

Interpretable Quantum Advantage in Neural Sequence Learning

Eric R. Anschuetz^{1,*}, Hong-Ye Hu^{2,3,4}, Jin-Long Huang², and Xun Gao^{4,†}

¹MIT Center for Theoretical Physics, 77 Massachusetts Avenue, Cambridge, Massachusetts 02139, USA

²Department of Physics, University of California San Diego, 9500 Gilman Drive, La Jolla, California 92093, USA

³Harvard Quantum Initiative, Harvard University, 17 Oxford Street, Cambridge, Massachusetts 02138, USA

⁴Department of Physics, Harvard University, 17 Oxford Street, Cambridge, Massachusetts 02138, USA



(Received 7 November 2022; accepted 6 March 2023; published 8 June 2023)

Quantum neural networks have been widely studied in recent years, given their potential practical utility and recent results regarding their ability to efficiently express certain classical data. However, analytic results to date rely on assumptions and arguments from complexity theory. Because of this, there is little intuition as to the source of the expressive power of quantum neural networks or for which classes of classical data any advantage can be reasonably expected to hold. Here, we study the relative expressive power between a broad class of neural network sequence models and a class of recurrent models based on Gaussian operations with non-Gaussian measurements. We explicitly show that quantum contextuality is the source of an unconditional memory separation in the expressivity of the two model classes. We use this intuition to study the relative performance of our introduced model on a standard translation data set exhibiting linguistic contextuality. In doing so, we demonstrate that our introduced quantum models are able to outperform state-of-the-art classical models even in practice.

DOI: [10.1103/PRXQuantum.4.020338](https://doi.org/10.1103/PRXQuantum.4.020338)

I. INTRODUCTION

The field of quantum information processing has reached a watershed in recent years, with the first demonstrations of quantum processors performing tasks on the verge of classical intractability [1–6]. Spurred on by these recent experimental developments, there has been a push for finding algorithms that can be performed using either near-term quantum devices, or early error-corrected ones. One of the leading candidates for such algorithms are *quantum machine-learning* (QML) algorithms, where training can be offloaded to a classical computer working in conjunction with a quantum computer, potentially minimizing the coherence requirements of the quantum device [7–14]. These algorithms are also motivated by the ability of quantum systems to naturally represent complex probability distributions that are believed to be difficult to represent classically [15–18], with many proposed architectures for such quantum models [9, 19–21].

However, any proof of advantage in the expressivity of these models over classical models relies on results from computational complexity theory, themselves conditional on complexity theoretic assumptions [15–18, 22]. As the proofs of separation are abstract, it is unclear what realistic classical data sets one should expect a separation to hold in practice. Also, due to the universality of many of these models, they are very likely to be untrainable due to phenomena such as barren plateaus [23–26] and bad local minima [27–29] present in their loss landscapes. Because of these concerns, it has become increasingly clear that quantum models should be carefully constructed to fit the task at hand. Above all else, the *interpretability* of any expressivity separation achieved by a QML model has become increasingly important. Interpretability reveals which features of quantum mechanics yield more expressive models compared to classical models and, armed with this knowledge, allows one to find classes of problems where a practical quantum advantage on real data is potentially achievable.

Wishing to construct a model with an interpretable quantum advantage, we here focus on *sequence-to-sequence* learning tasks [30], and consider a quantization of *linear recurrent neural networks* (LRNNs) [31]. Classical LRNNs are recurrent neural networks with only linear activation functions. Such models can equivalently be considered a classical dynamical system governed by quadratic Hamiltonian evolution in the canonical variables (\mathbf{q}, \mathbf{p}) . By

*eans@mit.edu

†xungao@g.harvard.edu

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

lifting these canonical variables to operators $(\hat{\mathbf{q}}, \hat{\mathbf{p}})$ that satisfy the canonical commutation relations (in units where $\hbar = 1/2$)

$$[\hat{q}_j, \hat{p}_k] = \frac{i}{2} \delta_{jk}, \quad (1)$$

we arrive at a continuous-variable (CV) quantum model where time evolution on an eigenstate of the canonical operators is performed under a quadratic Hamiltonian. To measure properties of the state of the system, the most natural choice is to perform *homodyne measurement*; that is, measure linear combinations of the canonical operators \hat{q}_j and \hat{p}_k . This yields a quantum generative model where all operations are Gaussian. However, as all operations, initial states, and measurements are Gaussian, there are efficient Wigner function–based simulations of sampling from such a system [32]. In other words, such models on n modes are equivalent to deep belief networks [33]—a class of commonly used classical models—with $2n$ latent variables.

Instead, we extend this model slightly further by allowing for measurements of the canonical operators *modulo* 2π , beginning in an eigenstate of periodic functions of the canonical operators [34,35]. We call this introduced class of models *contextual recurrent neural networks* (CRNNs). Our main result is that CRNNs are more memory efficient at expressing certain distributions than essentially all trainable classical sequence models—*independent of their internal, latent representations—even though CRNNs are not universal for CV quantum computation. Concretely, we show unconditionally that there exists a class of CRNNs with $O(n)$ qumodes that can express certain distributions that no “reasonable” (which we later describe) classical model is able to represent without an $\Omega(n^2)$ -dimensional latent space. Though this is only a quadratic separation in memory, the time complexity of inference for classical models is typically superlinear in the model size [31,36–38], yielding a superquadratic time separation. As we show a memory (rather than a time) separation, our results also potentially point to a practical *generalization* advantage for CRNNs, as smaller models tend to generalize better than larger models due to formalized versions of Occam’s razor [39].*

Moreover, we are able to show directly that this quantum advantage is due to quantum contextuality [35,40–43] present in our quantum model. Previously, quantum contextuality was known to be the resource for the expressive power of a certain class of quantized Bayesian networks [44]. Our results show that this resource can be used to separate quantum models even from neural networks, which are exponentially more efficient than generic Bayesian networks. Intuitively, quantum contextuality is the statement that quantum measurement results depend on which

measurements were previously performed, even if the measurements in question commute. In other words, quantum contextuality is the statement that the measurement of quantum observables cannot be thought of as the revealing of preexisting classical values for the observables. Here, we give a proof of the intuition that reasonable classical models cannot get around the need to “memorize” the measurement context of given observables, which is what yields the quadratic memory separation between the quantum and classical models.

Qualitatively, quantum contextuality is similar to the linguistic contextuality present in sentences. Namely, the meaning of a given word in a sentence depends heavily on other words in the sentence, and without this context has no fixed, single meaning. Inspired by this, we also test our constructed model against state-of-the-art classical models on a real-world translation task to investigate whether the ability of the quantum model to store information in its measurement context yields a practical quantum advantage in modeling the long-time correlations present in typical sequential data sets. In particular, we evaluate the performance of a LRNN [45], a RNN with gated recurrent units (GRU RNN) [37], a transformer [38], a Gaussian model, and our introduced contextual model on a standard Spanish-to-English data set [46]. We show that our introduced contextual model achieves better translation performance compared to all other models at each model size we consider. This separation holds even when the online models are constrained to have a similar (and where possible, the same) number of trainable parameters in each recurrent cell.

Our methods provide a novel strategy for designing QML models: through the quantization of simple classical machine-learning models with some minimal quantum extension. Though such models are most likely unable to outperform state-of-the-art classical machine-learning models on *all* tasks, the intuition gleaned from the simplicity of the quantum models gives guidance as to which problems the quantum models may outperform classical models on. Furthermore, the simplicity of the quantum models may circumvent the recent deluge of untrainability results of general quantum models [23–29] as it is known that there exists a trade-off between the trainability of such architectures and their generality [47]. Finally, as such models are restricted in their allowed operations, they are more amenable to experimental implementation than completely generic quantum models.

II. CLASSICAL AND QUANTUM NEURAL SEQUENCE LEARNING

A. Classical sequence learning

Sequence-to-sequence or *sequence learning* [30] is the approximation of some given conditional distribution $p(\mathbf{y} | \mathbf{x})$ with a model distribution $q(\mathbf{y} | \mathbf{x})$. This framework

encompasses sentence translation tasks [30], speech recognition [48], image captioning [49], and many more practical problems.

Sequence modeling today is typically performed using neural network–based generative models, or *neural sequence models*. Generally, these models are parameterized functions that take as input the sequence \mathbf{x} and output a sample from the conditional distribution $q(\mathbf{y} | \mathbf{x})$. The parameters of these functions are trained to minimize an appropriate loss function, such as the (forward) empirical cross entropy

$$\hat{H}(p, q) = -\frac{1}{|\mathcal{T}|} \sum_{(\mathbf{x}, \mathbf{y}) \in \mathcal{T}} p(\mathbf{y} | \mathbf{x}) \log(q(\mathbf{y} | \mathbf{x})), \quad (2)$$

where $\mathcal{T} = \{(\mathbf{x}_i, \mathbf{y}_i)\}$ are samples from $p(\mathbf{x}, \mathbf{y})$. The backward empirical cross entropy is similarly defined, with $p \leftrightarrow q$. Note that a model with support on an incorrect translation (i.e., $q \neq 0, p = 0$) yields an infinite backward cross entropy, and a model failing to have support on a correct translation (i.e., $p \neq 0, q = 0$) yields an infinite forward cross entropy.

To maintain a resource scaling independent of the input sequence length, neural sequence models usually fall into one of two classes: *online sequence models* (also known as *autoregressive sequence models*) [31,36,37] or *encoder-decoder models* (which include state-of-the-art sequence learning architectures, such as transformers) [30,38]. We focus on online models here, and discuss encoder-decoder models in more detail in Appendix A.

In online models, input tokens \mathbf{x}_i are translated in sequence to output tokens \mathbf{y}_i via functions \mathcal{F}_i . An unobserved internal memory (or *latent space*) L shared between time steps allows the model to represent long-range correlations in the data. A diagram of the general form of online models is given in Fig. 1(a). Generally, there are no restrictions on the forms of \mathcal{F}_i , though most neural sequence models are composed of simple smooth (or almost everywhere smooth) functions out of training considerations [31,36–38]. Here, we generalize from the typical smoothness constraints and consider *locally Lipschitz* maps.

Assuming that the codomain of \mathcal{F}_i is \mathbb{R}^m , all maps that are almost everywhere differentiable with locally bounded Jacobian norm are locally Lipschitz [50]. Realistically, then, locally Lipschitz models can be thought of as all models trainable using gradient-based methods. Equivalently, they can be thought of as models not infinitely sensitive to infinitesimal changes in their inputs. This includes all models with standard nonlinearities, including those with rectified linear unit (ReLU), hyperbolic tangent, and sigmoid activation functions. Note that this condition is much weaker than a *globally Lipschitz* constraint. We give a formal definition of local Lipschitzness in Appendix A.

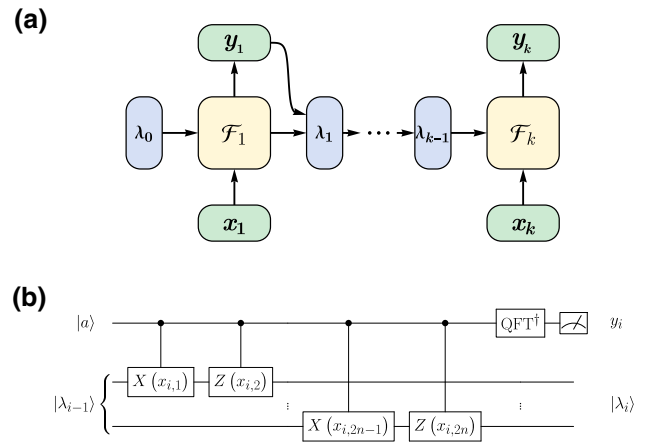


FIG. 1. (a) An online neural sequence model. The model autoregressively takes input tokens \mathbf{x}_i and output decoded tokens \mathbf{y}_i with the map \mathcal{F}_i . The model also has an unobserved internal memory with state $\lambda_{i-1} \in L$ that \mathcal{F}_i can depend on. When the model is quantized to a CRNN, the n -dimensional space of λ_i is promoted to the Hilbert space of n qumode states $|\lambda_i\rangle$. (b) An implementation of a phase estimation circuit for CV Pauli operators, which forms the recurrent cell of the CRNN we use to prove our separations. Here, $|a\rangle$ is a fixed ancilla state and “QFT” represents the quantum Fourier transform. Formally, if $|a\rangle$ is a GKP state, this circuit allows for infinite precision measurements. In practice, $|a\rangle$ can be a tensor product of a constant number of qubit $|+\rangle$ states for finite precision phase estimation.

Though neural networks are often described as functions of real-valued inputs, in practice they are implemented at finite precision. We emphasize that where we analytically consider such networks here—such as in Sec. III—we consider the formal description of neural networks, which assumes infinite precision. Our numerical experiments in Sec. IV, however, give evidence that our analytic results also hold in the finite precision regime. We discuss this in more detail in Appendix C.

B. Contextual recurrent neural networks

We now consider a quantization of a simple online model. Generally, online models can be interpreted as a classical dynamical process, where queries \mathbf{x}_i are made to a physical system described by the latent state λ_{i-1} , yielding a result \mathbf{y}_i and transforming the latent state $\lambda_{i-1} \mapsto \lambda_i$ [see Fig. 1(a)]. For *LRNNs*, this can be interpreted as the physical process of querying properties of an underlying system described by λ_i undergoing Hamiltonian evolution under a quadratic Hamiltonian; this can be seen straightforwardly from Hamilton’s equations and the linearity of the model. When quantizing the canonical position and momentum variables to operators satisfying the canonical commutation relations, such a model can then be interpreted as performing sequential measurements on a system undergoing

evolution via *Gaussian operations*. When these measurements are restricted to homodyne measurements and all inputs are Gaussian states, this process can be simulated classically with memory linear in the number of modes of the Gaussian system [32]. We minimally extend this, and allow for *non-Gaussian measurements*. In particular, we are here interested in measuring via phase estimation the CV analogues of the Pauli operators [51] (in units where $\hbar = 1/2$):

$$X_i(a) = e^{-2ia\hat{p}_i}, \quad Z_i(a) = e^{2ia\hat{q}_i}. \quad (3)$$

We also promote the initial state of the network to a *Gottesman–Kitaev–Preskill (GKP) state* [34], which is an eigenstate of CV Pauli operators. We call a recurrent online model beginning in a GKP state, with cell that takes as input \mathbf{x}_i a description of a CV Pauli operator and returns its measurement result y_i , a *contextual recurrent neural network (CRNN)*.

This measurement can formally be performed at infinite precision using Gaussian operations and homodyne measurement with fixed ancilla GKP states [34,52]. A circuit description of this is given in Fig. 1(b), where $|a\rangle$ is a uniform superposition over squeezed states $|s\rangle$ with $\hat{q}|s\rangle = q|s\rangle$, where $q \equiv 0 \pmod{2\pi}$. When performed sequentially on an initial GKP state, these measurements are what we consider when we compare in Sec. III CRNNs against the infinite precision classical neural networks described in Sec. II A. In this scenario, the model is not universal for CV quantum computation, even when additional Gaussian operations within the latent space are added [53]. Counterintuitively, when the initial state is the vacuum state or a finitely squeezed GKP state, the model *is* universal [53,54]; this suggests a potential superpolynomial advantage in the expressive power and the time complexity of inference when implemented at finite precision. We discuss this in more detail in Appendix C.

Just as in the classical case, one can consider a finite precision approximation of these measurements. In this scenario, phase estimation using ancilla qubits can be performed for each measurement [55]. We discuss proposals for the experimental implementation of such a measurement in Appendix C. In general, parameterized Gaussian operations can be included within each recurrent cell to yield a *trainable CRNN*. This is a special case of the CV neural networks considered in Ref. [14], which also considered the training of such networks. For our expressivity separations, however, we consider the fixed CRNN instance given in Fig. 1(b).

For our purposes, these measurements are important as CV Pauli operators exhibit *quantum contextuality* [35], in complete analogy with the contextuality present in qubit Pauli operators [43]. Quantum contextuality is the statement that no definite classical values can be assigned to quantum operators, even when the measured operators in

TABLE I. An example of CV quantum contextuality using a Mermin-Peres magic square [43], with CV Pauli operators $X_i(a), Z_i(a)$ generated by $-2ia\hat{p}_i, 2ia\hat{q}_i$, respectively. For any real $\alpha \neq 0$, all operators in each row and column commute. Additionally, the product of each row and column is the identity operator, except for the final column, which gives -1 . Thus, definite classical values cannot be assigned to each operator without yielding a contradiction.

$X_1(\alpha)$	$X_2(\alpha)$	$X_1(\alpha)^\dagger X_2(\alpha)^\dagger$ $-X_1(\alpha)Z_1(\pi/2\alpha)$
$X_1(\alpha)^\dagger Z_2(\pi/2\alpha)^\dagger$	$Z_1(\pi/2\alpha)^\dagger X_2(\alpha)^\dagger$	$X_2(\alpha)Z_2(\pi/2\alpha)$
$Z_2(\pi/2\alpha)$	$Z_1(\pi/2\alpha)$	$Z_1(\pi/2\alpha)^\dagger Z_2(\pi/2\alpha)^\dagger$

any given measurement scenario commute. For an example of this phenomenon, see Table I; there is no consistent assignment of classical values to each operator in the table for any real $\alpha \neq 0$.

C. Stabilizer measurement translation

We now focus on a classical sequence learning task that is naturally performed by the introduced CRNN. In particular, we consider the (k, n) *stabilizer measurement translation task*, parameterized by k and n . Leaving the formal definition for Appendix B, we give an informal definition here. We use the terminology of Fig. 1 for clarity.

Definition 1 ((k, n) stabilizer measurement translation task, informal). Given a k -long sequence of classical descriptions \mathbf{x}_i of CV Pauli operators on n modes, output a sequence of measurement outcomes y_i that is consistent with measuring these operators sequentially on a fixed GKP state $|\lambda_0\rangle$.

As described in Sec. II B, such measurement sequences can display nontrivial correlations due to quantum contextuality. Note that this task is distinct from the measurement of position and momentum operators. Here, we require the measurement of linear combinations of position and momentum operators *modulo* 2π , as we are measuring the phases of operators generated by position and momentum. This can be done using the CRNN cell described in Sec. II B. We consider in Appendix B a slight generalization of this task, though here we consider Definition 1 with its fixed GKP initial state for simplicity.

III. BOUNDS ON STABILIZER MEASUREMENT TRANSLATION

We now give statements and proof sketches of our main results, which are lower bounds on the performance of classical models in performing the stabilizer measurement translation task described in Sec. II C. This will give an expressivity separation between the classical and quantum sequence models.

For discrete models, quantum contextuality was the key resource for showing a separation in expressivity between classical and quantum models [44]. Using different proof techniques, we here show that quantum contextuality is also the resource giving the separation between continuous classical and quantum models with infinite-dimensional Hilbert spaces. To do this, we specialize to two classes of models: online neural sequence models and encoder-decoder models (which include state-of-the-art models such as seq2seq models [30] and transformers [38]). Here, we focus on the memory separation between CRNNs and classical online neural sequence models, and discuss a similar separation against encoder-decoder models in Appendix B. We also there formulate a general statement on the classical efficiency of simulating CV Pauli measurements on an initial GKP state, similar in spirit to the fact that the Gottesman–Knill theorem [56,57] is optimal for qubit stabilizer simulation [58].

Our main result can be informally stated as the following theorem (with the full statement and proof left to Appendix B). Note that, as discussed in Sec. II C, a CRNN can perform the stabilizer measurement translation task with n qumodes of memory.

Theorem 1 (Online stabilizer measurement translation memory lower bound, informal). *Consider a locally Lipschitz online model with latent space L . If $\dim(L) < n(n-3)/2$, this model cannot achieve a finite backward cross entropy on the $(n+2, n)$ stabilizer measurement translation task.*

Proof sketch. The strategy of our proof is to show that, when the dimension of L is less than $n(n-3)/2$, the model must map an embedded submanifold K of the space of the first n inputs to the same point in L ; in other words, the model loses the ability to distinguish between inputs in K . The nontrivial aspect of this proof is to demonstrate that such a K exists, where distinct points in K yield different translations. As the model is unable to distinguish between points in K , this then demonstrates that the model will get a translation incorrect, corresponding to an infinite backward cross entropy on the stabilizer measurement translation task. This is equivalent to demonstrating that the quantum mechanical processes being described by points in K yield quantum states that are single-shot distinguishable.

To demonstrate that such a K exists, we use the local Lipschitzness of the model and the constant rank theorem [59]. This then implies that the map \mathcal{F} —given by the n -fold composition of the \mathcal{F}_i as shown in Fig. 1(a)—locally induces a fiber bundle on the input space (as shown in Fig. 2), where \mathcal{F} can be considered a projection onto the base manifold of this induced fiber bundle. We slightly abuse notation in the remainder of this proof sketch,

and conflate the first n inputs (and their associated outputs) with the quantum state that would arise from this measurement sequence; this same labeling is used in Fig. 2.

We consider a fiber of this fiber bundle, with the goal of proving that there exist points in this fiber that are single-shot distinguishable. We show in Appendix B that the dimension of this fiber is large enough such that there exist three states in this fiber with stabilizers that exhibit quantum contextuality. We claim (and prove in Appendix B) that, due to the presence of quantum contextuality in these stabilizers, these states have a distinguishing measurement sequence of length two. When performing this distinguishing measurement sequence, then, the model must give the incorrect measurement results for one of the three states, giving the lower bound on classical simulation. It is easy to see from Eq. (2) that this yields both an infinite backward cross entropy when these sequences are in data set \mathcal{T} .

As a simple example of this phenomenon, assume that three states $|\psi_1\rangle, |\psi_2\rangle, |\psi_3\rangle$ with classical representations in the same fiber are respectively stabilized by the rows of Table I for some real $\alpha \neq 0$. As $|\psi_3\rangle$ is stabilized by $Z_1(\pi/2\alpha)^\dagger Z_2(\pi/2\alpha)^\dagger$, upon measuring this operator, the measurement result is constrained to be 1 for the simulation to be accurate. The postmeasurement state of $|\psi_1\rangle$ is then stabilized by $X_1(\alpha)X_2(\alpha)$ and $Z_1(\pi/2\alpha)^\dagger Z_2(\pi/2\alpha)^\dagger$. In particular, it is also stabilized by $X_1(\alpha)Z_1(\pi/2\alpha)^\dagger X_2(\alpha)Z_2(\pi/2\alpha)^\dagger$. Conversely, the postmeasurement state of $|\psi_2\rangle$ is stabilized by $-X_1(\alpha)Z_1(\pi/2\alpha)^\dagger X_2(\alpha)Z_2(\pi/2\alpha)^\dagger$. Thus, then measuring $X_1(\alpha)Z_1(\pi/2\alpha)^\dagger X_2(\alpha)Z_2(\pi/2\alpha)^\dagger$ gives an incorrect translation for one of these states. This measurement sequence is single shot, as only a single copy of the state being measured is used. ■

Our results show that there is a general n versus $\Omega(n^2)$ bound in the memory requirements of contextual and classical models performing the stabilizer measurement translation task. In practice, this can yield an even greater separation in time complexity for given implementations of these models, as the time complexity of inference using classical models is typically superlinear in the model size. We discuss this in more detail in Appendix E 3.

IV. NUMERICAL EXPERIMENTS

We now showcase the practical benefit of finding an interpretable advantage in the expressivity of our quantum model. Namely, it is able to give us intuition as to which data sets—beyond the constructed data set used in our proof—a CRNN may outperform classical machine-learning models on. As previously discussed, the contextuality present in quantum operators behaves qualitatively similar to the linguistic contextuality present in language. That is, words can have one of many meanings, and their exact definition only becomes apparent when considering

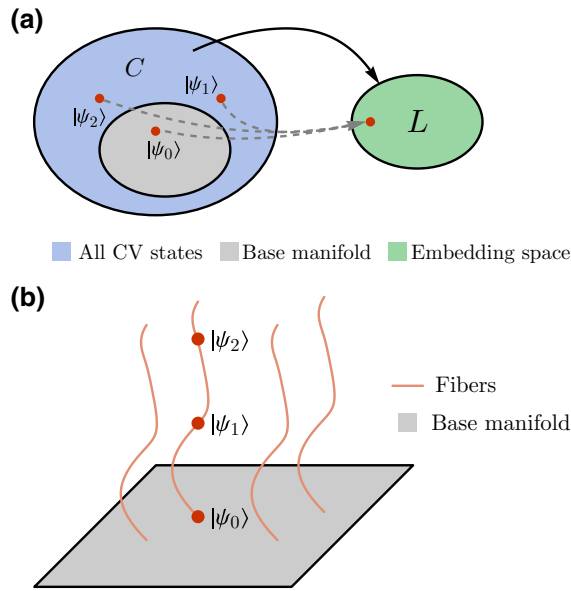


FIG. 2. (a) A schematic of the classical model when the dimension of the latent space L (green) is less than $n(n-3)/2$, where n is the number of modes in the stabilizer measurement translation task. Here, “ $|\psi_i\rangle$ ” labels the input sequence that is composed of the stabilizers of $|\psi_i\rangle$; we discuss this labeling in more detail in our proof sketch of Theorem 1. We show that, when $\dim(L) < n(n-3)/2$, in the neighborhood of some input, only a subspace of inputs (gray) of the same dimension as L are mapped injectively. (b) A sketch of the space of inputs, with fibers locally induced by the model. The base manifold is mapped injectively to L . All points on a fiber (e.g., $|\psi_1\rangle, |\psi_2\rangle$) map to the same point as their base point (e.g., $|\psi_0\rangle$) in L . When the dimension of the fiber is large enough, we show that these states have contextual stabilizers. We then show that this implies that the states have a single-shot distinguishing measurement sequence.

their context in a sequence. This is important for translation tasks, where different meanings of a single word in one language have different translations in other languages. We here investigate whether the ability of the quantum model to store information in its measurement context yields a practical quantum advantage in modeling the long-time correlations present in a typical sequential data set.

To explore this intuition, we consider the application of a CRNN on a standard Spanish-to-English translation data set [46], with trainable Gaussian interactions within each recurrent cell. We also consider the performance of GRUs [37] in a seq2seq learning framework [30], and Gaussian models (with Gaussian measurements). Details of our numerical simulations for all of the models we consider are given in Appendix E, along with details of the architectures. We also discuss in Appendices D and E a $\Theta(n^2)$ memory classical simulation of CRNNs with n latent modes on a restricted space of Gaussian operations, which is what we use in our numerical simulations. The Gaussian and contextual models are constrained to have

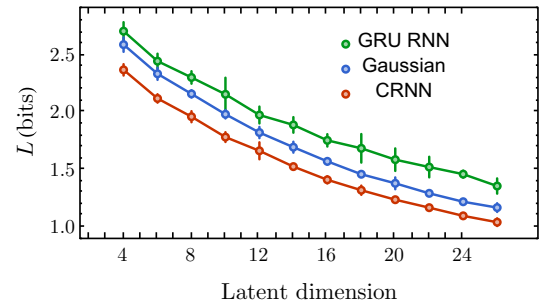


FIG. 3. The forward empirical cross entropy (L) on a test set for a Spanish-to-English translation task as a function of the model dimension n for GRU RNNs, Gaussian RNNs, and CRNNs. The models are constrained such that the Gaussian and CRNN models have an identical number of parameters. The recurrent cells of the GRU RNN and quantum models have a number of parameters within 2.5% of each other at $n = 26$. Error bars denote the standard deviations of the loss over five independent training runs.

exactly the same number of trainable parameters, and each recurrent cell of the GRU has a parameter count within 2.5% of those of the quantum models at the largest model size considered. In Fig. 3, we plot the final training performance of all of our models over five independent training runs. It is easy to see that the contextual model outperforms all models under consideration in forward empirical cross entropy at a wide range of model dimensions n . It is also apparent that training CRNNs is no more difficult than training GRUs, justifying our intuition that restricted quantum machine-learning models should be more trainable than generic models [23–29]. Random samples of translation results after training are shown in Table II.

We also compared the performance of CRNNs against classical LRNNs [45] and transformers [38]; for the latter,

TABLE II. Random samples of translation results for $n = 26$ models.

Input	“Debemos limpiar la cocina.”
Truth	“We must clean up the kitchen.”
CRNN	“We must clean the kitchen.”
GRU	“We have to turn the right address.”
Input	“Admití que estaba equivocada.”
Truth	“I admitted that I was wrong.”
CRNN	“I was wrong to say that.”
GRU	“They had a thing to be true.”
Input	“Cual es el lugar más bonito del mundo?”
Truth	“What’s the most beautiful place in the world?”
CRNN	“What’s the world largest place?”
GRU	“What’s the best of is in?”
Input	“La caja es pesada.”
Truth	“The box is heavy.”
CRNN	“The box is heavy.”
GRU	“My box is.”

we also tested the model sizes required for the quantum and classical models to achieve some fixed target in performance. We find that CRNNs substantially outperformed both LRNNs and transformers. We also verified that to attain a fixed target loss realized by a CRNN of size $n = 26$ a transformer needed a memory of dimension roughly $n(n - 3)/2$, which is the same as that proven in the setting of Theorem 1. We give details of these results in Appendix F.

V. OUTLOOK

Our results pinpoint quantum contextuality as a resource that can be used to enhance traditional machine-learning models. We achieved this by constructing a sequence learning task parameterized by n that a contextual quantum model (a CRNN) of size n is able to model, yet provably no classical neural networks of size subquadratic in n can model due to their noncontextuality. To our knowledge, this is the first unconditional proof of an expressivity separation between a quantum neural network and classical neural networks on classical data. By explicitly demonstrating that quantum contextuality is the source of this advantage, we are also able to provide intuition as to which classes of problems CRNNs are able to outperform traditional machine-learning models in solving. Our numerics confirm the intuition that CRNNs perform extremely well on problems exhibiting linguistic contextuality, such as the Spanish-to-English translation task we consider here.

The simple structure of CRNNs also allow (finite precision approximations of) them to be more amenable to potential experimental implementations when compared with completely general quantum architectures. In particular, all operations in the contextual model are Gaussian, up to the requirement for interactions with fixed ancilla states to perform the required non-Gaussian measurements. Furthermore, though we do not consider the effects of noise here in detail, a key component of CRNNs is intermediate measurement using ancilla GKP states; this procedure is an important building block of CV quantum error correction [34], and in future work we hope to study how the addition of fast classical feedback may allow for error correction in CRNNs. The restricted nature of the model may also circumvent the poor training landscapes of generic quantum neural networks [23–29], though we leave further investigation of this to future work.

We believe that the specifics of the CRNN architecture can be relaxed somewhat. Because of recent results linking non-Gaussian operations to quantum contextuality [60,61], we suspect that any non-Gaussian measurement would make a suitable replacement for the stabilizer measurements we consider here for technical reasons. We also suspect that the technical requirement that the measurements be made with infinite precision to be an artifact of the nature of our proof, which compares the quantum

architecture with infinite precision classical models. We believe that, in practice, performing phase estimation using ancilla qubits instead of GKP (or other non-Gaussian CV) states is all that is necessary for a practical separation. In fact, such a finite precision implementation may counterintuitively yield a *larger* quantum advantage, as our architecture implemented with a finitely squeezed initial Gaussian state is universal for CV quantum computation [53,54]. We discuss these two points in more detail in Appendix C.

CRNNs demonstrate that even the quantization of a very simple class of classical architectures—here, the class of LRNNs—is able to outperform a wide range of classical models on certain tasks, even if the classical models are much more powerful than LRNNs. We leave for future work the quantization of more powerful classical architectures, which may achieve a practical quantum advantage on a wider variety of tasks than we consider here.

ACKNOWLEDGMENTS

We thank Li Jing for sharing with us the code used in Ref. [45]. We also thank Honghao Fu, Pranav Gokhale, Liang Jiang, Robert Huang, Bobak T. Kiani, Seth Lloyd, Mikhail D. Lukin, and Quntao Zhuang for insightful discussions. E.R.A. is supported by the National Science Foundation Graduate Research Fellowship Program under Grant No. 4000063445, and a Lester Wolfe Fellowship and the Henry W. Kendall Fellowship Fund from M.I.T. H.Y.H. is supported by the UC Hellman Fellowship and the Harvard Quantum Initiative Fellowship. J.L.H. is supported by the Simons Collaboration on Ultra-Quantum Matter, which is a grant from the Simons Foundation (652264). X.G. is supported by the Postdoctoral Fellowship in Quantum Science of the MPHQ, the Templeton Religion Trust Grant No. TRT 0159, and by the Army Research Office under Grant No. W911NF1910302 and MURI Grant No. W911NF2010082.

APPENDIX A: BACKGROUND ON SEQUENCE LEARNING

1. Sequence learning

Sequence-to-sequence or *sequence* learning [30] is the approximation of some given conditional distribution $p(\mathbf{y} | \mathbf{x})$ with a model distribution $q(\mathbf{y} | \mathbf{x})$. This framework encompasses sentence translation tasks [30], speech recognition [48], image captioning [49], and many more practical problems.

The training and evaluation of sequence-to-sequence models is most often performed on the *forward (conditional) cross entropy*:

$$H(p, q) = - \int d\mathbf{x} \int d\mathbf{y} p(\mathbf{x}, \mathbf{y}) \log(q(\mathbf{y} | \mathbf{x})). \quad (\text{A1})$$

Here, “forward” indicates the ordering of the arguments of H ; the *backward cross entropy* is given by $p \leftrightarrow q$. Given a finite test set $\mathcal{T} = \{(\mathbf{x}_i, \mathbf{y}_i)\}$ of M points sampled from $p(\mathbf{x}, \mathbf{y})$, we can also define the forward *empirical cross entropy*

$$\hat{H}(p, q) = -\frac{1}{M} \sum_{(\mathbf{x}, \mathbf{y}) \in \mathcal{T}} p(\mathbf{y} | \mathbf{x}) \log(q(\mathbf{y} | \mathbf{x})), \quad (\text{A2})$$

with the backward empirical cross entropy once again given by $p \leftrightarrow q$.

Historically, sequence learning was performed using Bayesian networks such as hidden Markov models [62,63]. However, in recent years, the performance of these models has been eclipsed by neural network–based models.

2. Neural sequence models

Sequence modeling today is typically performed using neural network–based generative models, or *neural sequence models*. Generally, these models are parameterized functions that take as input the sequence \mathbf{x} and output a sample from the conditional distribution $p(\mathbf{y} | \mathbf{x})$; the parameters of these functions are trained to minimize an appropriate loss function, such as the empirical cross entropy of Eq. (A2).

To maintain a resource scaling independent of the input sequence length, neural sequence models usually are one of two classes: *online sequence models* (also known as

autoregressive sequence models) [31,36,37] or *encoder-decoder models* [30,38]. Examples of both are given in Fig. 4. In the former class of models, input tokens \mathbf{x}_i are translated in sequence to output tokens \mathbf{y}_i via functions \mathcal{F}_i . An unobserved internal memory (or *latent space*) shared between time steps allows the model to represent long-range correlations in the data.

In the latter of these models, an encoder \mathcal{E} maps the input sequence \mathbf{x} to a latent space representation $\lambda \in L$; then, a decoder \mathcal{D} transforms this representation to the output sentence \mathbf{y} . The advantage of encoder-decoder models over generic representations of $p(\mathbf{y} | \mathbf{x})$ is the improved time complexity when considering a lower-dimensional representation λ of \mathbf{x} . When the encoder map is trivial (i.e., when L is congruent to the input space and \mathcal{E} is the identity), then no compression occurs, and the model is equivalent to a general representation of $p(\mathbf{y} | \mathbf{x})$ given by \mathcal{D} .

Generally, there are no restrictions on the forms of \mathcal{F}_i , \mathcal{E} , or \mathcal{D} , though most neural sequence models are composed of simple smooth (or almost everywhere smooth) functions out of training considerations [31,36–38]. Here, we generalize from the typical smoothness constraints and consider *locally Lipschitz* maps. A function $\mathcal{F} : K \rightarrow L$ for metric spaces K and L is locally Lipschitz if

$$d_L(\mathcal{F}(\mathbf{x}), \mathcal{F}(\mathbf{x}')) \leq C_x d_K(\mathbf{x}, \mathbf{x}'), \quad (\text{A3})$$

where C_x is constant in some neighborhood of \mathbf{x} . Here, d_S is the distance function on the metric space S .

All practical neural sequence models are locally Lipschitz. Indeed, assuming that $L = \mathbb{R}^m$, all maps that are almost everywhere differentiable with locally bounded Jacobian norm are locally Lipschitz [50]. Realistically, then, locally Lipschitz models can be thought of as all models trainable using gradient-based methods; equivalently, they can be thought of as models trainable via methods not arbitrarily sensitive to local noise $\mathbf{x}_i \mapsto \mathbf{x}_i + \epsilon$.

APPENDIX B: PROOFS OF EXPRESSIVITY SEPARATIONS

Before giving proofs of expressivity separations between our quantum model and classical models, we first give a formal definition of the translation task we will prove a separation on, namely, (k, n) *stabilizer measurement translation*, parameterized by k and n . Note that the technical description of the task described here is a certain limit of the construction presented in the main text. First, we take $k \rightarrow k - n$, as our task is only explicitly defined when the sequence length is at least n . For instance, the $(n + 2, n)$ stabilizer measurement translation task as presented in the main text will, here, be referred to as the $(2, n)$ stabilizer measurement translation task. We make

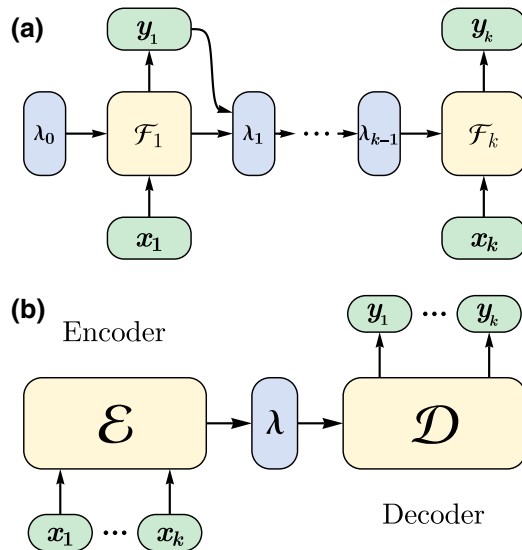


FIG. 4. (a) An online neural sequence model. The model autoregressively takes input tokens \mathbf{x}_i and outputs decoded tokens \mathbf{y}_i with map \mathcal{F}_i . The model also has an unobserved internal memory with state $\lambda_i \in L$ after decoding token i that \mathcal{F}_{i+1} can depend on. (b) A general encoder-decoder model. Here \mathcal{E} encodes input \mathbf{x} into some latent representation $\lambda \in L$. A decoder \mathcal{D} then outputs the decoded sequence \mathbf{y} .

this change as the formal definition of this task as presented here is undefined when $k + n < n$. Second, we now set the first n measurements to be infinite precision Gaussian measurements, i.e., measurements of linear combinations of position and momentum operators. These measurements are a limit of the periodic measurements we consider in the main text, with infinitely large periods.

To be more explicit, we consider an input language given by $(n + k)$ -long sequences of linear combinations of position and momentum operators on n modes. Specifically, input sentences are composed of words that are of the form of rows of

$$\mathbf{x} = \begin{pmatrix} s_{1,1}^q & \cdots & s_{1,n}^q & s_{1,1}^p & \cdots & s_{1,n}^p \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ s_{n+k,1}^q & \cdots & s_{n+k,n}^q & s_{n+k,1}^p & \cdots & s_{n+k,n}^p \end{pmatrix}. \quad (\text{B1})$$

The first n rows describe the sequential measurement of each operator

$$\hat{s}_i = \sum_{j=1}^n s_{ij}^q \hat{q}_j + \sum_{j=1}^n s_{ij}^p \hat{p}_j \quad (\text{B2})$$

when beginning in some given fixed state $|\psi_0\rangle$ on n modes that is either a GKP state [34] or an infinitely squeezed Gaussian state, which maintains the nonuniversality of the model [53]. The final k rows describe the sequential measurement of each operator

$$\hat{s}_i = \exp\left(i \sum_{j=1}^n s_{ij}^q \hat{q}_j + i \sum_{j=1}^n s_{ij}^p \hat{p}_j\right) \quad (\text{B3})$$

via, e.g., phase estimation, as shown in the main text. Note that the measurement of \hat{s}_i is *not* equivalent to the measurement of its generator. For instance, given states such that

$$\hat{q} |\psi_1\rangle = 0, \quad \hat{q} |\psi_2\rangle = 2\pi, \quad (\text{B4})$$

one has

$$(\exp(i\hat{q}) - 1) |\psi_1\rangle = (\exp(i\hat{q}) - 1) |\psi_2\rangle = 0. \quad (\text{B5})$$

A translation \mathbf{y} of \mathbf{x} is considered correct if it is of the form

$$\mathbf{b} = \begin{pmatrix} m_1 \\ \vdots \\ m_{n+k} \end{pmatrix}, \quad (\text{B6})$$

where the measurement outcomes m_i are consistent with those of quantum mechanics. To prove our separations, we consider input sentences that exhibit quantum contextuality.

1. Expressivity separation for online models

We now show that locally Lipschitz online with latent space dimension less than $n(n - 3)/2$ can stabilize measurement translate. Our general proof strategy is as follows.

1. We first define a potentially random online learning model with locally Lipschitz cell maps $\mathcal{F}_i^{\mathbf{r}}(\mathbf{s}_i, \lambda_{i-1}) = (m_i, \lambda_i)$, where we use m_i to indicate the measurement result when measuring the nullifier described by \mathbf{s}_i , and \mathbf{r} is a random vector shared between all \mathcal{F}_i . We assume for any \mathbf{r} that $\mathcal{F}_i^{\mathbf{r}}$ is deterministic. Let $\mathcal{F}^{\mathbf{r}}(\mathbf{s}_1, \dots, \mathbf{s}_n) = \lambda_n$ be the n -fold composition of $\mathcal{F}_i^{\mathbf{r}}$ on some fixed initial λ_0 (where any m_i is implicit, as each m_i is fully determined by \mathbf{r} and \mathbf{s}_i). Note that once \mathbf{r} is specified, $\mathcal{F}_i^{\mathbf{r}}$ is a deterministic function. Because of this, in the following, we take the \mathbf{r} dependence to be implicit.
2. We assume that the dimension of λ_i is less than $n(n - 3)/2$. We prove that then the described online model will give a wrong measurement outcome on the final two measurement results m_{n+1}, m_{n+2} . In the following, we refer to this as *the theorem statement*.
3. To prove that *the theorem statement* is true, we show that it is true for a subspace K of inputs that describe *CV graph states*, where the associated graphs have no self-loops. It is easy to see that the dimension of such a space is $n(n - 1)/2$, by studying the space of adjacency matrices. We let $\mathcal{F}|_K$ be \mathcal{F} restricted to this space K . Let \mathbf{B} be coordinates of K , as defined in Eq. (B23). We assume that the Jacobian of this map achieves its maximal rank at $\mathbf{B} = \mathbf{0}$, which describes a set of measurements yielding the position squeezed state $|\mathbf{0}\rangle_{\hat{q}}$. By doing so, we guarantee the robustness of the Jacobian rank in a neighborhood of $\mathbf{B} = \mathbf{0}$. The assumption that $\mathbf{B} = \mathbf{0}$ is a point of maximal rank is taken without loss of generality, as there exist Gaussian operations that transform the \mathbf{B} at which the Jacobian of $\mathcal{F}|_K$ attains its maximal rank to $\mathbf{B} = \mathbf{0}$.
4. As the rank is constant in the neighborhood of $\mathbf{B} = \mathbf{0}$, by the constant rank theorem [59], $\mathcal{F}_r|_K$ induces a fiber bundle structure in the neighborhood of $\mathbf{B} = \mathbf{0}$. That is, $\mathcal{F}_r|_K$ is a projection of fibers in a neighborhood of $\mathbf{B} = \mathbf{0}$ to their base points. This means that the model is unable to distinguish between points that share a fiber in this neighborhood.
5. We then show that, when $\dim(\lambda_n) < \dim(K) - n$, there exist $\mathbf{B}', \mathbf{B}''$ on the fiber with base point $\mathbf{B} = \mathbf{0}$ such that $\mathbf{B}, \mathbf{B}', \mathbf{B}''$ describe distinct states. We show in Lemma 1 below that these states have stabilizers that share quantum contextuality, yielding distinguishing one-shot measurement sequences. As the model is unable to distinguish between $\mathbf{B}, \mathbf{B}', \mathbf{B}''$, this implies that there exist $\mathbf{s}_{n+1}, \mathbf{s}_{n+2}$ describing this

distinguishing measurement sequence such that the model returns the wrong measurement result for at least one of $\mathbf{B}, \mathbf{B}', \mathbf{B}''$ with certainty. This proves *the theorem statement*. We also use this general proof strategy when considering encoder-decoder models in Theorem 3, up to some minor details.

With our proof strategy now clear, we now proceed to prove the details. First, we prove our lemma demonstrating that, indeed, the stabilizer operators we consider exhibit quantum contextuality. We also show that this contextuality induces a one-shot distinguishing measurement sequence on the states stabilized by the given operators.

Lemma 1 (CV graph state stabilizers exhibit quantum contextuality). *Consider $|\mathbf{0}\rangle_{\hat{\mathbf{q}}}$, the state nullified by all \hat{q}_i . Consider two states $|\psi_1\rangle$ and $|\psi_2\rangle$ that are CV graph states (up to arbitrary phases on their stabilizers) with no loops with distinct (modulo π) adjacency matrices. There exist operators that stabilize these three states that exhibit quantum contextuality. Furthermore, there exists a distinguishing measurement given by one of the stabilizers of $|\mathbf{0}\rangle_{\hat{\mathbf{q}}}$ that maps $|\psi_1\rangle$ and $|\psi_2\rangle$ to orthogonal postmeasurement states when the measurement result is 1; in other words, there exists a distinguishing measurement sequence of length two that distinguishes these three states.*

Proof. As $|\psi_1\rangle$ and $|\psi_2\rangle$ are CV graph states with distinct adjacency matrices (modulo π), they must differ (modulo π) in the edges \mathbf{e}_i touching some vertex i . That is, $|\psi_1\rangle$ is stabilized by some

$$s'_i = e^{2i\theta'} X_i(1)\mathbf{Z}(\mathbf{e}'_i) \quad (\text{B7})$$

and $|\psi_2\rangle$ by some

$$s''_i = e^{2i\theta''} X_i(1)\mathbf{Z}(\mathbf{e}''_i), \quad (\text{B8})$$

where \mathbf{e}'_i and \mathbf{e}''_i differ (modulo π) in some element indexed by $j \neq i$ (as there are no loops in either graph), and where $2\theta', 2\theta''$ are phases. Here, $\mathbf{Z}(\cdot)$ is defined as the tensor product:

$$\mathbf{Z}(\mathbf{v}) = \bigotimes_i Z_i(v_i). \quad (\text{B9})$$

An example diagram of the entries of $\mathbf{e}'_i, \mathbf{e}''_i$ is given in Fig. 5. By the symmetry of CV graph state adjacency matrices, we thus find that the former is also stabilized by

$$s'_j = e^{2i\phi'} X_j(1)\mathbf{Z}(\mathbf{e}'_j) \quad (\text{B10})$$

and the latter by

$$s''_j = e^{2i\phi''} X_j(1)\mathbf{Z}(\mathbf{e}''_j), \quad (\text{B11})$$

where \mathbf{e}'_j and \mathbf{e}''_j differ (modulo π) in some element indexed by i , and where $2\phi', 2\phi''$ are phases. Note in particular that we have the commutation relations

$$[s'_i, s''_i] = 0, \quad [s'_j, s''_j] = 0, \quad [s'_i, s'_j] = 0, \quad [s''_i, s''_j] = 0, \quad (\text{B12})$$

$$\begin{aligned} s'_i s''_j &= e^{2i\zeta} s''_j s'_i, & s'_j s''_i &= e^{2i\zeta} s''_i s'_j, & s'_i s''_j{}^\dagger &= e^{-2i\zeta} s''_j{}^\dagger s'_i, \\ s'_j s''_i{}^\dagger &= e^{-2i\zeta} s''_i{}^\dagger s'_j, \end{aligned} \quad (\text{B13})$$

where

$$\zeta \equiv e'_{ij} - e''_{ij} = e'_{ji} - e''_{ji} \neq 0 \pmod{\pi}. \quad (\text{B14})$$

As $\zeta \neq 0 \pmod{\pi}$, there exists some $\alpha \in \mathbb{R}_+^*$ such that

$$\{s'_i{}^\alpha, s''_j{}^\alpha\} = 0, \quad \{s'_j{}^\alpha, s''_i{}^\alpha\} = 0, \quad (\text{B15})$$

$$\{s'_i{}^\alpha, s''_j{}^{\alpha\dagger}\} = 0, \quad \{s'_j{}^\alpha, s''_i{}^{\alpha\dagger}\} = 0. \quad (\text{B16})$$

To save on notation, we redefine all stabilizers to be given by their α power, i.e., $s^\alpha \rightarrow s$ (and similarly redefine the phases $\theta', \theta'', \phi', \phi''$ by their scaling by α). As $|\psi_1\rangle, |\psi_2\rangle$ are CV graph states, these rescaled operators are still stabilizers. We also define

$$s_i \equiv e^{-2i(\theta' - \theta'')} s'_i s''_i{}^\dagger, \quad (\text{B17})$$

$$s_j \equiv e^{-2i(\phi' - \phi'')} s'_j s''_j{}^\dagger, \quad (\text{B18})$$

which are stabilizers of $|\mathbf{0}\rangle_{\hat{\mathbf{q}}}$. Thus, we have constructed nine observables with constraints satisfying those of a Mermin-Peres magic square (see Table III for an example), a well-known proof of quantum contextuality [43].

Consider now the postmeasurement states $|\psi'_1\rangle, |\psi'_2\rangle$ of $|\psi_1\rangle, |\psi_2\rangle$, respectively, when $s_i s_j$ is measured to be 1. By Table III, $|\psi'_1\rangle$ is stabilized by $s'_i s'_j$ and $s_i s_j$; furthermore, $|\psi'_2\rangle$ is stabilized by $s''_i s''_j$ and $s_i s_j$, and therefore is also stabilized by

$$s_i s_j s''_i s''_j = -e^{-2i(\theta' - \theta'' + \phi' - \phi'')} s'_i s'_j. \quad (\text{B19})$$

If

$$\theta' - \theta'' + \phi' - \phi'' \neq \frac{\pi}{2} \pmod{\pi}, \quad (\text{B20})$$

$|\psi'_1\rangle$ and $|\psi'_2\rangle$ are orthogonal. If it is congruent to $\pi/2 \pmod{\pi}$ then either $\theta' - \theta'' \neq 0 \pmod{\pi}$ or $\phi' - \phi'' \neq 0 \pmod{\pi}$ (or both). Assume the latter without loss of generality (the former case is the same with $\phi \rightarrow \theta$ and $j \rightarrow i$),

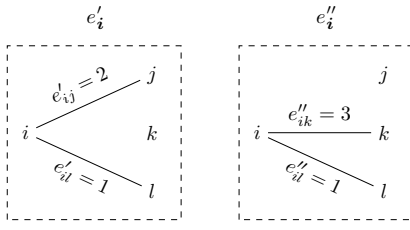


FIG. 5. An example of two graphs, with edges leaving vertex i given by \mathbf{e}'_i and \mathbf{e}''_i , respectively. As the two graphs differ, they must differ by an edge; indeed, they differ in the weights of both $\langle i, j \rangle$ and $\langle i, k \rangle$.

and instead consider the postmeasurement states $|\psi'_1\rangle, |\psi'_2\rangle$ of $|\psi_1\rangle, |\psi_2\rangle$, respectively, when s_j is measured to be 1. By Table III, $|\psi'_1\rangle$ is stabilized by s'_j and s_j ; furthermore, $|\psi'_2\rangle$ is stabilized by s''_j and s_j , and therefore is also stabilized by

$$s_j s'_j = e^{-2i(\phi' - \phi'')} s'_j. \quad (\text{B21})$$

Thus, again, $|\psi'_1\rangle$ and $|\psi'_2\rangle$ are orthogonal. ■

We now consider the locally Lipschitz online learner, with structure given by Fig. 4(a). We assume that the learner is deterministic, and discuss the extension to randomized models at the end of this section.

Online neural sequence models at time step i map an input token \mathbf{x}_i and a latent vector λ_{i-1} to an output token m_i and a new latent vector λ_i . After n steps, then, we can consider the online model as a locally Lipschitz map

$$\mathcal{F}^{\mathbf{r}} : (\mathbb{R}^{2n})^n \rightarrow L \times \mathbb{R}^n, \quad (\text{B22})$$

where L is a locally Lipschitz latent manifold and \mathbf{r} is a random vector such that, for any \mathbf{r} , $\mathcal{F}^{\mathbf{r}}$ is deterministic. This consideration of $\mathcal{F}^{\mathbf{r}}$ as a deterministic function of a random \mathbf{r} is typically the implementation of stochastic learners, such as generative adversarial networks [64] and flow-based models [65]. This also includes implementations of stochastic simulation algorithms such as Wigner function simulation [32]. As for each \mathbf{r} , there

TABLE III. The Mermin-Peres magic square of stabilizers of states mapping to the same latent space under a locally Lipschitz map (with $\theta' = \theta'' = \phi' = \phi'' = 0$ for simplicity). Stabilizers of $|\psi_1\rangle, |\psi_2\rangle$, and $|0\rangle_{\hat{q}}$ make up the three rows. All observables in each row and column commute. Furthermore, the product of observables in each row and column is the identity, except for the third column, which gives minus the identity. See the main text for a special case of this magic square.

s'_i	s'_j	$s_i s'_j$
s''_i	s''_j	$s_i s''_j$
s_i	s_j	$s_i s_j$

exists an input sequence such that any classical model with $\dim(L) < n(n-3)/2$ deterministically outputs a measurement sequence inconsistent with quantum mechanics; our results still hold for these classes of random models. Because of this, in the following, we take the \mathbf{r} dependence to be implicit.

With the preliminaries in place, we now prove our expressivity separation.

Theorem 2 (Online stabilizer measurement translation lower bound). *Consider an online model with locally Lipschitz latent manifold L and locally Lipschitz map \mathcal{F} as described in Eq. (B22). If $\dim(L) < n(n-3)/2$, this model cannot achieve a finite backward empirical cross entropy on the $(2, n)$ stabilizer measurement translation task.*

Proof. Consider $K \subset (\mathbb{R}^{2n})^n$, with elements of the form

$$\mathbf{Q} = (\mathbf{B} + \mathbf{H}/2 \mid \|\mathbf{B}\|_F \mathbf{I}_n); \quad (\text{B23})$$

here $\|\mathbf{B}\|_F$ is the Frobenius norm of \mathbf{B} ; \mathbf{B} is an $n \times n$ hollow (zero diagonal elements) symmetric matrix with entries bounded to be $[-1/4, 1/4]$; and \mathbf{H} is the fixed $n \times n$ symmetric hollow matrix of ones, i.e.,

$$\mathbf{H} = \begin{pmatrix} 0 & 1 & \cdots & 1 \\ 1 & 0 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 0 \end{pmatrix}. \quad (\text{B24})$$

It is obvious from this construction that K is an $[n(n-1)/2]$ -dimensional embedding of a compact subspace of hollow symmetric matrices (with bounded entries and norm) \mathbf{B} . Note that the states described by the measurement scenarios of points in K are exactly CV graph states without loops (with bounded weight edges, as K is compact) and, depending on the measurement results, perhaps overall phases on the stabilizers. To see this, note that the symmetric constraint on \mathbf{B} ensures that the symplectic product of any two rows of \mathbf{Q} is zero; furthermore, the final n columns of \mathbf{Q} are linearly independent for all $\mathbf{B} \neq \mathbf{0}$, and the first n columns for $\mathbf{B} = \mathbf{0}$ due to the shift by \mathbf{H} . Thus, all points in K are full row rank, and the rows of \mathbf{Q} completely determine the CV stabilizer state, up to phases given by the measurement results of these operators. Furthermore, as \mathbf{B} and \mathbf{H} are hollow, the CV graph state \mathbf{Q} describes has no loops. Also, note that, up to independent rescalings of the rows of \mathbf{Q} , different \mathbf{Q} correspond to different graph states. We assume without loss of generality that the Jacobian of \mathcal{F} attains its maximum rank at $\mathbf{B} = \mathbf{0}$ (that is, the squeezed state $|0\rangle_{\hat{q}}$); this can always be done by implicitly transforming the basis of inputs to the model [i.e., by appropriately relabeling points in $(\mathbb{R}^{2n})^{n+2}$], and then considering K as previously defined in this new basis.

We proceed as follows. First, we show that, when $\dim(L)$ is sufficiently small, \mathcal{F} must map three distinct CV graph states described by different \mathbf{Q} to the same point in latent space. Then, we use Lemma 1 to show that the stabilizers of these states exhibit quantum contextuality (independent of the associated n measurement results), and give rise to a distinguishing measurement sequence. Thus, by considering measurement sequences of length $n + 2$ that include these three \mathbf{Q} and the length two distinguishing measurement sequence, one of the final two measurement outcomes must be incorrect. This implies that there is an infinite backward empirical cross entropy on any finite set containing these three measurement sequences.

Let us begin by showing that $\mathcal{F}|_K$ (i.e., the locally Lipschitz map that is \mathcal{F} restricted to K) must map three nontrivially distinct \mathbf{Q} (i.e., three distinct CV graph states) to the same point in latent space when $\dim(L)$ is sufficiently small. By the constant rank theorem and the local Lipschitzness of $\mathcal{F}|_K$, $\mathcal{F}|_K$ is not injective for

$$\dim(L) < \dim(K) = \frac{n(n-1)}{2}. \quad (\text{B25})$$

In particular, in a sufficiently small neighborhood of $\mathbf{B} = \mathbf{0}$ (where the Jacobian of $\mathcal{F}|_K$ attains its maximal rank), there exist local coordinates $\tilde{\mathbf{x}}$ of K and L such that

$$\mathcal{F}|_K(\tilde{x}_1, \dots, \tilde{x}_{n(n-1)/2}) = (\tilde{x}_1, \dots, \tilde{x}_l, 0, \dots, 0) \quad (\text{B26})$$

for some $l \leq \dim(L) < n(n-1)/2$ [59]. Without loss of generality, we identify $\tilde{\mathbf{x}} = \mathbf{0}$ with $\mathbf{B} = \mathbf{0}$, which is the state infinitely squeezed in all \hat{q}_i . We call C the fiber with local coordinates

$$\tilde{\mathbf{x}} = (0, \dots, 0, \tilde{x}_{l+1}, \dots, \tilde{x}_{n(n-1)/2}), \quad (\text{B27})$$

which is of dimension at least

$$\Delta \equiv \frac{n(n-1)}{2} - \dim(L) \geq 1. \quad (\text{B28})$$

By construction, all points in C —including $\mathbf{B} = \mathbf{0}$ —map to the same point $l \in L$ under $\mathcal{F}|_K$. We now assume that $\dim(L) < n(n-3)/2$ such that $\Delta \geq n + 1$.

Now fix $\mathbf{B} = \mathbf{0}$ and $\mathbf{B}' \neq \mathbf{B}$ in C . As described previously, \mathbf{Q} (and thus \mathbf{B}) completely determines a CV graph state after n measurements, up to independent rescalings of the rows of \mathbf{Q} (and the measurement results). As the dimension of the space of points that differ (modulo π) from $\mathbf{B}' + \mathbf{H}/2$ by just a scaling factor in each row is at most n , because $\Delta \geq n + 1$, it must follow that there exists another $\mathbf{B}' \neq \mathbf{B}$, \mathbf{B}' describing a distinct CV graph state. Therefore, by Lemma 1, there exists a distinguishing measurement sequence for these three states. Note that, as Lemma 1 does not depend on the phases of the CV stabilizers, the existence of this distinguishing

measurement holds regardless of what the measurement results are (i.e., independently from what the model outputs for the first n tokens in the decoded sequence). As after n tokens all three sequences map to the same point in latent space in the model, and as they share a distinguishing measurement sequence, the model must obtain an infinite backward empirical cross entropy on these three input sequences when followed by the distinguishing measurement sequence. ■

2. A CV Gottesman–Knill lower bound

We now show that our results can be reformulated as a memory lower bound on the classical simulation of stabilizer measurement scenarios. In practice, using finite resources (i.e., at finite precision), any classical ontological model simulating $p(\mathbf{y} | \mathbf{x})$ can only be evaluated at a finite number of \mathbf{x} . We now show that *any* locally Lipschitz interpolation of such a model to real \mathbf{x} cannot accurately simulate Gaussian operations on an initial GKP state. This includes, for instance, any polynomial interpolation (which always exists).

Corollary 1 (CV Gottesman–Knill lower bound). *Consider a classical ontological model $p(\mathbf{y} | \mathbf{x})$ with a latent space of dimension less than $n(n-3)/2$, simulating the (k, n) stabilizer measurement translation with $k \geq 2$. Assume that this ontological model is defined at a finite number of \mathbf{x} . There exists a locally Lipschitz interpolation of this model to all \mathbf{x} . Furthermore, no locally Lipschitz interpolation of this model can faithfully perform stabilizer measurement translation at all \mathbf{x} .*

Proof. As there exists a polynomial interpolation of p , and as all polynomials of finite degree are locally Lipschitz, there exists a locally Lipschitz interpolation of this model to all \mathbf{x} . Furthermore, no locally Lipschitz interpolation of this model can faithfully perform stabilizer measurement translation by Theorem 2, as the composition of locally Lipschitz functions is locally Lipschitz. ■

3. Expressivity separation for encoder-decoder models

Though online sequence models are perhaps conceptually the simplest as they directly map input tokens to output tokens, in practice, encoder-decoder models outperform them [30,38]. We now show that no encoder-decoder model with a locally Lipschitz encoder (and an additional technical assumption) can perform stabilizer measurement translation to finite backward empirical cross entropy. The proof will be similar to that of Theorem 2; however, as the model can see the entire input sequence at once, we do not directly have the freedom to choose the distinguishing measurement sequence as in Theorem 2. Instead, we require an input sequence of length quadratic in n (and

our additional technical assumption) to force the distinguishing measurement sequence. Note that, as the memory of the contextual learner is independent of the sequence length, this new sequence length has no impact on the memory separation.

We consider an encoder-decoder model with structure given by Fig. 4(b). The encoder of such a model can be considered a locally Lipschitz map

$$\mathcal{E}^{\mathbf{r}} : (\mathbb{R}^{2n})^{n^2} \rightarrow L \quad (\text{B29})$$

to some locally Lipschitz latent manifold L . As in Appendix B 1, \mathbf{r} is a random vector such that, for any \mathbf{r} , $\mathcal{E}^{\mathbf{r}}$ is deterministic. We once again make the \mathbf{r} dependence implicit in the following.

For technical reasons, we slightly change the definition of the (k, n) stabilizer measurement translation task, where now the final k measurement descriptions instead describe the measurements of the operators, i.e.,

$$\hat{s}_i = \exp \left(i \sum_{j=1}^n \frac{\mathbf{1}[s_{ij}^q \neq 0]}{s_{ij}^q} \hat{q}_j + i \sum_{j=1}^n \frac{\mathbf{1}[s_{ij}^p \neq 0]}{s_{ij}^p} \hat{p}_j \right), \quad (\text{B30})$$

where we define $\mathbf{1}[x \neq 0]/x$ to be zero when $x = 0$. We call this the *modified (k, n) stabilizer measurement translation task*. This can obviously still be performed perfectly with a CRNN of model size n , by either changing the parameters of the phase estimation circuit used in the CRNN to be given by $\mathbf{1}[x_{ij} \neq 0]/x_{ij}$, or, more formally, by introducing a quantum circuit computing $\mathbf{1}[x_{ij} \neq 0]/x_{ij}$ on which these gates control.

We now discuss our additional technical assumption. Defining the subspace R of inputs as in the proof of Theorem 3, we assume that the Jacobian of the encoder restricted to R attains its maximal rank at some point of the form $(\mathbf{Q}, \mathbf{0}) \in R$. A sufficient condition for this is requiring that some point of the form $(\mathbf{Q}, \mathbf{0})$ is not a critical point of $\mathcal{E}|_R$; this condition is satisfied by generic \mathcal{E} when $\dim(L) < n(n-3)/2$, and also by models with encoders constrained to be submersions (such as encoders composed of linear transformations and tanh or sigmoid nonlinearities). In fact, when the encoder is constrained to be a submersion, it is easy to see from the proof of Theorem 3 that the separation still holds on the *unmodified (k, n) stabilizer measurement translation task*, as all properties we use that hold locally then hold globally. Any one of these conditions is sufficient, and needed for our proof technique to be able to analyze any neighborhood of the non-Gaussian measurements we consider.

Theorem 3 (Encoder-decoder stabilizer measurement translation lower bound). *Consider an encoder-decoder model with locally Lipschitz latent manifold L . Let \mathcal{E} be the*

associated locally Lipschitz encoder function, as defined in Eq. (B29), and assume that the Jacobian of the map $\mathcal{E}|_R$ (where the subspace R is defined below) attains its maximal rank at some point of the form $(\mathbf{Q}, \mathbf{0}) \in R$. If $\dim(L) < n(n-3)/2$, this model cannot achieve a finite backward empirical cross entropy on the modified $(n^2 - n, n)$ stabilizer measurement translation task.

Proof. Consider $R \equiv K \times (\mathbb{R}^{2n})^{n^2-n} \subset (\mathbb{R}^{2n})^n \times (\mathbb{R}^{2n})^{n^2-n} \cong (\mathbb{R}^{2n})^{n^2}$ with elements (\mathbf{Q}, \mathbf{P}) of the following form:

1. $\mathbf{Q} \in K$ is given by rows of matrices of the form

$$\mathbf{Q} = (\mathbf{B} + \mathbf{H}/2 \quad | \quad \|\mathbf{B}\|_F \mathbf{I}_n), \quad (\text{B31})$$

where $\|\mathbf{B}\|_F$ is the Frobenius norm of \mathbf{B} ; \mathbf{B} is a hollow symmetric matrix with entries bounded to be $[-1/4, 1/4]$; and \mathbf{H} is the fixed symmetric hollow matrix of ones, i.e.,

$$\mathbf{H} = \begin{pmatrix} 0 & 1 & \cdots & 1 \\ 1 & 0 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 0 \end{pmatrix}; \quad (\text{B32})$$

2. $\mathbf{P} \in (\mathbb{R}^{2n})^{n^2-n}$ is an $n^2 - n \times 2n$ matrix that is arbitrary.

It is obvious from this construction that K is an $[n(n-1)/2]$ -dimensional embedding of a compact subspace of hollow symmetric matrices (with bounded entries and norm) \mathbf{B} . We assume without loss of generality that the Jacobian of $\mathcal{E}|_K$ [that is, the locally Lipschitz restriction of \mathcal{E} to points of the form $(\mathbf{Q}, \mathbf{0}) \in R$] attains its maximum rank at $\mathbf{B} = \mathbf{0}$ (that is, the squeezed state $|\mathbf{0}\rangle_{\hat{q}}$); this can always be done by implicitly transforming the basis of the inputs to the model [i.e., by appropriately relabeling points in $(\mathbb{R}^{2n})^{n^2}$], and then considering K as previously defined in this new basis.

We now give some intuition behind points in the smooth manifold (with boundary) R . At fixed $\mathbf{P} = \mathbf{0}$, the states described by the measurement scenarios of points in R are exactly CV graph states without loops (with bounded weight edges, as K is compact) and, depending on the measurement results, perhaps overall phases on the stabilizers. To see this, note that the symmetric constraint on \mathbf{B} ensures that the symplectic product of any two rows of \mathbf{Q} is zero; furthermore, the final n columns of \mathbf{Q} are linearly independent for all $\mathbf{B} \neq \mathbf{0}$, and the first n columns for $\mathbf{B} = \mathbf{0}$ due to the shift by \mathbf{H} . Thus, all points in K are full row rank, and the rows of \mathbf{Q} completely determine the CV stabilizer state, up to phases given by the measurement results of these operators (which are not yet determined at the time of encoding). Furthermore, as \mathbf{B} and \mathbf{H} are hollow,

the CV graph state \mathbf{Q} describes has no loops. Also, note that, up to trivial rescalings of the rows of \mathbf{Q} , different \mathbf{Q} correspond to different graph states. At general \mathbf{P} , the state after the first n measurements is still a CV graph state completely determined by \mathbf{Q} (up to phases from the first n measurement results); different \mathbf{P} correspond to different (non-Gaussian) measurement scenarios given an initial CV graph state determined by \mathbf{Q} (and the first n measurement results).

We proceed as follows. First, we show that, as $\dim(L) < n(n-3)/2$, \mathcal{E} must map three distinct CV graph states described by different \mathbf{Q} to the same point in latent space when $\mathbf{P} = \mathbf{0}$. Then, we use Lemma 1 to show that the stabilizers of these states exhibit quantum contextuality (independent of the associated n measurement results), and give rise to a distinguishing measurement sequence. Finally, we show that one can locally find $\mathbf{P} \neq \mathbf{0}$ mapping to the same point in latent space that contains this distinguishing measurement sequence, forcing an incorrect measurement outcome on one of these states. This gives rise to an infinite backward empirical cross entropy on this task.

Let us begin by showing that $\mathcal{E}|_R$ (i.e., the locally Lipschitz map that is \mathcal{E} restricted to R) must map three nontrivially distinct \mathbf{Q} (i.e., three distinct CV graph states) to the same point in latent space. We consider the restriction $\mathcal{E}|_K$, which is the (locally Lipschitz) restriction of \mathcal{E} to points of the form $(\mathbf{Q}, \mathbf{0}) \in R$. We show that this map is not injective for small enough $\dim(L)$. As $\mathcal{E}|_R$ lifts to $\mathcal{E}|_K$, this will show that three distinct \mathbf{Q} map to the same point in latent space under $\mathcal{E}|_R$. This is similar to the construction for $\mathcal{F}|_K$ in the proof Theorem 2; we repeat it here for completeness.

By the constant rank theorem and the local Lipschitzness of $\mathcal{E}|_K$, $\mathcal{E}|_K$ is not injective for

$$\dim(L) < \dim(K) = \frac{n(n-1)}{2}. \quad (\text{B33})$$

In particular, in a sufficiently small neighborhood of $\mathbf{B} = \mathbf{0}$ (where the Jacobian of $\mathcal{E}|_K$ attains its maximal rank), there exist local coordinates $\tilde{\mathbf{x}}$ of K and L such that

$$\mathcal{E}|_K(\tilde{x}_1, \dots, \tilde{x}_{n(n-1)/2}) = (\tilde{x}_1, \dots, \tilde{x}_l, 0, \dots, 0) \quad (\text{B34})$$

for some $l \leq \dim(L) < n(n-1)/2$ [59]. Without loss of generality, we identify $\tilde{\mathbf{x}} = \mathbf{0}$ with $\mathbf{B} = \mathbf{0}$, which is the state infinitely squeezed in all \hat{q}_i . We call C the fiber with local coordinates

$$\tilde{\mathbf{x}} = (0, \dots, 0, \tilde{x}_{l+1}, \dots, \tilde{x}_{n(n-1)/2}), \quad (\text{B35})$$

which is of dimension at least

$$\Delta \equiv \frac{n(n-1)}{2} - \dim(L) \geq 1. \quad (\text{B36})$$

By construction, all points in C —including $\mathbf{B} = \mathbf{0}$ —map to the same point $l \in L$ under $\mathcal{E}|_K$. We now assume that $\dim(L) < n(n-3)/2$ such that $\Delta \geq n+1$.

Now fix $\mathbf{B} = \mathbf{0}$ and $\mathbf{B}' \neq \mathbf{B}$ in C . As described previously, \mathbf{Q} (and thus \mathbf{B}) completely determines a CV graph state after n measurements, up to trivial rescalings of the rows of \mathbf{Q} and the measurement results. As the dimension of the space of points that differ (modulo π) from $\mathbf{B}' + \mathbf{H}/2$ by just a scaling factor in each row is at most n , because $\Delta \geq n+1$, it must follow that there exists another $\mathbf{B}'' \neq \mathbf{B}, \mathbf{B}'$ describing a distinct CV graph state. Therefore, by Lemma 1, there exists a distinguishing measurement s that is a stabilizer of the state corresponding to $\mathbf{B} = \mathbf{0}$. Note that, as Lemma 1 does not depend on the phases of the CV stabilizers, the existence of this distinguishing measurement holds regardless of what the measurement results are (i.e., independently from what the model outputs for the first n tokens in the decoded sequence). Depending on these measurement results, however, the distinguishing measurement could be one of three different measurement sequences, depending on the validity of Eq. (B20).

We have now shown that there exists $(\mathbf{Q}_i, \mathbf{0})$ for $1 \leq i \leq 3$ such that all \mathbf{Q}_i are distinct, and that there exists a measurement of a stabilizer of \mathbf{Q}_1 such that the post-measurement states of $\mathbf{Q}_2, \mathbf{Q}_3$ are orthogonal. We now show that there exist \mathbf{P}_i such that $(\mathbf{Q}_i, \mathbf{P}_i)$ also maps to the same point in latent space, and \mathbf{P}_i is a distinguishing measurement sequence. This will give the final separation.

First, note that one can find a \mathbf{P}_i given an arbitrarily small bound on its norm that encodes a distinguishing measurement sequence; this is because, in the proof of Lemma 1, one can repeatedly take $\alpha \rightarrow 3\alpha$ to yield a distinguishing measurement sequence using arbitrarily large stabilizer powers, which corresponds to arbitrarily small \mathbf{P}_i in the modified stabilizer measurement translation task. Now, consider $\mathcal{E}|_S$, defined as the restriction of \mathcal{E} to points of the form $(\mathbf{Q}_i, \mathbf{P}) \in R$ for any of the fixed $\mathbf{Q}_i \in K$, where \mathbf{P} is zero in all rows except for the last S rows. Initially, let $S = S_1 = 2$; that is, \mathbf{P} is restricted to be all zero except for its final two rows, which are allowed to vary. Let d_{S_1} be the dimension of the image of $\mathcal{E}|_S$. If $d_{S_1} = 0$ then \mathcal{E} is locally independent from the final two rows of \mathbf{P} at $\mathbf{Q}_i, \mathbf{m}_i$ when all other rows are fixed to be zero, and we are done as we can set these two rows to be the distinguishing measurement sequence. If $d_{S_1} > 0$ then consider $S = S_2 = 4$. If $d_{S_2} = d_{S_1}$ then, for all local choices of the third and fourth final rows, there exists a choice of the final two rows such that \mathcal{E} is constant. Then, all of the possible distinguishing measurement scenarios can be encoded into the third and fourth final rows of \mathbf{P} (with the appropriate choice of final two rows) and map to the same point in latent space as when $\mathbf{P} = \mathbf{0}$, yielding the appropriate distinguishing measurement sequence.

If $d_{S_2} > d_{S_1}$ instead, we are able to iterate this procedure once more. As $\dim(L) < n(n-3)/2$, eventually this

iteration will stop with $d_{s_i} = d_{s_{i-1}}$, and we have the freedom to set two rows of \mathbf{P} to be the distinguishing measurement sequence and map to the same point in latent space as $\mathbf{P} = \mathbf{0}$.

We have thus shown that the encoder-decoder model must map three points in R that give rise to a distinguishing measurement scenario to the same point in latent space. That is, when $\dim(L) < n(n-3)/2$, there exists input sequences \mathbf{s}_i that map to the same point in the latent space of the model that must give rise to orthogonal measurement results. As they are mapped to the same point in the latent space of the model, the model must output the same translation for all of them, giving at least one incorrect result. Thus, the backward empirical cross entropy when translating one of these \mathbf{s}_i must be infinite. ■

APPENDIX C: CONSIDERATIONS FOR EXPERIMENTAL IMPLEMENTATIONS

We have shown in Appendix B that CRNNs are more expressive than both online and encoder-decoder sequence models, assuming a locally Lipschitz condition on the models (and an additional technical condition on the latter class of models). Though CRNNs only utilize Gaussian operations—which are believed to be much simpler to implement than universal CV quantum computing [51]—our model also utilizes non-Gaussian ancilla states to perform non-Gaussian measurements. As we formally compare our quantum model with infinite precision classical models, we have a formal requirement for GKP ancilla states with infinite homodyne precision measurements to show a separation.

Of course, in practice, classical neural sequence models are finite precision. In particular, tensor processing unit implementations of classical neural networks often use as imprecise as 8-bit arithmetic. We therefore expect that one can circumvent the formal need for GKP states to use *qubit* ancilla states to perform phase estimation of the CV stabilizer operators to a precision matching that of classical neural networks. There are proposals for engineering the required longitudinal photon-qubit interactions using circuit quantum electrodynamics [66,67]. As similar couplings are already used in proposals for the generation of approximate GKP states [68], this direct approach is likely more experimentally feasible. Furthermore, such a finite precision implementation may actually *gain* expressive power compared to infinite precision CRNNs. Assuming that the back action of the finite precision measurement yields a finitely squeezed state in the CRNN, with this, one can construct a model capable of universal CV quantum computation [53,54]. This also holds when the initial state of the model is taken to be the vacuum state (or any other finitely squeezed Gaussian state). This suggests a potential superpolynomial advantage in the expressive power

and the time complexity of inference when this model is implemented at finite precision.

If one wishes to avoid coupling to qubits, we make the bolder conjecture that any non-Gaussian ancilla state is enough to yield a separation. Our intuition for this comes from recent work [60,61] demonstrating that non-Gaussian operations are equivalent to the presence of quantum contextuality. As the presence of quantum contextuality is the source of the separations in our proofs, this gives evidence that one could use a more experimentally feasible non-Gaussian ancilla state than a GKP state—such as a photon subtracted state [69]—and achieve similar results.

APPENDIX D: CLASSICAL SIMULATION OF GAUSSIAN OPERATIONS AND GKP STATES

We now describe the high level strategy of classically simulating both Gaussian operations applied to Gaussian states, and (restricted) Gaussian operations applied to GKP states. The former strategy will roughly follow that given in Ref. [70], and the latter that given in Ref. [71]. These will be the building blocks of the Gaussian RNN and contextual RNN cells we numerically test in the main text; we give the details of the full architectures, including the classical architectures, in Appendix E.

1. Gaussian states

First, we describe the simulation of Gaussian operations performed on Gaussian states. It is well known that any N mode Gaussian pure state can be created from the ground state of N harmonic oscillators with unitary operations $e^{if(\hat{\mathbf{q}}, \hat{\mathbf{p}})}$, where $f(\hat{\mathbf{q}}, \hat{\mathbf{p}})$ are at most quadratic in terms of the quadrature operators $\hat{\mathbf{q}}$ and $\hat{\mathbf{p}}$. By stacking $\hat{\mathbf{q}}$ and $\hat{\mathbf{p}}$ together, we call $\hat{\mathbf{x}} = (\hat{\mathbf{q}}, \hat{\mathbf{p}})^\top$ the quadrature operator. The Heisenberg evolution of the quadrature operator under a Gaussian unitary operator \mathbf{R} is in general

$$\hat{\mathbf{x}}' = \mathbf{R}^\dagger \hat{\mathbf{x}} \mathbf{R} = \mathbf{S} \hat{\mathbf{x}} + \mathbf{c}, \tag{D1}$$

where \mathbf{S} is a symplectic matrix of c numbers, and $\mathbf{c} = (\mathbf{c}_q, \mathbf{c}_p)^\top$ is a vector of c numbers denoting the mode center shift. There are various ways to decompose a symplectic matrix \mathbf{S} ; without loss of generality (as discussed in Ref. [70]), we consider symplectic matrices of the form

$$\mathbf{S} = \begin{pmatrix} \mathbf{U}^{-1/2} & \mathbf{0} \\ \mathbf{V} \mathbf{U}^{-1/2} & \mathbf{U}^{1/2} \end{pmatrix}, \tag{D2}$$

where \mathbf{U} and \mathbf{V} are both real symmetric matrices. In addition, \mathbf{U} is positive definite, i.e., $\mathbf{U} = \mathbf{U}^\top > 0$. If the covariance matrix of the N mode ground state is

$$\text{cov}(\hat{\mathbf{x}}_0) = \frac{1}{2} \mathbf{I}, \tag{D3}$$

then the covariance matrix of the transformed Gaussian state is

$$\begin{aligned}\boldsymbol{\Sigma} &= \text{cov}(\hat{\mathbf{x}}) \\ &= \text{cov}(\mathbf{S}\hat{\mathbf{x}}_0) \\ &= \frac{1}{2}\langle\{(\mathbf{S}\hat{\mathbf{x}}_0)^\dagger, (\mathbf{S}\hat{\mathbf{x}}_0)^\top\}\rangle \\ &= \frac{1}{2}\mathbf{S}\mathbf{S}^\top \\ &= \frac{1}{2}\begin{pmatrix} \mathbf{U}^{-1} & \mathbf{U}^{-1}\mathbf{V} \\ \mathbf{V}\mathbf{U}^{-1} & \mathbf{U} + \mathbf{V}\mathbf{U}^{-1}\mathbf{V} \end{pmatrix}.\end{aligned}\quad (\text{D4})$$

We can also write down its wave function in position space as

$$\begin{aligned}\psi_{\mathbf{Z},\mathbf{c}}(\mathbf{q}) &= \pi^{-N/4}(\det(\mathbf{U}))^{1/4} \\ &\quad \times \exp\left(-\frac{1}{2}(\mathbf{q} - \mathbf{c}_q)^\top(\mathbf{U} - i\mathbf{V})(\mathbf{q} - \mathbf{c}_q)\right) \\ &= \pi^{-N/4}(\det(\mathbf{U}))^{1/4} \\ &\quad \times \exp\left(\frac{i}{2}(\mathbf{q} - \mathbf{c}_q)^\top\mathbf{Z}(\mathbf{q} - \mathbf{c}_q)\right),\end{aligned}\quad (\text{D5})$$

where \mathbf{q} and \mathbf{c}_q are c number column vectors, and $\mathbf{Z} = \mathbf{V} + i\mathbf{U}$ is a complex symmetric matrix. We can interpret \mathbf{Z} as the adjacency matrix for an undirected graph with complex-valued edge weights. Therefore, any Gaussian pure states can be interpreted as Gaussian graph states with complex-valued weights on the graph edge. As Gaussian states are uniquely identified by the linear combinations of position and momentum operators that nullify them [72], we can consider what the nullifiers are for the CV graph state defined by \mathbf{Z} and $\mathbf{c} = (\mathbf{c}_q, \mathbf{c}_p)^\top$:

$$\begin{aligned}(\hat{\mathbf{p}} - \mathbf{Z}\hat{\mathbf{q}} + \mathbf{Z}\mathbf{c}_q - \mathbf{c}_p) |\psi_{\mathbf{Z},\mathbf{c}}\rangle & \\ &= (\hat{\mathbf{p}} - \mathbf{Z}\hat{\mathbf{q}} + \mathbf{Z}\mathbf{c}_q - \mathbf{c}_p)\mathbf{R}_{\mathbf{Z},\mathbf{c}} |0\rangle \\ &= \mathbf{R}_{\mathbf{Z},\mathbf{c}}\mathbf{R}_{\mathbf{Z},\mathbf{c}}^\dagger(\hat{\mathbf{p}} - \mathbf{Z}\hat{\mathbf{q}} + \mathbf{Z}\mathbf{c}_q - \mathbf{c}_p)\mathbf{R}_{\mathbf{Z},\mathbf{c}} |0\rangle \\ &= \mathbf{R}_{\mathbf{Z},\mathbf{c}}(-\mathbf{Z}, \mathbf{I})(\mathbf{S}_Z\hat{\mathbf{x}} + \mathbf{c}) |0\rangle + (\mathbf{Z}\mathbf{c}_q - \mathbf{c}_p)\mathbf{R}_{\mathbf{Z},\mathbf{c}} |0\rangle \\ &= \mathbf{R}_{\mathbf{Z},\mathbf{c}}(-\mathbf{Z}, \mathbf{I})\begin{pmatrix} \mathbf{U}^{-1/2}\hat{\mathbf{q}} + \mathbf{c}_q \\ \mathbf{V}\mathbf{U}^{-1/2}\hat{\mathbf{q}} + \mathbf{U}^{1/2}\hat{\mathbf{p}} + \mathbf{c}_p \end{pmatrix} \\ &|0\rangle + (\mathbf{Z}\mathbf{c}_q - \mathbf{c}_p)\mathbf{R}_{\mathbf{Z},\mathbf{c}} |0\rangle \\ &= \mathbf{R}_{\mathbf{Z},\mathbf{c}}(-i\mathbf{U}^{1/2}\hat{\mathbf{q}} + \mathbf{U}^{1/2}\hat{\mathbf{p}} - \mathbf{Z}\mathbf{c}_q + \mathbf{c}_p) |0\rangle \\ &+ (\mathbf{Z}\mathbf{c}_q - \mathbf{c}_p)\mathbf{R}_{\mathbf{Z},\mathbf{c}} |0\rangle \\ &= \mathbf{R}_{\mathbf{Z},\mathbf{c}}(-\mathbf{Z}\mathbf{c}_q + \mathbf{c}_p) |0\rangle + (\mathbf{Z}\mathbf{c}_q - \mathbf{c}_p)\mathbf{R}_{\mathbf{Z},\mathbf{c}} |0\rangle \\ &= (-\mathbf{Z}\mathbf{c}_q + \mathbf{c}_p)\mathbf{R}_{\mathbf{Z},\mathbf{c}} |0\rangle + (\mathbf{Z}\mathbf{c}_q - \mathbf{c}_p)\mathbf{R}_{\mathbf{Z},\mathbf{c}} |0\rangle \\ &= \mathbf{0}.\end{aligned}\quad (\text{D6})$$

In the third to last line, we have used the fact that $-\mathbf{Z}\mathbf{c}_q + \mathbf{c}_p$ is a c -number vector, which commutes with $\mathbf{R}_{\mathbf{Z},\mathbf{c}}$. In

addition, we also used the facts that $\mathbf{Z} = \mathbf{V} + i\mathbf{U}$ and $(-i\hat{\mathbf{q}} + \hat{\mathbf{p}}) |0\rangle = -i\sqrt{2}\hat{\mathbf{a}} |0\rangle = 0$. Therefore, the Gaussian graph state with complex adjacency matrix \mathbf{Z} and mode shift center \mathbf{c} has complex nullifiers

$$(\hat{\mathbf{p}} - \mathbf{c}_p - \mathbf{Z}(\hat{\mathbf{q}} - \mathbf{c}_q)) |\psi_{\mathbf{Z},\mathbf{c}}\rangle = \mathbf{0}.\quad (\text{D7})$$

We now restrict to the case $\mathbf{V} = \mathbf{0}$ and $\mathbf{c}_p = \mathbf{0}$, which are the class of states we consider in our numerical experiments. Then, the nullifiers are of the form

$$(\hat{\mathbf{p}} - i\mathbf{U}\hat{\mathbf{q}} + i\mathbf{U}\mathbf{c}_q) |\psi_{\mathbf{Z},\mathbf{c}}\rangle = \mathbf{0}.\quad (\text{D8})$$

We also restrict to symplectic transformations of the quadratures of the form

$$\mathbf{S} = \begin{pmatrix} \mathbf{W}^\top & \mathbf{0} \\ \mathbf{0} & \mathbf{W}^{-1} \end{pmatrix},\quad (\text{D9})$$

where \mathbf{W} is some assumed invertible matrix. We restrict to operations of this form to more efficiently allow for the classical simulation of the contextual RNN using Gaussian operations of an identical form, as discussed in Ref. [71] and Appendix D 2. After performing the quantum operation described by such a symplectic matrix, the nullifier for the whole system is updated as $(\mathbf{W}^{-1}\hat{\mathbf{p}} - i\mathbf{U}\mathbf{W}^\top\hat{\mathbf{q}} + i\mathbf{U}\mathbf{c}_q) |\psi\rangle = \mathbf{0}$, which is equivalent to $(\hat{\mathbf{p}} - i\mathbf{W}\mathbf{U}\mathbf{W}^\top\hat{\mathbf{q}} + i\mathbf{W}\mathbf{U}\mathbf{c}_q) |\psi\rangle = \mathbf{0}$. Homodyne detection of the position quadrature on m qumodes is then, in general, a multivariate Gaussian random variable that is centered at $\boldsymbol{\Pi}_Y\mathbf{W}^{-1\top}\mathbf{c}_q$ with variance $\boldsymbol{\Pi}_Y\mathbf{U}^{-1}\boldsymbol{\Pi}_Y^\top$, where $\boldsymbol{\Pi}_Y$ is the projection operator onto the subspace of the m qumodes being measured. To make the training of our Gaussian models more stable—and to maintain simulability with GKP input states, as is done in Appendix D 2—we assume in our numerical simulations that there is an implicit large scaling for \mathbf{U} , \mathbf{c}_q such that this variance is small. After the measurement, the hidden state at the i th step is updated to a generalized graph state with adjacency matrix $\boldsymbol{\Pi}_H\mathbf{U}\boldsymbol{\Pi}_H^\top$ and mode shift $\boldsymbol{\Pi}_H\mathbf{W}^{-1\top}\mathbf{c}_q$, where $\boldsymbol{\Pi}_H$ is the projection operator onto the subspace of the latent n qumodes.

2. Gaussian operations on GKP states

Our methods for the simulation of (restricted) Gaussian operations on GKP states are similar to the methods used in Ref. [71]. We restrict to (unnormalized) states of the form

$$|\psi\rangle = \sum_{\ell \in \mathbb{Z}^n} |\psi_\ell\rangle,\quad (\text{D10})$$

where each $|\psi_\ell\rangle$ is a Gaussian state with nullifiers given by

$$\hat{\mathbf{n}}_\ell = \epsilon\hat{\mathbf{p}} - \mathbf{Z}(\hat{\mathbf{q}} - \mathbf{c}_q - \mathbf{L}\ell).\quad (\text{D11})$$

We assume that $\epsilon \rightarrow 0^+$, such that all $|\psi_\ell\rangle$ are approximately orthogonal. Time evolution is simulated as in

Appendix D 1, simultaneously for all $|\psi_\ell\rangle$. As all $|\psi_\ell\rangle$ are approximately orthogonal, measurement is simulated via choosing ℓ uniformly at random, and performing the corresponding measurement. In principle, using many measurements (over multiple instances of the state), one can read out \mathbf{L} and \mathbf{c}_q ; these are the measurement results we use in our numerical experiments, as described in Appendix E 2. For any given measurement on a subset of the modes, the postmeasurement state on the remainder of the modes is just the uniform superposition over all ℓ consistent with the resulting measurement outcome \mathbf{y} .

To make this latter observation more concrete, assume after evolution under \mathbf{S} of the form of Eq. (D9) that the nullifiers of $|\psi_\ell\rangle$ are (see Appendix E)

$$\hat{\mathbf{n}}_\ell = \epsilon \hat{\mathbf{p}} - \mathbf{WZW}^\top (\hat{\mathbf{q}} - \mathbf{W}^{-1\top} \mathbf{c}_q - \mathbf{W}^{-1\top} \mathbf{L} \ell). \quad (\text{D12})$$

We wish to find all ℓ consistent with the position measurement result \mathbf{y} (in the limit $\epsilon \rightarrow 0^+$); that is, all ℓ such that

$$\mathbf{\Pi}_Y (\mathbf{W}^{-1\top} \mathbf{c}_q + \mathbf{W}^{-1\top} \mathbf{L} \ell) = \mathbf{y}, \quad (\text{D13})$$

where $\mathbf{\Pi}_Y$ is the projector onto the m mode space being measured. Let H label the space of the other n modes, and the projector onto this space $\mathbf{\Pi}_H$. Writing (with the assumptions that $\tilde{\mathbf{W}}_{YY}$ and \mathbf{W}_{HH} are full rank)

$$\mathbf{W}^{-1\top} = \begin{pmatrix} \tilde{\mathbf{W}}_{HH} & \tilde{\mathbf{W}}_{HY} \\ \tilde{\mathbf{W}}_{YH} & \tilde{\mathbf{W}}_{YY} \end{pmatrix}, \quad (\text{D14})$$

$$\mathbf{W}^\top = \begin{pmatrix} \mathbf{W}_{HH} & \mathbf{W}_{HY} \\ \mathbf{W}_{YH} & \mathbf{W}_{YY} \end{pmatrix}, \quad (\text{D15})$$

$$\ell = \begin{pmatrix} \ell_H \\ \ell_Y \end{pmatrix}, \quad (\text{D16})$$

and assuming the \mathbf{L} (assumed full rank) is of the form

$$\mathbf{L} = \begin{pmatrix} \mathbf{L}_{HH} & 0 \\ 0 & \mathbf{L}_{YY} \end{pmatrix}, \quad (\text{D17})$$

we find from Eq. (D13) that all ℓ consistent with the measurement result satisfy

$$\mathbf{L}_{YY} \ell_Y = \tilde{\mathbf{W}}_{YY}^{-1} (\mathbf{y} - \mathbf{\Pi}_Y \mathbf{W}^{-1\top} \mathbf{c}_q - \tilde{\mathbf{W}}_{YH} \mathbf{L}_{HH} \ell_H). \quad (\text{D18})$$

Assuming that the entries of \mathbf{L}_{YY} are sufficiently small, up to any given machine precision, the ℓ_H are in one-to-one

correspondence with the ℓ consistent with the measurement result. Furthermore, for all such ℓ ,

$$\begin{aligned} & \mathbf{\Pi}_H (\mathbf{W}^{-1\top} \mathbf{c}_q + \mathbf{W}^{-1\top} \mathbf{L} \ell) \\ &= \mathbf{\Pi}_H \mathbf{W}^{-1\top} \mathbf{c}_q + \tilde{\mathbf{W}}_{HH} \mathbf{L}_{HH} \ell_H + \tilde{\mathbf{W}}_{HY} \mathbf{L}_{YY} \ell_Y \\ &= \mathbf{\Pi}_H \mathbf{W}^{-1\top} \mathbf{c}_q + \tilde{\mathbf{W}}_{HH} \mathbf{L}_{HH} \ell_H \\ &+ \tilde{\mathbf{W}}_{HY} \tilde{\mathbf{W}}_{YY}^{-1} (\mathbf{y} - \mathbf{c}_q - \tilde{\mathbf{W}}_{YH} \mathbf{L}_{HH} \ell_H) \\ &= \mathbf{\Pi}_H \mathbf{W}^{-1\top} \mathbf{c}_q + (\tilde{\mathbf{W}}_{HH} - \tilde{\mathbf{W}}_{HY} \tilde{\mathbf{W}}_{YY}^{-1} \tilde{\mathbf{W}}_{YH}) \mathbf{L}_{HH} \ell_H \\ &+ \tilde{\mathbf{W}}_{HY} \tilde{\mathbf{W}}_{YY}^{-1} (\mathbf{y} - \mathbf{\Pi}_Y \mathbf{W}^{-1\top} \mathbf{c}_q) \\ &= \mathbf{\Pi}_H \mathbf{W}^{-1\top} \mathbf{c}_q + \mathbf{W}_{HH}^{-1} \mathbf{L}_{HH} \ell_H \\ &+ \tilde{\mathbf{W}}_{HY} \tilde{\mathbf{W}}_{YY}^{-1} (\mathbf{y} - \mathbf{\Pi}_Y \mathbf{W}^{-1\top} \mathbf{c}_q). \end{aligned} \quad (\text{D19})$$

Using the appropriate displacement operator to remove the final term of Eq. (D19), then yields the effective transformation

$$\mathbf{c}_q \mapsto \mathbf{\Pi}_H \mathbf{W}^{-1\top} \mathbf{c}_q, \quad (\text{D20})$$

$$\mathbf{L} \mapsto \mathbf{W}_{HH}^{-1} \mathbf{L}_{HH}. \quad (\text{D21})$$

APPENDIX E: DETAILS OF THE NUMERICAL SIMULATIONS

We now discuss the details of our numerical simulations. For all models, we study the performance in modeling a standard Spanish-to-English data set [46]. In the main text, we considered the performance over five independent training runs for each model. For each model and each n , the training set is taken to be a random sample of 80% of the data set, and the test set 20%. Each model is trained for 80 epochs, with a batch size of 64. To map the words in this data set to vectors (taken to also be of dimension n , the model dimension), we use the Keras [73] implementation of word2vec (“TextVectorization”) adapted to the data, with a maximum vocabulary size of 5000. The first 5000 most frequent words are mapped to distinct integers, and other words are mapped to a unique token “[Unk].” At the beginning and end of each sentence, we add unique “[Begin]” and “[End]” tokens. For the recurrent translation models, such as the GRU RNN or the CRNN, we do not need to set the length of the sentences. For the transformer model, we set the sentence length to be 20 words. If the sentence is shorter than 20 words, the additional token “[Pad]” is added to the sentence. For the rare case when the sentence contains more than 20 words, additional words are truncated. All networks are trained using Adam [74] (with a learning rate of 10^{-3} , $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-7}$), trained on the forward empirical cross entropy (as the backward empirical cross entropy is difficult to train on).

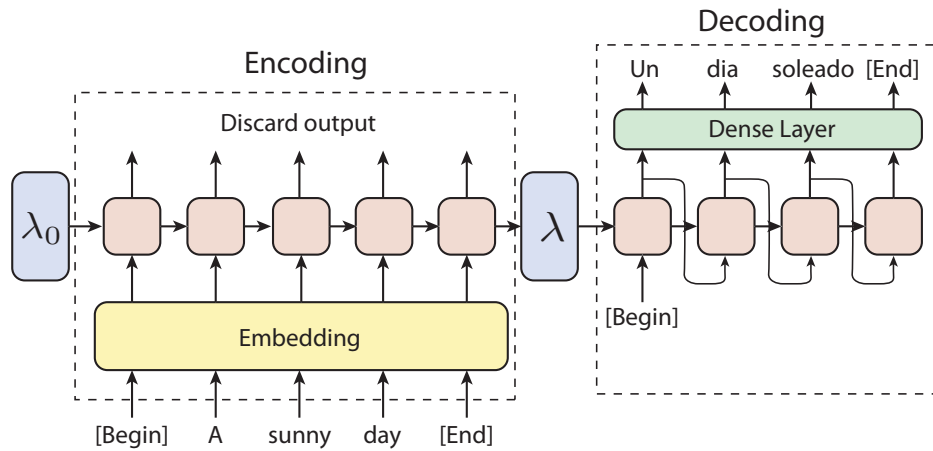


FIG. 6. An overview of the recurrent models we study. Each red box represents the recurrent cell, and the variational parameters in the recurrent cells are shared within the encoder and decoder. Here λ_0 is a random fixed hidden memory vector. In the decoding phase, the output of each recurrent cell is also treated as the input for the next recurrent cell.

1. Classical sequence models

We study three standard classical sequence models in our numerical experiments: an implementation of an orthogonal recurrent neural network [45], an implementation of a network using GRUs [37], and an implementation of a transformer [38]. The first two networks are trained in a seq2seq configuration [30]; the models autoregressively map the input sequence to a latent space, which is then autoregressively decoded. For the orthogonal recurrent neural network, we use the implementation of Ref. [45], with network capacity equal to the model dimension. Illustrations of these architectures are given in Fig. 6.

The transformer models we considered follow the standard construction of Ref. [38]. To fairly compare against the shallow RNN cells we consider, we use a single transformer encoder and decoder layer for each model. Our implementation uses a trained positional embedding with uniform initialization, and the encoders and decoders use ReLU activations in the feedforward network layers, Glorot weight initialization, zero bias initialization, no dropout, and a single head. Each encoder and decoder layer is followed by the layer normalization implementation of Keras [73] with its default parameters. The final layer normalization of the decoder of each transformer we considered is followed by a dense layer with a softmax activation function.

2. Quantum sequence models

Based on the discussion in Appendix D, we simulate both the Gaussian RNN and the contextual RNN described in the main text. The training and architecture of these models are identical to those of the orthogonal neural network described in Appendix E 1, other than the structure of each unit cell of the recurrent network. Once again, Fig. 6 describes the overall architecture of the models, and Fig. 7

describes the recurrent cell of these models. For the Gaussian model, the simulated homodyne position measurements are what we use for our cell outputs; for the CRNN, we simulate the lattice (and position measurement) readout procedure described at the end of Appendix D 2. The pseudocode for the cells of both are given in Algorithm 1; for the Gaussian model, one takes $\mathbf{J}_0, \mathbf{K} = \mathbf{0}$. By considering the Gaussian RNN cell as a limit of the contextual RNN cell, and by not training \mathbf{K} or \mathbf{J}_0 in the contextual RNN, we are able to maintain identical parameter counts for the two models. For the contextual RNN, \mathbf{J}_0 is fixed to the identity, and \mathbf{K} is the result of an untrained dense layer \mathbf{h} applied to the cell input (with uniform Glorot initialization and linear activation function, and biased such that \mathbf{K} has mean the

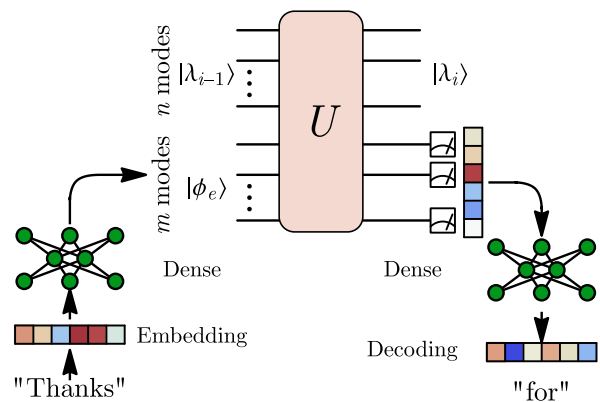


FIG. 7. One recurrent cell of the quantum recurrent architectures. Note that the only trained part of the dense network at the input of each recurrent cell are displacements in phase space acting on $|\phi_e\rangle$, in order to keep the number of trainable parameters in line with the GRU RNN at the same n . Here $|\phi_e\rangle$ is a Gaussian state for the Gaussian model, and a GKP state for the CRNN.

```

Input: // cell inputs
   $n \times n$  latent graph adjacency matrix  $\mathbf{A}_{i-1}$ 
   $n \times n$  lattice  $\mathbf{J}_{i-1}$ 
   $n \times 1$  stabilizer phases  $\boldsymbol{\alpha}_{i-1}$ 
   $m \times 1$  input  $\mathbf{x}_i$ 

  // trainable parameters
   $(n+m) \times (n+m)$  weight matrix  $\mathbf{W}$ 
  Dense layers  $\mathbf{f}, \mathbf{g}$ 

  // constants
  Dense layers  $\mathbf{h}, \mathbf{r}$ 
  Projectors onto latent and input spaces, respectively,  $\Pi_H, \Pi_Y$ 
Output:  $n \times n$  latent graph adjacency matrix  $\mathbf{A}_i$ 
   $n \times n$  lattice  $\mathbf{J}_i$ 
   $n \times 1$  stabilizer phases  $\boldsymbol{\alpha}_i$ 
   $m \times 1$  measurement outcome  $\mathbf{y}$ 

begin
   $\boldsymbol{\alpha} \leftarrow \boldsymbol{\alpha}_{i-1} + \mathbf{A}_{i-1}^{-1} \mathbf{f}(\mathbf{x}_i);$  // perform mode shifts using a general function  $f$ 
   $\boldsymbol{\beta} \leftarrow \mathbf{g}(\mathbf{x}_i);$  // prepare the state associated with the input register
   $\mathbf{K} \leftarrow \mathbf{h}(\mathbf{x}_i);$ 
   $\mathbf{S} \leftarrow \mathbf{r}(\mathbf{x}_i);$ 
   $\mathbf{B} = \mathbf{S}\mathbf{S}^\top;$  // ensure the adjacency matrix is positive semidefinite
   $\mathbf{U} \leftarrow \mathbf{A}_{i-1} \oplus \mathbf{B};$  // consider the tensor product of the latent and input states
   $\boldsymbol{\gamma} \leftarrow \boldsymbol{\alpha} \oplus \boldsymbol{\beta};$ 
   $\mathbf{L} \leftarrow \mathbf{J}_{i-1} \oplus \mathbf{K};$ 
   $\mathbf{U} \leftarrow \mathbf{W}\mathbf{U}\mathbf{W}^\top;$  // transform the graph state by performing a Gaussian operation
   $\boldsymbol{\gamma} \leftarrow \mathbf{W}^{-1\top} \boldsymbol{\gamma};$ 
   $\mathbf{L} \leftarrow \mathbf{W}^{-1\top} \mathbf{L};$ 
   $\mathbf{y} \leftarrow \Pi_Y \mathbf{L}, \Pi_Y \boldsymbol{\gamma};$  // read out the lattice and stabilizer phases
   $\mathbf{A}_i \leftarrow \Pi_H \mathbf{U} \Pi_H^\top;$  // project out the measured register
   $\mathbf{J}_i \leftarrow (\Pi_H \mathbf{W} \Pi_H^\top)^{-1} \mathbf{J}_{i-1};$ 
   $\boldsymbol{\alpha}_i \leftarrow \Pi_H \boldsymbol{\gamma};$ 

```

Algorithm 1. Contextual RNN cell

identity). Specializing to the notation of Algorithm 1: \mathbf{r} is similar; \mathbf{f} and \mathbf{g} are similar, except with no bias.

3. Time complexity

We now discuss the time complexity of implementing a CRNN, both as a quantum model implemented on a quantum computer and as a quantum-inspired classical algorithm. On a quantum computer, each cell of the CRNN we consider in our proofs of an expressivity separation can be implemented in depth $O(n)$, assuming access to the fixed ancilla state $|a\rangle$ used for non-Gaussian measurement. This further decreases to $O(1)$ time if one assumes access to quantum FAN-OUT [75]. More general Gaussian operations can also be implemented in depth $O(n)$ utilizing a swap network [76].

Examining Algorithm 1, it is easy to see that our algorithm simulates inference on a CRNN with model dimension n with time complexity $O(n^\omega)$. Here, ω is the matrix multiplication exponent, with best-known bounds $2 \leq \omega < 2.37286$ [77]. Our results show that on certain tasks, such a CRNN of model dimension n performs on par with, e.g., GRU RNNs with model dimension $\Omega(n^2)$, which performs inference in time $\Omega(n^4)$ due to matrix-vector multiplications present in the model [37]. Thus, our classical simulation of CRNNs may be thought of as a quantum-inspired classical model that, though it is not efficient as implementing a CRNN on a quantum computer, is asymptotically more time efficient in inference and training than typical RNNs with an n^2 -dimensional latent space. Of course, our Algorithm 1 relies on matrix inversion. Though asymptotically matrix inversion takes time $O(n^\omega)$, unlike matrix multiplication it has poor GPU implementations

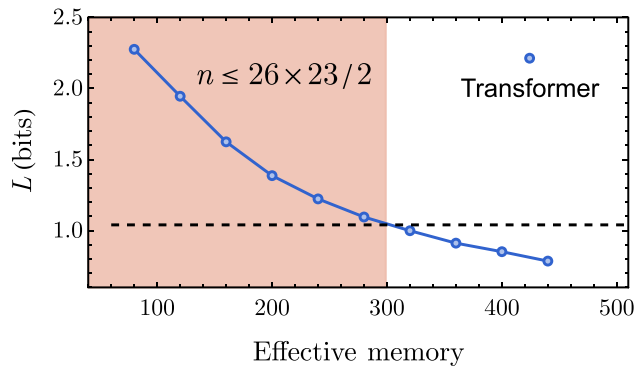


FIG. 8. The performances of an $n = 26$ CRNN model and the transformer models. The dashed line shows the converged forward empirical cross entropy (L) for the $n = 26$ CRNN model. The dimension of the latent space (labeled “Effective memory”) of a transformer is the model dimension multiplied by the length of the (padded) input sentences. The red region labeled “ $n \leq 26 \times 23/2$ ” is where the dimension of the memory of the transformer is at most $n(n - 3)/2$, where n is the model dimension of the CRNN. We find that the transformer models with dimensions roughly equal to $n(n - 3)/2$ achieve losses approximately equal to that of the CRNN.

and thus is often slow in practice. We leave further investigation of the practical utility of these quantum-inspired classical models to future work.

APPENDIX F: SUPPLEMENTARY NUMERICAL RESULTS

We now provide supplemental numerical experiments, comparing CRNNs with transformers [38] and a formulation of linear RNNs dubbed efficient unitary neural networks (EUNNs) [45].

The difficulty in comparing CRNNs and transformers is that the effective memory of a transformer—in the language of Fig. 4(b), the dimension n of the latent space of the model—grows with the sentence length. Thus, we fixed a trained $n = 26$ CRNN, and compare the performance of a transformer at a variety of model dimensions against this model. This also allows us to test for the Spanish-to-English translation task whether, given a fixed performance target achieved by a CRNN of model dimension n , a transformer is only able to achieve the same target given a memory dimension of $n(n - 3)/2$; this is the separation we prove in Theorem 3 on the constructed translation task we consider there. We plot these results in Fig. 8, where we indeed find that the performances of the transformer models achieve that of the $n = 26$ CRNN when their memory is of dimension roughly $n(n - 3)/2$.

We also consider the performance of EUNNs compared with CRNNs, as both are linear models. We constrain the EUNN to be real—as in our simulations of Gaussian models and CRNNs—and call the resulting model an *orthogonal recurrent neural network* (ORNN), using

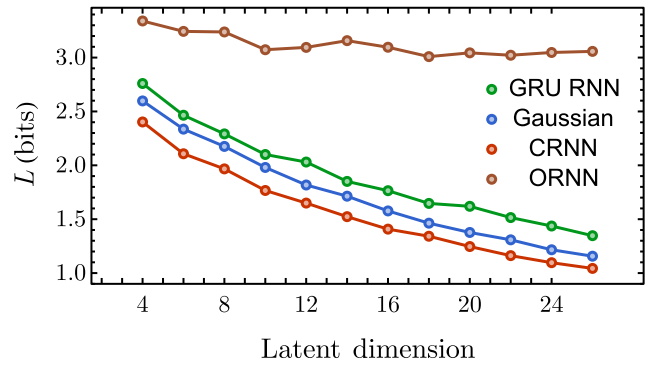


FIG. 9. The converged forward empirical cross entropy (L) as a function of the model dimensions n for ORNNs, and the online models we considered in the main text. We see that ORNNs are greatly outperformed by the other online models we consider at the given task.

the implementation from Ref. [45]. We see in Fig. 9 that CRNNs—and indeed all of the online models we consider—greatly outperform ORNNs at a variety of model dimensions.

-
- [1] F. Arute *et al.*, Quantum supremacy using a programmable superconducting processor, *Nature* **574**, 505 (2019).
 - [2] H.-S. Zhong *et al.*, Quantum computational advantage using photons, *Science* **370**, 1460 (2020).
 - [3] Y. Wu *et al.*, Strong Quantum Computational Advantage Using a Superconducting Quantum Processor, *Phys. Rev. Lett.* **127**, 180501 (2021).
 - [4] H.-S. Zhong *et al.*, Phase-Programmable Gaussian Boson Sampling Using Stimulated Squeezed Light, *Phys. Rev. Lett.* **127**, 180502 (2021).
 - [5] Q. Zhu *et al.*, Quantum computational advantage via 60-qubit 24-cycle random circuit sampling, *Sci. Bull.* **67**, 240 (2022).
 - [6] D. Hangleiter and J. Eisert, Computational advantage of quantum random sampling, [arXiv:2206.04079](https://arxiv.org/abs/2206.04079) [quant-ph] (2022).
 - [7] M. Schuld, I. Sinayskiy, and F. Petruccione, An introduction to quantum machine learning, *Contemp. Phys.* **56**, 172 (2015).
 - [8] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd, Quantum machine learning, *Nature* **549**, 195 (2017).
 - [9] M. H. Amin, E. Andriyash, J. Rolfe, B. Kulchitsky, and R. Melko, Quantum Boltzmann Machine, *Phys. Rev. X* **8**, 021050 (2018).
 - [10] A. Perdomo-Ortiz, M. Benedetti, J. Realpe-Gómez, and R. Biswas, Opportunities and challenges for quantum-assisted machine learning in near-term quantum computers, *Quantum Sci. Technol.* **3**, 030502 (2018).
 - [11] S. Lloyd and C. Weedbrook, Quantum Generative Adversarial Learning, *Phys. Rev. Lett.* **121**, 040502 (2018).

- [12] M. Schuld and N. Killoran, Quantum Machine Learning in Feature Hilbert Spaces, *Phys. Rev. Lett.* **122**, 040504 (2019).
- [13] V. Havlíček, A. D. Córcoles, K. Temme, A. W. Harrow, A. Kandala, J. M. Chow, and J. M. Gambetta, Supervised learning with quantum-enhanced feature spaces, *Nature* **567**, 209 (2019).
- [14] N. Killoran, T. R. Bromley, J. M. Arrazola, M. Schuld, N. Quesada, and S. Lloyd, Continuous-variable quantum neural networks, *Phys. Rev. Res.* **1**, 033063 (2019).
- [15] X. Gao, Z.-Y. Zhang, and L.-M. Duan, A quantum machine learning algorithm based on generative models, *Sci. Adv.* **4**, eaat9004 (2018).
- [16] B. Coyle, D. Mills, V. Danos, and E. Kashefi, The Born supremacy: Quantum advantage and training of an Ising Born machine, *Npj Quantum Inf.* **6**, 1 (2020).
- [17] Y. Du, M.-H. Hsieh, T. Liu, and D. Tao, Expressive power of parametrized quantum circuits, *Phys. Rev. Res.* **2**, 033125 (2020).
- [18] R. Sweke, J.-P. Seifert, D. Hangleiter, and J. Eisert, On the quantum versus classical learnability of discrete distributions, *Quantum* **5**, 417 (2021).
- [19] E. Farhi and H. Neven, Classification with quantum neural networks on near term processors, [arXiv:1802.06002](https://arxiv.org/abs/1802.06002) [quant-ph] (2018).
- [20] L. Hu, S.-H. Wu, W. Cai, Y. Ma, X. Mu, Y. Xu, H. Wang, Y. Song, D.-L. Deng, C.-L. Zou, and L. Sun, Quantum generative adversarial learning in a superconducting quantum circuit, *Sci. Adv.* **5**, eaav2761 (2019).
- [21] E. R. Anschuetz and C. Zanoni, Near-term quantum-classical associative adversarial networks, *Phys. Rev. A* **100**, 052327 (2019).
- [22] Y. Liu, S. Arunachalam, and K. Temme, A rigorous and robust quantum speed-up in supervised machine learning, *Nat. Phys.* **17**, 1013 (2021).
- [23] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, and H. Neven, Barren plateaus in quantum neural network training landscapes, *Nat. Commun.* **9**, 4812 (2018).
- [24] M. Cerezo, A. Sone, T. Volkoff, L. Cincio, and P. J. Coles, Cost function dependent barren plateaus in shallow parametrized quantum circuits, *Nat. Commun.* **12**, 1791 (2021).
- [25] C. Ortiz Marrero, M. Kieferová, and N. Wiebe, Entanglement-Induced Barren Plateaus, *PRX Quantum* **2**, 040316 (2021).
- [26] J. Napp, Quantifying the barren plateau phenomenon for a model of unstructured variational ansätze, [arXiv:2203.06174](https://arxiv.org/abs/2203.06174) [quant-ph] (2022).
- [27] E. R. Anschuetz, in *International Conference on Learning Representations* (OpenReview.net, Amherst, USA, 2022).
- [28] X. You and X. Wu, in *Proceedings of the 38th International Conference on Machine Learning*, Proceedings of Machine Learning Research, Vol. 139, edited by M. Meila and T. Zhang (PMLR, Cambridge, USA, 2021), p. 12144.
- [29] E. R. Anschuetz and B. T. Kiani, Quantum variational algorithms are swamped with traps, *Nat. Commun.* **13**, 7760 (2022).
- [30] I. Sutskever, O. Vinyals, and Q. V. Le, in *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14 (MIT Press, Cambridge, MA, USA, 2014), p. 3104.
- [31] J. J. Hopfield, Neural networks and physical systems with emergent collective computational abilities, *Proc. Natl. Acad. Sci. U.S.A.* **79**, 2554 (1982).
- [32] A. Mari and J. Eisert, Positive Wigner Functions Render Classical Simulation of Quantum Computation Efficient, *Phys. Rev. Lett.* **109**, 230503 (2012).
- [33] P. Cheeseman, J. Kelly, M. Self, J. Stutz, W. Taylor, and D. Freeman, Autoclass: A Bayesian classification system, in *Machine Learning Proceedings 1988*, edited by J. Laird (Morgan Kaufmann, San Francisco (CA), 1988), p. 54.
- [34] D. Gottesman, A. Kitaev, and J. Preskill, Encoding a qubit in an oscillator, *Phys. Rev. A* **64**, 012310 (2001).
- [35] A. R. Plastino and A. Cabello, State-independent quantum contextuality for continuous variables, *Phys. Rev. A* **82**, 022114 (2010).
- [36] S. Hochreiter and J. Schmidhuber, Long short-term memory, *Neural Comput.* **9**, 1735 (1997).
- [37] K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (Association for Computational Linguistics, Doha, Qatar, 2014), p. 1724.
- [38] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17 (Curran Associates Inc., Red Hook, NY, USA, 2017), p. 6000.
- [39] G. K. Dziugaite, Ph.D. thesis, University of Cambridge, 2020.
- [40] A. M. Gleason, Measures on the closed subspaces of a Hilbert space, *J. Math. Mech.* **6**, 885 (1957).
- [41] J. S. Bell, On the problem of hidden variables in quantum mechanics, *Rev. Mod. Phys.* **38**, 447 (1966).
- [42] S. Kochen and E. P. Specker, The problem of hidden variables in quantum mechanics, *J. Math. Mech.* **17**, 59 (1967).
- [43] N. D. Mermin, Simple Unified Form for the Major No-Hidden-Variables Theorems, *Phys. Rev. Lett.* **65**, 3373 (1990).
- [44] X. Gao, E. R. Anschuetz, S.-T. Wang, J. I. Cirac, and M. D. Lukin, Enhancing Generative Models via Quantum Correlations, *Phys. Rev. X* **12**, 021037 (2022).
- [45] L. Jing, Y. Shen, T. Dubcek, J. Peurifoy, S. Skirlo, Y. LeCun, M. Tegmark, and M. Soljačić, in *Proceedings of the 34th International Conference on Machine Learning*, Proceedings of Machine Learning Research, Vol. 70, edited by D. Precup and Y. W. Teh (PMLR, Cambridge, USA, 2017), p. 1733.
- [46] C. Kelly, Tab-delimited bilingual sentence pairs (2021).
- [47] Z. Holmes, K. Sharma, M. Cerezo, and P. J. Coles, Connecting Ansatz Expressibility to Gradient Magnitudes and Barren Plateaus, *PRX Quantum* **3**, 010313 (2022).
- [48] R. Prabhavalkar, K. Rao, T. N. Sainath, B. Li, L. Johnson, and N. Jaitly, in *Proc. Interspeech 2017* (International Speech Communication Association, Grenoble, France, 2017), p. 939.

- [49] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Institute of Electrical and Electronics Engineers, New York, USA, 2015).
- [50] H. Federer, in *Geometric Measure Theory*, edited by B. Eckmann and B. L. van der Waerden (Springer Berlin Heidelberg, Berlin, Heidelberg, 1996), p. 207.
- [51] S. L. Braunstein and P. van Loock, Quantum information with continuous variables, *Rev. Mod. Phys.* **77**, 513 (2005).
- [52] N. Liu, J. Thompson, C. Weedbrook, S. Lloyd, V. Vedral, M. Gu, and K. Modi, Power of one qumode for quantum computation, *Phys. Rev. A* **93**, 052304 (2016).
- [53] C. Calcluth, A. Ferraro, and G. Ferrini, The vacuum provides quantum advantage to otherwise simulatable architectures, [arXiv:2205.09781](https://arxiv.org/abs/2205.09781) [quant-ph] (2022).
- [54] B. Q. Baragiola, G. Pantaleoni, R. N. Alexander, A. Karanjai, and N. C. Menicucci, All-Gaussian Universality and Fault Tolerance with the Gottesman-Kitaev-Preskill Code, *Phys. Rev. Lett.* **123**, 200502 (2019).
- [55] M. A. Nielsen and I. L. Chuang, in *Quantum Computation and Quantum Information: 10th Anniversary Edition* (Cambridge University Press, Cambridge, UK, 2010), p. 216.
- [56] D. E. Gottesman, Ph.D. thesis, California Institute of Technology, 1997.
- [57] S. Aaronson and D. Gottesman, Improved simulation of stabilizer circuits, *Phys. Rev. A* **70**, 052328 (2004).
- [58] A. Karanjai, J. J. Wallman, and S. D. Bartlett, Contextuality bounds the efficiency of classical simulation of quantum processes, [arXiv:1802.07744](https://arxiv.org/abs/1802.07744) [quant-ph] (2018).
- [59] G. J. Butler, J. G. Timourian, and C. Viger, The rank theorem for locally Lipschitz continuous functions, *Can. Math. Bull.* **31**, 217 (1988).
- [60] R. I. Booth, U. Chabaud, and P.-E. Emeriau, Contextuality and Wigner negativity are equivalent for continuous-variable quantum measurements, [arXiv:2111.13218](https://arxiv.org/abs/2111.13218) [quant-ph] (2021).
- [61] J. Haferkamp and J. Bermejo-Vega, Equivalence of contextuality and Wigner function negativity in continuous-variable quantum optics, [arXiv:2112.14788](https://arxiv.org/abs/2112.14788) [quant-ph] (2021).
- [62] J. Pearl, in *Proceedings of the 7th Conference of the Cognitive Science Society, University of California, Irvine, CA, USA* (Cognitive Science Society, Seattle, USA, 1985), p. 329.
- [63] L. E. Baum and T. Petrie, Statistical inference for probabilistic functions of finite state Markov chains, *Ann. Math. Stat.* **37**, 1554 (1966).
- [64] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, in *Advances in Neural Information Processing Systems*, Vol. 27, edited by Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger (Curran Associates, Inc., 2014).
- [65] D. Rezende and S. Mohamed, in *Proceedings of the 32nd International Conference on Machine Learning*, Proceedings of Machine Learning Research, Vol. 37, edited by F. Bach and D. Blei (PMLR, Lille, France, 2015), p. 1530.
- [66] A. J. Kerman, Quantum information processing using quasiclassical electromagnetic interactions between qubits and electrical resonators, *New J. Phys.* **15**, 123011 (2013).
- [67] A. Blais, A. L. Grimsmo, S. M. Girvin, and A. Wallraff, Circuit quantum electrodynamics, *Rev. Mod. Phys.* **93**, 025005 (2021).
- [68] Y. Shi, C. Chamberland, and A. Cross, Fault-tolerant preparation of approximate GKP states, *New J. Phys.* **21**, 093007 (2019).
- [69] Y.-S. Ra, A. Dufour, M. Walschaers, C. Jacquard, T. Michel, C. Fabre, and N. Treps, Non-Gaussian quantum states of a multimode light field, *Nat. Phys.* **16**, 144 (2020).
- [70] N. C. Menicucci, S. T. Flammia, and P. van Loock, Graphical calculus for Gaussian pure states, *Phys. Rev. A* **83**, 042335 (2011).
- [71] C. Calcluth, A. Ferraro, and G. Ferrini, Efficient simulation of Gottesman-Kitaev-Preskill states with Gaussian circuits, [arXiv:2203.11182](https://arxiv.org/abs/2203.11182) [quant-ph] (2022).
- [72] S. D. Bartlett, B. C. Sanders, S. L. Braunstein, and K. Nemoto, Efficient Classical Simulation of Continuous Variable Quantum Information Processes, *Phys. Rev. Lett.* **88**, 097904 (2002).
- [73] F. Chollet *et al.*, Keras, <https://keras.io> (2015).
- [74] D. P. Kingma and J. Ba, in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, edited by Y. Bengio and Y. LeCun (OpenReview.net, Amherst, USA, 2015).
- [75] P. Høyer and R. Špalek, Quantum fan-out is powerful, *Theory Comput.* **1**, 81 (2005).
- [76] I. D. Kivlichan, J. McClean, N. Wiebe, C. Gidney, A. Aspuru-Guzik, G. K.-L. Chan, and R. Babbush, Quantum Simulation of Electronic Structure with Linear Depth and Connectivity, *Phys. Rev. Lett.* **120**, 110501 (2018).
- [77] J. Alman and V. V. Williams, in *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)* (Society for Industrial and Applied Mathematics, Philadelphia, USA, 2021), p. 522.