pair states," Physical Review B **95**, 195154 (2017).
[65] JPF LeBlanc, Andrey E Antipov, Federico Becca, Ireneusz W Bulik, Garnet Kin-Lic Chan, Chia-Min Chung, Youjin Deng, Michel Ferrero, Thomas M Henderson, Carlos A Jiménez-Hoyos, *et al.*, "Solutions of the two-dimensional hubbard model: Benchmarks and results from a wide range of numerical algorithms," Physical Review X **5**, 041041 (2015).
[66] Ehud Altman and Ronen Vosk, "Universal dynamics and renormalization in many-body-localized systems," Annu. Rev. Condens. Matter Phys. **6**, 383–409 (2015).
[67] Aaron van den Oord, Nal Kalchbrenner, and Koray Kavukcuoglu, "Pixel recurrent neural networks," arXiv preprint arXiv:1601.06759 (2016).
[68] Aaron van den Oord, Nal Kalchbrenner, Lasse Espeholt, Oriol Vinyals, Alex Graves, *et al.*, "Conditional image generation with pixelcnn decoders," in *Advances in Neural Information Processing Systems* (2016) pp. 4790–4798.
[69] Or Sharir and Amnon Shashua, "Sum-product-quotient networks," arXiv preprint arXiv:1710.04404 (2017).
[70] Henry W Lin and Max Tegmark, "Critical behavior from deep dynamics: A hidden dimension in natural language," arXiv preprint arXiv:1606.06737 (2016).
[71] Alex Graves, "Generating sequences with recurrent neural networks," arXiv preprint arXiv:1308.0850 (2013).

SUPPLEMENTARY MATERIAL

A. "No-Cloning" in Tensor Networks

The required operation of duplicating a vector and sending it to be part of two different calculations, which is simply achieved in any practical setting, is actually impossible to represent in the framework of TNs. We formulate this notion in the following claim:

Claim 1 *Let $v \in \mathbb{R}^P, P \in \mathbb{N}$ be a vector. v is represented by a node with one leg in the TN notation. The operation of duplicating this node,* i.e. *forming two separate nodes of degree* 1*, each equal to v, cannot be achieved by any TN.*

Proof. We assume by contradiction that there exists a TN ϕ which operates on any vector $v \in \mathbb{R}^P$ and clones it to two separate nodes of degree 1, each equal to v, to form an overall TN representing $v \otimes v$. Component wise, this implies that ϕ upholds $\forall v \in \mathbb{R}^P : \sum_{i=1}^P \phi_{ijk} v_i = v_j v_k$. By our assumption, ϕ duplicates the standard basis elements of $\mathbb{R}^P$, denoted $\{\hat{\mathbf{e}}^{(\alpha)}\}_{\alpha=1}^P$, meaning that $\forall \alpha \in [P]$:

$$\sum_{i=1}^{P} \phi_{ijk} \hat{e}_i^{(\alpha)} = \hat{e}_j^{(\alpha)} \hat{e}_k^{(\alpha)}. \tag{7}$$

By definition of the standard basis elements, the left hand side of Eq. (7) takes the form $\phi_{\alpha jk}$ while the right hand side equals 1 only if $j = k = \alpha$, and otherwise 0. In other words, in order to successfully clone the standard basis elements, Eq. (7) implies that ϕ must uphold $\phi_{\alpha jk} = \delta_{\alpha jk}$. However, for $v = \mathbf{1}$, *i.e.* $\forall j \in [P] : v_j = 1$, a cloning operation does not take place when using this value of ϕ, since $\sum_{i=1}^P \phi_{ijk} v_i = \sum_{i=1}^P \delta_{ijk} = \delta_{jk} \neq 1 = v_i v_j$, in contradiction to ϕ duplicating any vector in $\mathbb{R}^P$.

□

B. Entanglement Scaling in Deep Recurrent Networks

In the following, we prove the result in theorem 2 regarding the entanglement scaling supported by deep RACs:

Proof (of theorem 2). In [40], a lower bound of $\binom{\min\{R,M\}+N/2-1}{N/2}$ is shown for A that is placed to the right of B and $|A| = |B| = N/2$, for which the size of $|A|$ is the largest possible under the conditions of theorem 2. Essentially, the combinatorial dependence of the

lower bound follows from the indistinguishability of duplicated indices. Given $|A| \leq |B|$, we designate the final $|A|$ indices of the set B to form a set B^* which upholds by definition $|B^*| = |A|$. The lower bound in theorem 2 is obtained by replacing $N/2$ with $|A|$ and continuing with the same exact proof procedure as in [40], applied to B^* and A, when all the residual initial $|B| - |A|$ indices, corresponding to the set $B \backslash B^*$, are kept fixed.

□

C. Entanglement Scaling in Overlapping Convolutional Networks

The following theorem quantifies the effect of pooling layers in overlapping convolutional networks:

Theorem 4 *Under similar conditions to theorem 3, when introducing 2^d pooling operations in between convolution layers (Fig. 5), the maximal entanglement entropy w.r.t. (A, B) modeled by $\mathrm{DUP}(\mathcal{A}^{K,L,d})$ obeys:*

$$\Omega\left(\min\left\{\left(A_{lin}\right)^d, K\left(A_{lin}\right)^{d-1}\right\}\right).$$

Thus, the introduction of such pooling layers results in a diminished ability of the overlapping convolutional network to represent volume law entanglement scaling. In the following, we prove the results in theorems 3 and 4 regarding entanglement scaling supported by overlapping ConvACs:

Proof (of theorems 3 and 4). We begin by providing a succinct summary of the theoretical analysis of overlapping ConvACs that was shown by [50], including the necessary technical background on ConvACs required to understand their results. [50] shows lower bounds on the rank of the dup-tensor for various architectures when A is left half of the input and B the right half, in $d = 2$, when the convolutional kernel is anchored at the corner instead of at the center like presented in this paper.

For any layer $l \in [L]$ in a convolutional network, the local receptive field (or kernel size) $K^{(l)}$ is defined as the linear size of the window on which each convolutional kernel acts upon, and the stride $S^{(l)}$ is defined as the step size in each dimension between two neighboring windows (assumed to be 1 in this paper). The main result of [50] relies on two architecture dependent attributes that they referred to as the total receptive field and and the total stride of the l'th layer, defined as the projections on the input layer of the local receptive fields and strides from the perspective of the l'th layer, as illustrated in Fig. 6. In their main result they show that the first layer that has a total receptive field of at least half the linear dimension of the input size N, denoted l_0, gives rise to a lower bound on the rank of the matricized tensor that is proportional to $\min\{M, r_0, \ldots, r_{L-1}\}^{\mathcal{O}(N^2/4 \cdot T_S^{(l_0)})}$, where $T_S^{(l_0)}$ is the total stride of the l_0'th layer. To prove this result the authors rely on the ability of a sufficiently large total receptive field to represent identity matrices between pairs of input indices, each pair comprising one index from A and one from B, where the total stride limits the maximal number of pairs as it denotes the minimal distance between any two given pairs of indices.

To prove the lower bounds on the architectures described in theorems 3 and 4, it is sufficient to consider just the first $\tilde{L}$ convolutional layers with unit strides, specifically $\tilde{L} = L$ for the case of a sequence of K^d conv layers followed by global pooling (see Fig. 3a), and $\tilde{L} = 1$ for the case of alternating K^d conv and 2^d pooling layers (Fig. 5a). Under the above, the total receptive field of the $\tilde{L}$'th layer is simply $(K-1) \cdot \tilde{L} + 1$, which can be thought of a single large convolutional layer. Now, following the same proof sketch described above, we can use the combined convolutional layer to pair indices of A and B along the boundary between the two sets, where the size of the total receptive field determines the maximal number of pairs we can capture around each point on the boundary. In the special case of $K\tilde{L} > A_{\mathrm{lin}}$, nearly any index of A could be paired with a unique index from B. This results in a lower bound of $\min\{M, r_0, \ldots, r_{L-1}\}^{\Omega\left(\min\left\{(A_{\mathrm{lin}})^d, K\tilde{L}(A_{\mathrm{lin}})^{d-1}\right\}\right)}$.

□

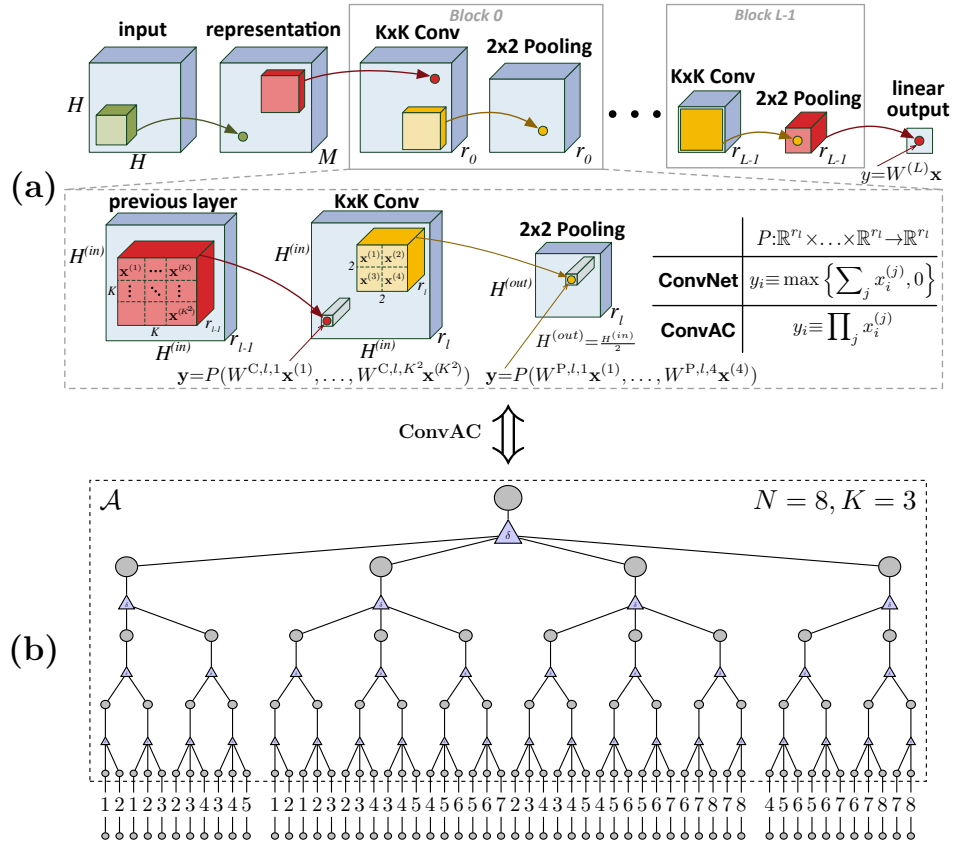

FIG. 5. A deep overlapping convolutional network with 2×2 pooling (decimation) layers in between convolution layers, and its TN equivalent in the 1D case for convolution kernel size is $K = 3$, spatial extent $N = 8$ (the network depth is given by $L = \log_2 N$ due to the pooling layers). This network supports volume law entanglement scaling until the linear dimension of the represented system exceeds the order of K (theorem 3). For legibility, $j \in [8]$ stands for d_j in this figure.

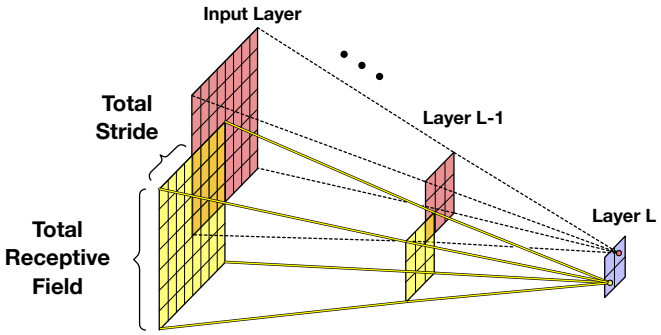

FIG. 6. Illustration of the total receptive field and the total stride.