Science Bulletin 64 (2019) 1228–1233

Contents lists available at ScienceDirect

Science Bulletin

journal homepage: www.elsevier.com/locate/scib

Article

Emergent Schrödinger equation in an introspective machine learning architecture

Ce Wang [a], Hui Zhai [a,*], Yi-Zhuang You [b,*]

[a] Institute for Advanced Study, Tsinghua University, Beijing 100084, China
[b] Department of Physics, University of California, San Diego, CA 92093, USA

ARTICLE INFO

Article history:
Received 10 June 2019
Received in revised form 6 July 2019
Accepted 8 July 2019
Available online 22 July 2019

Keywords:
Quantum physics
Machine learning
Potential-to-density mapping
Neural network
Recurrent autoencoder

ABSTRACT

Can physical concepts and laws emerge in a neural network as it learns to predict the observation data of physical systems? As a benchmark and a proof-of-principle study of this possibility, here we show an introspective learning architecture that can automatically develop the concept of the quantum wave function and discover the Schrödinger equation from simulated experimental data of the potential-to-density mappings of a quantum particle. This introspective learning architecture contains a machine translator to perform the potential to density mapping, and a knowledge distiller auto-encoder to extract the essential information and its update law from the hidden states of the translator, which turns out to be the quantum wave function and the Schrödinger equation. We envision that our introspective learning architecture can enable machine learning to discover new physics in the future.

1. Introduction

The ongoing third wave of artificial intelligence has made great achievements in employing neural-network-based machine learning for industry and social applications. Inspired by this great success, machine learning algorithms have also been rapidly applied to various directions of physics research, ranging from high-energy and string theory to condensed matter, atomic, molecular and optical physics [1–12]. While there has been many successful examples of machine assisted physics research, it remains an ambitious goal to explore the potential of machine learning in unsupervised discovery of concepts and laws of physics from observation data [13,14]. A major challenge is to understand how the machine "thinks", or what approaches have been developed inside its mind. This typically requires us to open up the black box of the neural network and to identify the most relevant emergent features in the neural activity. Can the analysis of the neural activity also be automated by the machine itself? Can knowledge emerges as the machine examines its own information flow introspectively? To demonstrate these possibilities, here we report an introspective learning architecture, as illustrated in Fig. 1, that allows the machine to distill the knowledge about quantum mechanics from the observation of the density distributions of a quantum particle in different shapes of potentials.

2. Recurrent neural network translator

As a proof-of-concept study, we consider a single quantum particle moving in a one-dimensional space with certain potential. Suppose we can measure the probability density of the particle for each given potential, we supply the machine with the potential profile as the input and the density profile as the target, and challenge the machine to discover the underlying rule governing the potential-to-density mapping. We discretize the potential $V(x)$ and density profiles $\rho(x)$ along the one-dimensional space and treat them as sequences of real numbers: $V_i = V(x_i)$ and $\rho_i = \rho(x_i)$, where $x_i = ai$ are the discrete coordinates for $i = 0, 1, 2, \cdots$, which are evenly distributed along the one-dimensional space with a fixed separation $a = 0.1$. We assume that the potential is always measured with respect to the energy of the particle, such that the particle energy is effectively fixed at zero. We will only consider the case of $V_i < 0$, such that the particle remains in extended states.

By treating both the potential and density profiles as sequential data, the potential-to-density problem belongs to a broader class of sequence-to-sequence mapping [15–18], which can be handled by the recurrent neural network (RNN) [19]. The RNN has been widely used in natural language processing to translate sequences of words from the source language to the target language [20]. We

* Corresponding authors.
E-mail addresses: hzhai@tsinghua.edu.cn (H. Zhai), yzyou@physics.ucsd.edu (Y.-Z. You).

https://doi.org/10.1016/j.scib.2019.07.014
2095-9273/

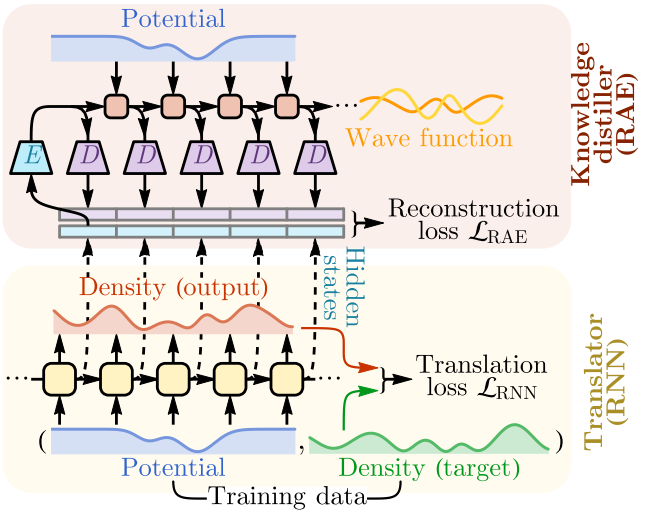

Fig. 1. (Color online) The architecture of an introspective recurrent neural network, called "the Schrödinger machine". It contains a translator (lower panel) and a knowledge distiller (upper panel). The translator is implemented as a recurrent neural network to perform the task of the potential-to-density mapping. The knowledge distiller compresses the hidden states generated by the translator using a recurrent auto-encoder and extracts the most essential variables in the hidden states together with its update rule.

apply the RNN architecture to perform the potential-to-density mapping as a translation task. In each step, the RNN takes an input V_i from the source sequence, modifies its internal hidden state h_i accordingly, and generates the output ρ'_i based on the hidden state, as illustrated in Fig. 2a. We adopt the following update equations

$$h_i = W(V_i)\cdot h_{i-1}, \quad \rho'_i = P(h_i), \tag{1}$$

where both the input $V_i \in \mathbb{R}$ and the output $\rho'_i \in \mathbb{R}$ are scalars and the hidden state $h_i \in \mathbb{R}^d$ is a d-dimensional vector. The hidden state h_i is updated by an input-dependent linear transformation, represented by a $d \times d$ matrix $W(V_i) \in \mathbb{R}^{d\times d}$ multiplied to the vector h_i. The output ρ'_i is generated from the hidden state by a projection map $P(h_i)$. The data flow is graphically represented in Fig. 2b. The output sequence ρ'_i is then compared with the target sequence ρ_i over a window of steps to evaluate the loss function

$$\mathcal{L}_{\rm RNN} = \sum_{i\in \text{window}} \left(\rho'_i - \rho_i\right)^2. \tag{2}$$

How the RNN updates its hidden state and generates output is determined by the functions W and P. In general, W and P could be non-linear functions modeled by feedforward neural networks for instance. However, for our problem, we find it sufficient to model W by a Taylor expansion (to the n_Wth order in V_i) and P by a linear projection,

$$W(V_i) = \sum_{n=0}^{n_W} W^{(n)} V_i^n, \quad P(h_i) = p^{\intercal}\cdot h_i, \tag{3}$$

where $W^{(n)}$ is the nth order Taylor expansion coefficient matrix (each of the dimension $d \times d$) and p is a d-dimensional vector. The elements in $W^{(n)}$ and p are model parameters to be trained to minimize the loss function $\mathcal{L}_{\rm RNN}$. The training dataset contains pairs of potential and density sequences that serve as parallel corpora to train the RNN translator. They are currently obtained from numerical simulation (see Appendix A for details about data acquisition), but can be collected from experiments in future applications, from instance, the quantum gas microscope can detect density of ultracold atoms nearly in their ground state in situ in the presence of different kind of potentials generated by optical speckles [21]. After

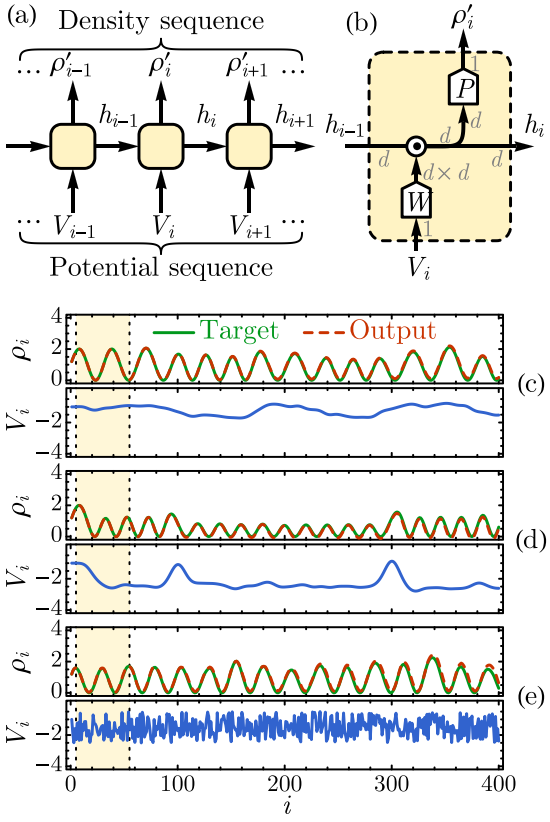

Fig. 2. (Color online) Architecture of the translator RNN for the potential-to-density mapping. (a) is the global structure and (b) is the network structure within each block. Arrows indicate the direction of information flow. The tensor dimensions are marked out in gray. W and P can be generic functions, although they are modeled by the Taylor expansions in our implementation. The symbol $\odot$ denotes matrix-vector multiplication. (c–e) Typical samples of the RNN output density profiles in comparison with the target density profiles for (c) a shallow and smooth potential, (d) a deep but smooth potential, and (e) a shallow but rough potential. The model is only trained on a small window indicated by the yellow shaded region. The trained RNN can perform the potential-to-density mapping over a much larger range.

minimizing the translator loss $\mathcal{L}_{\rm RNN}$, the RNN can predict the density profile based on the potential profile (see the Appendix A for network parameters).

We build the RNN with the Taylor expansion order $n_W = 2$ and the hidden state dimension up to $d = 6$. We observe that the loss $\mathcal{L}_{\rm RNN}$ will drop significantly as long as $d \geqslant 3$ (see the Appendix B for detailed analysis). Using the RNN model for the one-dimensional potential-to-density mapping is physically grounded because it respects the translational symmetry of the physical law that governs this mapping. Here we should emphasize that knowing the symmetry of the physics law is very different from knowing physical law as a prior. It is always natural to assume all physical laws respect the translational symmetry, and this does not mean we have used any concrete information of the physical law as a prior. With the assumption of the translational symmetry, there are still infinite possibilities of the physical laws described by the ordinary differential equations and the major goal of this work is to see whether the machine can find out the right one from data.

As a result of the translational symmetry, an immediate advantage of the RNN is to gain spatial scalability, that is, what has been learned over a small system can be readily generalized and applied to larger systems. For instance, as shown in Fig. 2c–e, the RNN is trained over a small window from $i = 5$ to $i = 55$ (the initial 5 outputs are excluded to reduce the sensitivity to initial conditions). After training, the RNN can perform the potential-to-density mapping for a much larger system, from $i = 0$ to $i = 400$.

Fig. 2c–e shows that the RNN output matches nicely with the target density profile on the test dataset for different classes of potential profiles, either shallow or deep, and either smooth or rough. The relative error does accumulates with the RNN steps (as error generally accumulates in recurrence), but remains within 10% towards the end of our prediction period (400 steps). This result demonstrates the prediction power of the RNN model.

3. Recurrent autoencoder knowledge distiller

By learning to perform the potential-to-density mapping, the RNN translator must have developed some intuitions about the underlying physics. Historically, advances in physics are often marked by formulating physical phenomena in term of differential equations, such as Newton's law of motion, Maxwell's equation of electromagnetism, and the Schrödinger equation of quantum mechanics. The RNN provides a universal representation of recurrent equations as discretized versions of the differential equations, and therefore the update rules of its hidden state can be interpreted as machine's understanding of the physical laws [22,23]. As the RNN performs the translation, it generates a sequence of hidden states containing the essential variables governing the physics of potential-to-density mapping, mixed with other redundant or irrelevant information. To extract the knowledge from these hidden state data, we design a higher-level machine, called the knowledge distiller, to learn from the neural activity (the hidden state sequence) of the lower-level translator. It works on the RNN hidden states to compress the information and to extract the underlying rule. The auto-encoder architecture is widely used for information compression [24,25]. Here we incorporate the auto-encoder in another recurrent neural network structure as a recurrent auto-encoder (RAE), because we not only need to find out the essential variables in the hidden states but also need to determine the update rules of these essential variables.

The architecture of the RAE knowledge distiller is illustrated in Fig. 3. The RAE distiller first encodes the hidden state h_{i_0} of the RNN translator at a given step i_0 to the latent variable g_i, and then tries to reconstruct the hidden states h_i for subsequent steps ($i \geqslant i_0$) by evolving and decoding the latent variable. The update equations are given by

$$\begin{aligned} &g_{i_0} = E(h_{i_0}),\\ &g_i = \widetilde{W}(V_i)\cdot g_{i-1}, \quad (i = i_0+1, i_0+2, \cdots),\\ &h'_i = D(g_i), \quad (i = i_0, i_0+1, i_0+2, \cdots), \end{aligned} \tag{4}$$

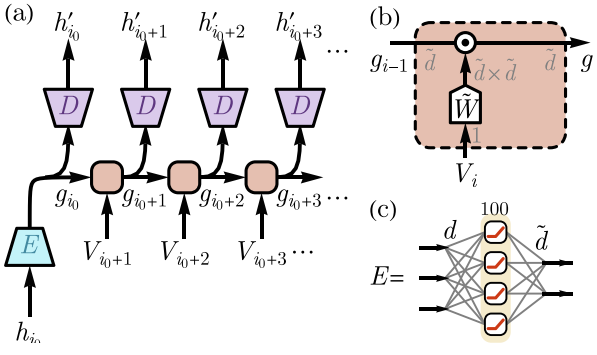

Fig. 3. (Color online) Architecture of the recurrent auto-encoder. (a) The global structure. (b) The network structure within each recurrent block. (c) The feedforward network of the encoder E. Arrows indicate the direction of information flow. Tensor dimensions are marked out in gray. In (b), $\widetilde{W}$ can be a generic function. The symbol $\odot$ denotes matrix-vector multiplication. In (c), we use one hidden layer of 100 dimension, with the Rectifier Linear Unit (ReLU) activation. The decoder D has a similar feedforward network in a revered structure as (c).

where E and D represent the encoder and decoder maps respectively. Here the RAE hidden state $g_i \in \mathbb{R}^{\tilde{d}}$ is updated by an linear transformation $\widetilde{W}(V_i)$ that will still depend on the input potential sequence V_i, as illustrated in Fig. 3b. The encoder and the decoder are implemented by feedforward networks as shown in Fig. 3c. The RAE is trained to minimize the reconstruction loss

$$\mathcal{L}_{\text{RAE}} = \sum_{i\in\text{window}} (h'_i - h_i)^2. \tag{5}$$

It is important that the RAE (knowledge distiller) hidden state g_i has a smaller dimension $\tilde{d}$ compared to the dimension d of the RNN (translator) hidden state h_i, therefore it can enforce an information bottleneck that only allows the essential information to be passed down in g_i. We will not specify a fixed dimension $\tilde{d}$ at this point. The optimal choice of $\tilde{d}$ will be determined by monitoring the reconstruction loss, as to be discussed soon.

We would like to mention that instead of using a single auto-encoder to compress the hidden state at each step independently, the RAE connects a series of decoders together by a recurrent neural network. This design is to ensure that the latent representation g_i remains coherent among a series of steps and contains the key variables that should be passed down along the sequence. A similar RAE architecture was proposed in Ref. [26] and recently redesigned in Ref. [13] to enable AI scientific discovery on sequential data. In this way, the RAE compresses the original RNN to a more compact RNN capturing the most essential information and its induced update rules.

As shown in Fig. 4a, we find that the reconstruction loss $\mathcal{L}_{\text{RAE}}$ of the RAE increases dramatically only when its hidden state dimension $\tilde{d}$ is squeezed below two (i.e. $\tilde{d} < 2$), implying that the key feature can be stored in a two-component real vector (i.e. $\tilde{d} = 2$) in the most parsimonious manner, as $g_i = (g_{i,1}, g_{i,2})$. Here we show that g_i in fact represents the quantum wave function and its first order derivation. The evidences are twofold.

First, we try to use the trained RNN to predict the density with a constant potential V, the result of which should be $\cos^2(kx_i)$ with

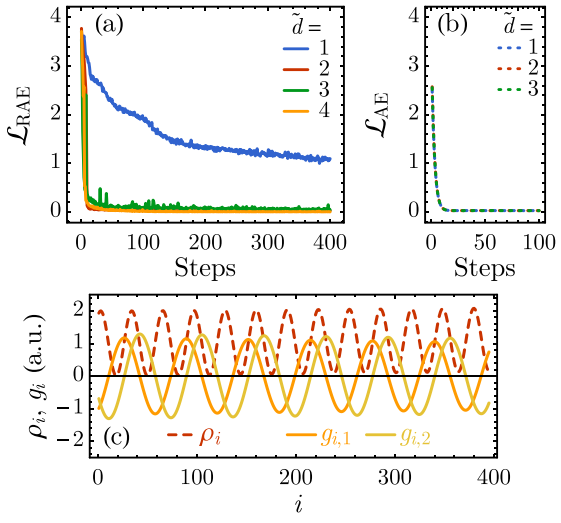

Fig. 4. (Color online) (a) The RAE reconstruction loss $\mathcal{L}_{\text{RAE}}$ vs. the training steps for the quantum case. Different curves are for different RAE hidden state dimensions $\tilde{d}$. $\tilde{d} = 2$ turns out to be the minimal $\tilde{d}$ without sacrificing the reconstruction loss. (b) The AE reconstruction loss $\mathcal{L}_{\text{AE}}$ vs. training steps for the classical thermal gas. The vanishing $\mathcal{L}_{\text{AE}}$ implies that there is no need to pass any variable along the sequence in this case. (c) The RNN output density profile ρ_i and the RAE hidden state $g_i = (g_{i,1}, g_{i,2})$ for a constant potential $V_i = 1$. It shows that the periodicity of g_i is twice of ρ_i.

$k=\sqrt{-V}$ being the momentum. If $g_{i,1}$ and $g_{i,2}$ are the wave function and its derivative, it should be $\cos(kx_i)$ and $\sin(kx_i)$, respectively, whose periods are twice of the period of ρ_i with phases shifted by $\pi/2$ relative to each other. As shown in Fig. 4c, g_i indeed displays the periodicity doubling and the relative phase shift.

Second, we open up the recurrent block of the RAE to extract the update rules for g_i, which is machine's formulation of the physical rules. The update rules are encoded in the transformation matrix $\widetilde{W}(V_i)=\sum_{n=0}^{n_w}\widetilde{W}^{(n)}V_i^n$, which are parameterized by the Taylor expansion coefficient matrices $\widetilde{W}^{(n)}$. At this point, we can already claim that the machine has summarized the governing equation for the essential variable g_i as a recurrent equation $g_i=\widetilde{W}(V_i)\cdot g_{i-1}$, which can be regarded as machine's version of the Schrödinger equation. It may look different from the discrete version of the Schrödinger equation that we are familiar with, because there is an ambiguity in the basis choice of g_i. In another word, there are a class of equivalent equations related by linear basis transformations $\widetilde{W}\to M^{-1}\widetilde{W}M$ for $M\in \mathrm{GL}(2,\mathbb{R})$, and machine can discover any one of them. To make connection to our familiar version of Schrödinger equation, we do need some human input. We find that it is always possible to find a proper linear transformation that can simultaneously bring all $\widetilde{W}^{(n)}$ to the following form

$$\begin{aligned} M^{-1}\widetilde{W}^{(0)}M&=\begin{bmatrix}0.9993 & 0.1007\\ 0.0013 & 0.9987\end{bmatrix}\approx\begin{bmatrix}1 & a\\ 0 & 1\end{bmatrix},\\ M^{-1}\widetilde{W}^{(1)}M&=\begin{bmatrix}0.0067 & 0.0004\\ 0.1001 & 0.0024\end{bmatrix}\approx\begin{bmatrix}0 & 0\\ a & 0\end{bmatrix}.\end{aligned} \tag{6}$$

Here the numerical matrix elements are what we obtained from a particular instance of the trained RAE. They can be associated to the lattice constant a to the leading order given that $a=0.1$, and we have also verified that they scale correctly with a as proposed. The result in Eq. (6) points to the following difference equation

$$\begin{bmatrix}g_{i+1,1}\\ g_{i+1,2}\end{bmatrix}=\begin{bmatrix}1 & a\\ aV_i & 1\end{bmatrix}\begin{bmatrix}g_{i,1}\\ g_{i,2}\end{bmatrix}. \tag{7}$$

If we interpret $g_{i,1}$ as the quantum wave function $\psi(x_i)$ and $g_{i,2}$ as its first order derivative $\partial_x\psi(x_i)$, Eq. (7) corresponds to a discrete version of the Schrödinger equation $\partial_x^2\psi(x)=V(x)\psi(x)$ as the particle energy is taken to be zero. So the RAE identifies two real numbers as the essential variables in the hidden states. They can be interpreted as the quantum wave function and its first order derivative. Their update rule is consistent with the Schrödinger equation. Admittedly, we rely on human intelligence to show that the machines formulation is equivalent to the Schrödinger equation under change of variables, but we do not have to do so. The only purpose of formulating it in the version we familiar with is to breach-mark the result.

In this way, without any prior knowledge of quantum mechanics, the introspective learning architecture can identify the essential variables and extract their governing equation from the data of potential and density profiles. The emergent variables and governing equation can be interpreted as the quantum wave function and the Schrödinger equation. As a consistency check, we train the same introspective recurrent neural network on the potential and density data of the high-temperature thermal gas following $\rho_i\propto \mathrm{e}^{-\beta V_i}$ at a fixed inverse temperature β. In this case, we can even reduce the RAE to an auto-encoder (AE) without sacrificing the reconstruction loss $\mathcal{L}_{\mathrm{AE}}$. As shown in Fig. 4b, the $\mathcal{L}_{\mathrm{AE}}$ remains vanishing for any $\tilde{d}$, implying that there is no need to pass any variable along the sequence and hence the Schödinger equation will not emerge for thermal gas.

4. Conclusion

In conclusion, we design the architecture that combines a task machine directly learning the experimental data and an introspective machine working on the neural activations of the task machine. The separation of the task machine from the introspective machine effectively isolates the knowledge distillation from affecting the task performance, such that the whole system can simultaneously improve the task performance and approach the parsimonious limit of knowledge representation, without trading off between one another. Here we show that this architecture can discover the Schrödinger equation from the potential-to-density data. We envision that the same architecture can be generally applied to other machine learning applications to physics problems and enable machine learning to discover new physics in the future.

Besides, there are another few points worth highlighting in this work. Firstly, although the use of Taylor expansion for the nonlinear functions in our RNN is not essential and can be replaced by neural network models, it has the advantage of being analytical tractability which makes it easier to understand how the RNN works. Secondly, the potential-to-density mapping is also an essential component in the density functional theory, known as the Kohn–Sham mapping [27]. The existing machine learning solutions for this task include the kernel method and the convolutional neural network approach [28–32]. The RNN approach introduced here has the advantage of being spatially scalable without retraining, which could find potential applications in boosting the density functional calculation and material search. Thirdly, we invent a model that incorporates the auto-encoder with the recurrent neural network, which can find a compact representation of the entire RNN model. This algorithm can find its application in other occasions of model compression and knowledge transfer.

Conflict of interest

The authors declare that they have no conflict of interest.

Acknowledgments

This work was financially supported by the National Key Research and Development Program of China (2016YFA0301600) and the National Natural Science Foundation of China (11734010). CW acknowledges the support of the China Scholarship Council. YZY acknowledges the stimulating discussion with Da Xiao, Lei Ma and Mingli Yuan in the 2017 and 2018 Swarma Club Kaifeng Research Camp.

Author contributions

All authors discussed extensively and contributed to the entire process of the project.

Appendix A. Technical details

The data for training RNN are generated by solving the "simplified" Schrödinger equation in 1d

$$V(x)\psi(x)=\partial_x^2\psi(x). \tag{A.1}$$

x labels the position in 1D. The potential begins at $x=0$ and $V(x_i)=V_i$ for $x_i\equiv ia$ where $a=0.1$ is a short range cut-off. We define $k_i=\sqrt{-V_i}$, then the wave function should take the form of $\psi(x)=A_i\sin(k_ix)+B_i\cos(k_ix)$ for $x_i\leqslant x<x_{i+1}$. Matching the wave function and its derivative will give the relations,

$$k_{i+1}A_{i+1} = A_i(k_{i+1}\sin(k_i x_i)\sin(k_{i+1}x_i) + k_i\cos(k_i x_i)\cos(k_{i+1}x_i)) + B_i(k_{i+1}\cos(k_i x_i)\sin(k_{i+1}x_i) - k_i\sin(k_i x_i)\cos(k_{i+1}x_i)), \tag{A.2}$$

$$k_{i+1}B_{i+1} = B_i(k_i\sin(k_i x_i)\sin(k_{i+1}x_i) + k_{i+1}\cos(k_i x_i)\cos(k_{i+1}x_i)) + A_i(k_{i+1}\sin(k_i x_i)\cos(k_{i+1}x_i) - k_i\cos(k_i x_i)\sin(k_{i+1}x_i)). \tag{A.3}$$

With these relations, we can solve all the $A_i,\ B_i$ starting from a fixed initial condition $A_0 = 1, B_0 = 1$, hence we can construct the wave function $\psi(x)$. Finally the density at x_i is given by

$$\rho_i = \psi(x_i)^2. \tag{A.4}$$

In summary, each data is generated in following steps:

1. Set $V_1 = -1$ and the rest $V_i = -2\ \times\ \mathrm{rand} - R$. Where rand is a random number uniformly distributed in $[0, 1]$ for each V_i, and R is a random number uniformly distributed in $[0, 1]$ which is the same for each sequence. We use R to randomly shift the energy scale for each data.
2. Make the potential V_i more smooth by performing a flatten operation, $V_{i+1} = 0.5\ \times (V_i + V_{i+1})$, for q times, where q is a random integer between 1 and 20.
3. Get the density sequence ρ_i for this potential by solving Eq. (A.2), (A.3) and using Eq. (A.4).

In practice, we collect 15,000 data, 10,000 of them used for training and 5,000 of them are used for validation.

While the potential data for RAE are generated in the same way as for RNN, and the hidden state h_i are collected by evolving the trained RNN. We collect 15,000 data, 10,000 of them are used for training and 5,000 of them are used for the validation.

We elaborate on the details of our training process. For the RNN based on Taylor expansion, we cut off the expansion at power $n_W = 2$, and consider the hidden space dimension d from 1 to 6. Taking $d = 6$ as an example, the initial $h_0 = (1, 1, 1, 1, 1, 1)$ and the vector $p = (p_1, 0, 0, 0, 0, 0)$ without loss of generality, the parameter p_1 is set to be 1 initially. We initialize the coefficient matrices $W^{(n)}$ to

$$W^{(0)} = \mathbf{1}_{d\times d} + \frac{0.01}{d}\mathrm{randn}_{d\times d},\ (\text{for } n = 0),$$
$$W^{(n)} = \frac{0.01}{d}\mathrm{randn}_{d\times d},\ (\text{for } n > 0), \tag{A.5}$$

where $\mathbf{1}_{d\times d}$ stands for the $d \times d$ dimensional identity matrix and $\mathrm{randn}_{d\times d}$ stands for the $d \times d$ dimensional random matrix whose elements follow independent Gaussian distributions (with unit variance and zero mean). We use the ADAM optimizer with learning rate 0.0002. The mini-batch size is 5. The training window is from $i = 5$ to $i = 55$.

For the RAE network, the encoder is a feedforward network of $d = 6 \to 100 \to \mathrm{ramp} \to \tilde{d}$ structure and the decoder is also a feedforward network of $\tilde{d} \to 100 \to \mathrm{ramp} \to d = 6$ structure. We use the ADAM optimizer with learning rate 0.001. The mini-batch size is 5. The training window is from $i = 5$ to $i = 60$.

Appendix B. Analysis of RNN translator loss

The RNN translator may not be able to formulate physical laws in the most parsimonious language. The hidden state of the RNN may contain redundant information. In fact, there is an analytically tractable limit where we can explicitly demonstrate this possibility. For example, the RNN may try to capture the differential equation for the density profile directly, instead of that for the quantum wave function. To simplify the analysis, let us take $\hbar^2/(2ma^2)$ as our energy unit and define the potential energy with respect to the single-particle energy level, then the Schrödinger equation for the BEC wave function $\psi(x)$ takes a rather simple form of $\partial_x^2\psi(x) = V(x)\psi(x)$. However, in terms of the density profile $\rho(x) = |\psi(x)|^2$, the Schrödinger equation implies

$$\partial_x \begin{bmatrix}\rho(x)\\ \eta(x)\\ \xi(x)\end{bmatrix} = \begin{bmatrix}0 & 2 & 0\\ V(x) & 0 & 1\\ 0 & 2V(x) & 0\end{bmatrix}\begin{bmatrix}\rho(x)\\ \eta(x)\\ \xi(x)\end{bmatrix}, \tag{B.1}$$

where $\eta(x) = \mathrm{Re}\psi^*(x)\partial_x\psi(x)$ and $\xi(x) = |\partial_x\psi(x)|^2$ are two other real profiles that combine with $\rho(x)$ to form a system of linear differential equations. The recurrent rule for such a system lies within the description power of our RNN architecture. If the RNN choose to identify its hidden state as $h_i = [\rho(x_i), \eta(x_i), \xi(x_i)]^\intercal$, the following parameters will allow it to model Eq. (B.1) with good accuracy to the first order in a:

$$W^{(0)} = \begin{bmatrix}1 & 2a & 0\\ 0 & 1 & a\\ 0 & 0 & 1\end{bmatrix}, W^{(1)} = \begin{bmatrix}0 & 0 & 0\\ a & 0 & 0\\ 0 & 2a & 0\end{bmatrix}, p = \begin{bmatrix}1\\ 0\\ 0\end{bmatrix}. \tag{B.2}$$

This theoretical construction at least provides us a base RNN that demonstrates why the proposed architecture could work in principle. The performance can be further improved by relaxing the parameters from this idea limit or by enlarging the hidden state dimension d.

However, what is the minimum hidden state dimension d (in terms of real variables) for the RNN to function well in the potential-to-density mapping? Can the RNN discover that the quantum wave function $\psi(x)$ could provide a more parsimonious description, which only requires two real variables $\mathrm{Re}\psi(x)$ and $\mathrm{Im}\psi(x)$ to parameterize? To answer these questions, we train the RNN translator under different hidden state dimensions d. As shown in Fig. B1, we observe that the loss $\mathcal{L}_{\mathrm{RNN}}$ only drops significantly if $d \geqslant 3$, implying that the RNN is unable to realize the more efficient ($d = 2$) wave function description. For the $d = 3$ case, as we read out the hidden states h_i at each step, we find that they indeed correspond to the vector $[\rho(x_i), \eta(x_i), \xi(x_i)]^\intercal$ up to specific linear transformation (depending on the random initialization of the model parameters), confirming that the RNN indeed works like the base model Eq. (B.2). From this example, we see that the RNN could develop legitimate and predictive rules of physics, such as Eq. (B.1), from the observation data. It tends to work directly with the variables present in the observation data to get the job done. Sometimes the rules it found can work well enough that the RNN may not have the motivation to develop higher-level concepts like quantum wave functions.

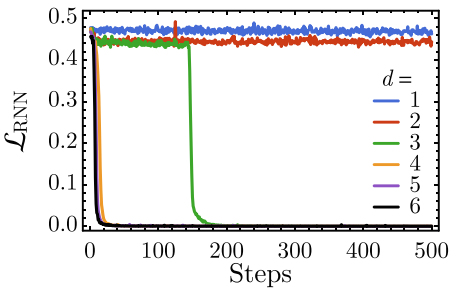

Fig. B1. (Color online) The RNN translator loss $\mathcal{L}_{\mathrm{RNN}}$ (on the test data set) vs. the training steps, for different hidden state dimensions $d = 1, 2, \cdots, 6$. The RNN is only able to master the potential-to-density mapping for $d \geqslant 3$.

References

[1] Carifio J, Halverson J, Krioukov D, et al. Machine learning in the string landscape. J High Energy Phys 2017;9:157.
[2] Koch-Janusz M, Ringel Z. Mutual information, neural networks and the renormalization group. Nat Phys 2018;14:578.
[3] You Y-Z, Yang Z, Qi X-L. Machine learning spatial geometry from entanglement features. Phys Rev B 2018;97:045153.
[4] Hashimoto K, Sugishita S, Tanaka A, et al. Deep learning and the AdS/CFT correspondence. Phys Rev D 2018;98:046019.
[5] Torlai G, Melko RG. Learning thermodynamics with Boltzmann machines. Phys Rev B 2016;94:165134.
[6] Wang L. Discovering phase transitions with unsupervised learning. Phys Rev B 2016;94:195105.
[7] Carrasquilla J, Melko RG. Machine learning phases of matter. Nat Phys 2017;13:431.
[8] van Nieuwenburg EPL, Liu Y-H, Huber SD. Learning phase transitions by confusion. Nat Phys 2017;13:435.
[9] Zhang Y, Kim E-A. Quantum loop topography for machine learning. Phys Rev Lett 2017;118:216401.
[10] Wang C, Zhai H. Machine learning of frustrated classical spin models. I. Principal component analysis. Phys Rev B 2017;96:144432.
[11] Wang C, Zhai H. Machine learning of frustrated classical spin models (II): kernel principal component analysis. Front Phys 2018;13:130507.
[12] Zhang P, Shen H, Zhai H. Machine learning topological invariants with neural networks. Phys Rev Lett 2018;120:066401.
[13] Iten R, Metger T, Wilming H, et al. Discovering physical concepts with neural networks. arXiv:1807.10300, 2018.
[14] Wu T, Tegmark M. Toward an AI physicist for unsupervised learning. arXiv:1810.10525, 2018.
[15] Kalchbrenner N, Blunsom P. Recurrent continuous translation models. Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics; 2013. p. 1700–9.
[16] Sutskever I, Vinyals O, Le QV. Sequence to sequence learning with neural networks. In: Ghahramani Z, Welling M, Cortes C, et al., editors. Advances in Neural Information Processing Systems, vol. 27. Curran Associates Inc; 2014. p. 3104–12.
[17] Cho K, van Merrienboer B, Gulcehre C, et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv:1406.1078, 2014.
[18] Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate. arXiv:1409.0473, 2014.
[19] Goodfellow I, Bengio Y, Courville A. Deep learning. MIT Press; 2016.
[20] Neubig G. Neural machine translation and sequence-to-sequence models: a tutorial. arXiv:1703.01619, 2017.
[21] Lye JE, Fallani L, Modugno M, et al. Bose-Einstein condensate in a random potential. Phys Rev Lett 2005;95:070401.
[22] Ma C, Wang J, Weinan E. Model reduction with memory and the machine learning of dynamical systems. arXiv:1808.04258, 2018.
[23] Banchi L, Grant E, Rocchetto A, et al. Modelling non-markovian quantum processes with recurrent neural networks. arXiv:1808.01374, 2018.
[24] Bengio Y, Courville A, Vincent P. Representation learning: a review and new perspectives. IEEE Trans Pattern Anal Mach Intelligence, 2160-9292 2013;35:1798.
[25] Kingma DP, Welling M. Auto-encoding variational bayes. arXiv:1312.6114, 2013.
[26] Mirowski P, Ranzato M, LeCun Y. Dynamic auto-encoders for semantic indexing. Proceedings of the NIPS 2010 Workshop on Deep Learning, vol. 2.
[27] Kohn W, Sham LJ. Self-consistent equations including exchange and correlation effects. Phys Rev 1965;140:A1133.
[28] Snyder JC, Rupp M, Hansen K, et al. Finding density functionals with machine learning. Phys Rev Lett 2012;108:253002.
[29] Li L, Snyder JC, Pelaschier IM, et al. Understanding machine-learned density functionals. arXiv:1404.1333, 2014.
[30] Li L, Baker TE, White SR, et al. Pure density functional for strong correlation and the thermodynamic limit from machine learning. Phys Rev B 2016;94:245129.
[31] Brockherde F, Vogt L, Li L, et al. Bypassing the Kohn-Sham equations with machine learning. Nat Commun 2017;8:872.
[32] Khoo Y, Lu J, Ying L. Solving parametric PDE problems with artificial neural networks. arXiv:1707.03351, 2017.

Ce Wang is a post-doc in Institute for Advanced Study of Tsinghua University. He got both his Bachelor and Doctor degrees in Tsinghua University. His work mainly focuses on machine learning applications in quantum physics.

Hui Zhai is full Professor in Institute for Advanced Study of Tsinghua University. He got both his Bachelor and Doctor degrees in Tsinghua University. His work mainly focuses on the theory of quantum matters, including cold atomic gases, holographic quantum matters and machine learning applications in quantum physics.

Yi-Zhuang You is an Assistant Professor of Physics at University of California San Diego. He works in theoretical condensed matter physics, including strongly correlated systems, topological quantum phases of matter, quantum information dynamics, holographic duality, and machine learning applied to quantum physics and holography.