**Title:** Language models for Quantum Simulation

**Speakers:** Roger Melko

**Collection/Series:** Waterloo-Munich Joint Workshop

**Subject:** Quantum Information

**Date:** October 03, 2024 - 9:45 AM

**URL:** https://pirsa.org/24100055

# Language models for quantum simulation

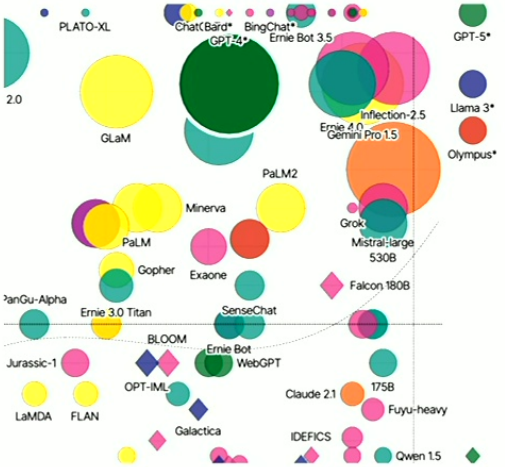
Roger Melko

UNIVERSITY OF WATERLOO

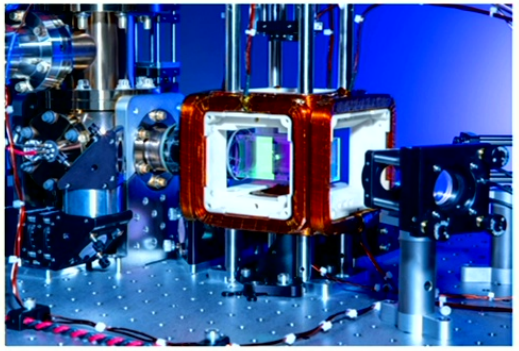
QUANTUM INTELLIGENCE LAB

# Outline

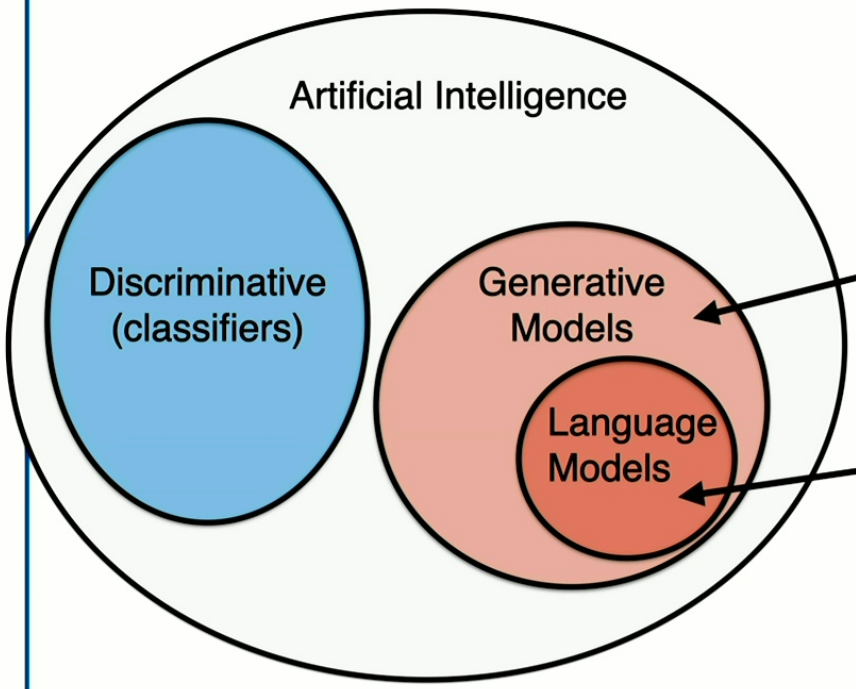
## A crash course in language models




## Quantum simulation & beyond



# The age of language models in AI

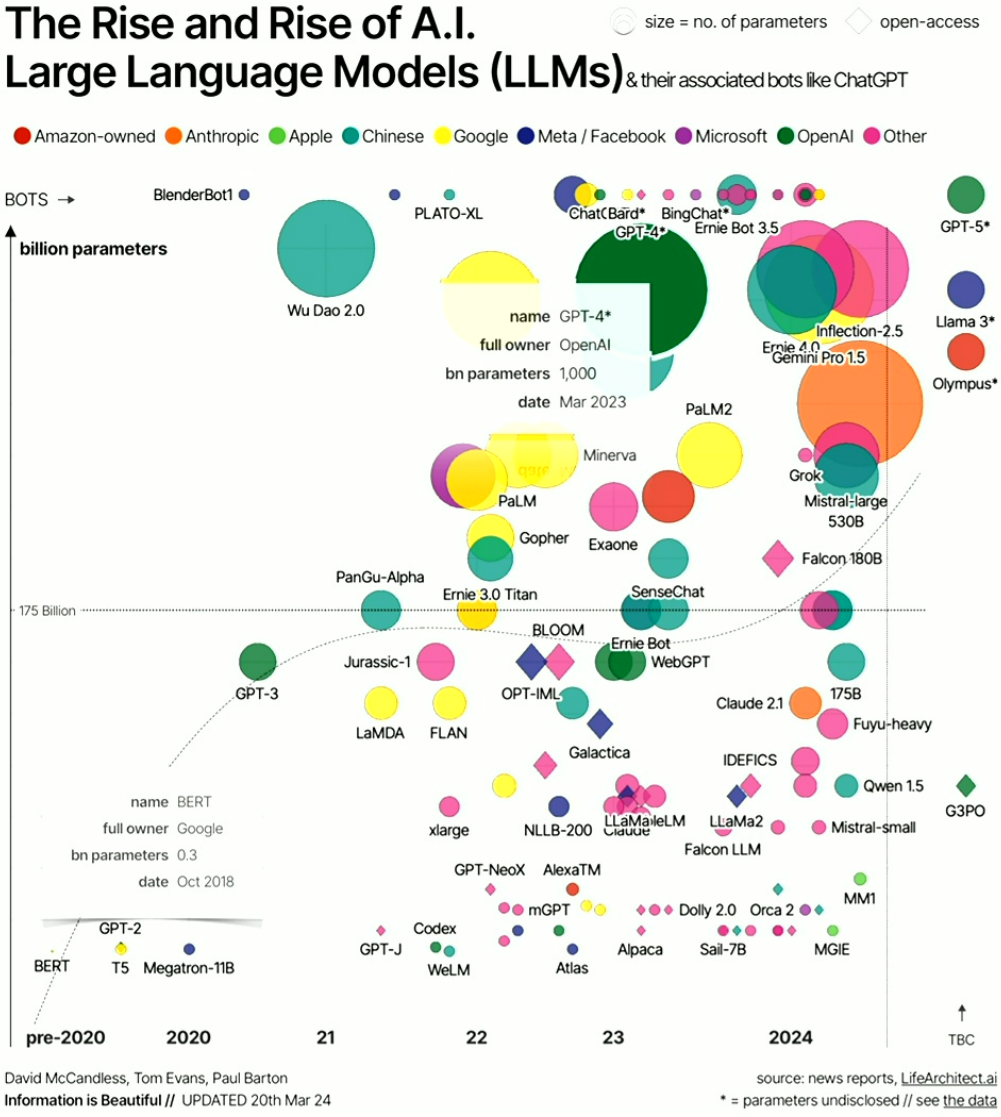



```
/imagine a stormtrooper family
in the style of Norman Rockwell
```

Hickory dickory dock, the mouse ran up the clock, up the clock, up the clock to claim his house."

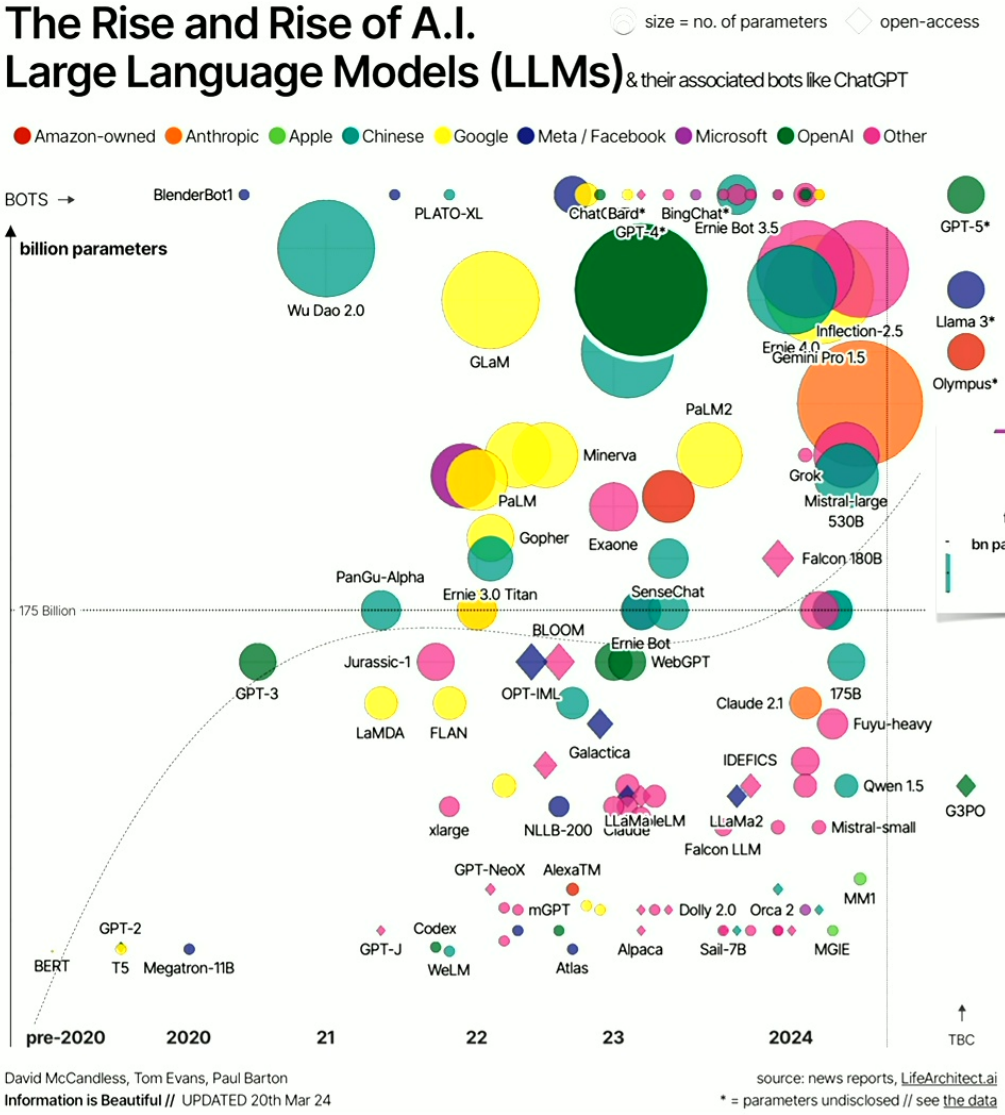
"Let it go," said the Mama, giving me a gentle, stern stare.

Breakthroughs in large language models arose from large **parameterization**, **data**, and **compute**





Typical large-scale DMRG might have
1 to 10 billion parameters



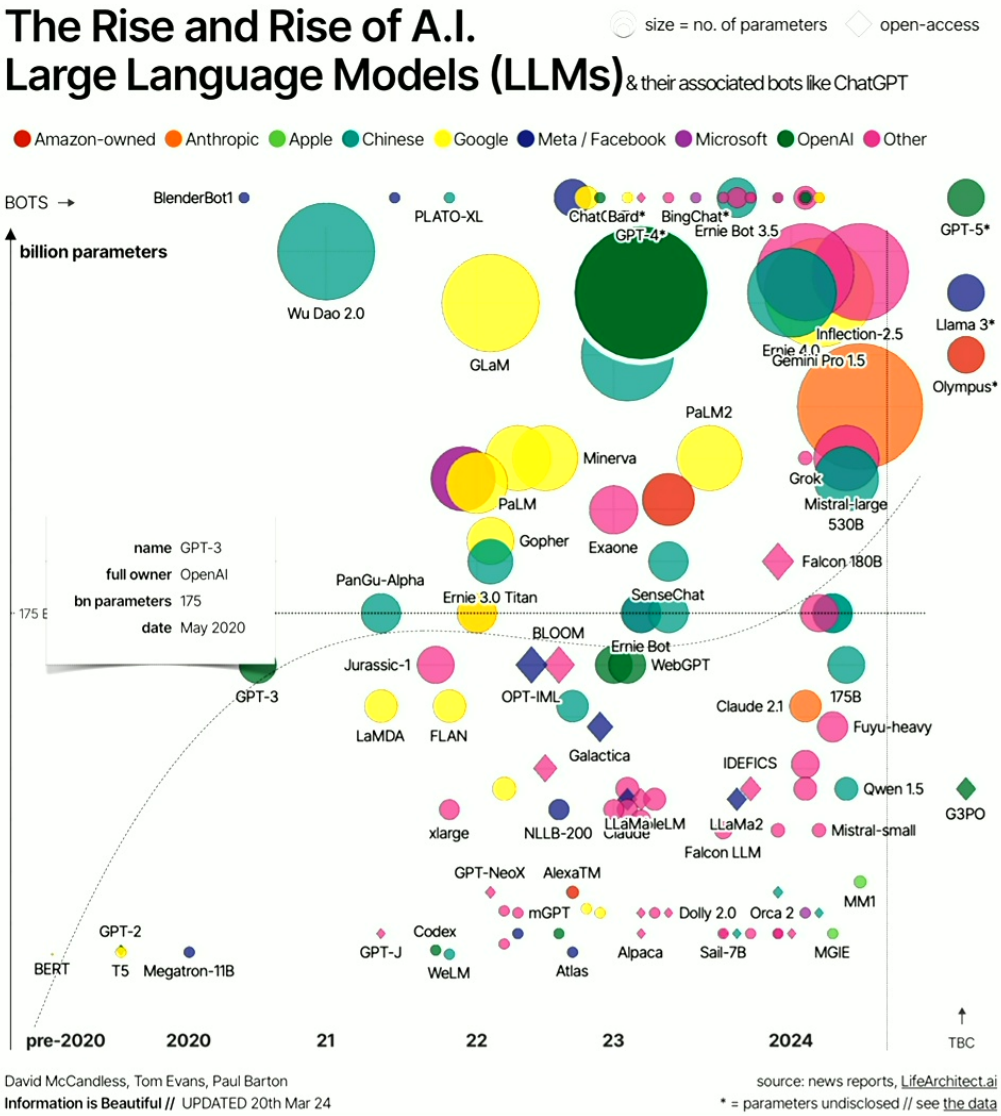

name Grok
note Twitter chatbot trained on Twitter data
full owner Twitter
bn parameters 33
date Nov 2023

## Elon Musk Bets on 100K Nvidia H100 GPUs to Train xAI's Grok

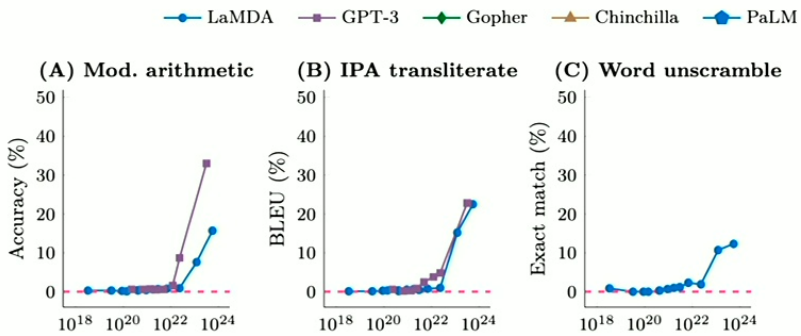
xAI is pushing the limits with what its boss claims is the world's most powerful AI cluster.

## Three Mile Island nuclear plant to help power Microsoft's data-center needs

Unit 1 has been shut down for five years for economic reasons; it is separate from Unit 2, which melted down in 1979.




Scaling Laws for Neural Language Models, arxiv:2001.08361
Emergent Abilities of Large Language Models, arXiv:2206.07682

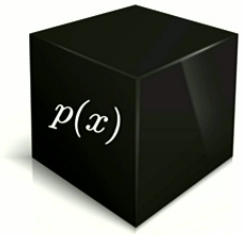



"Emergence is when quantitative changes in a system result in qualitative changes in behavior."

Steinhardt (2022) , Anderson, (1972)

# Generative models: density estimation

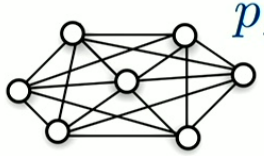
In the typical setting, generative models *learn* to approximate a target probability distribution that underlies a dataset:




$$\mathbf{x}_1 = (1,0,0,1,1,1,0,0,0,0,\cdots,1)$$
$$\mathbf{x}_2 = (1,1,1,0,1,1,0,1,1,1,\cdots,1)$$
$$\mathbf{x}_3 = (0,1,1,0,0,1,0,1,0,1,\cdots,0)$$
$$\vdots$$

Use this data to *train* the optimal parameters in some representation of the unknown target distribution/state

*a model:*



$p_\lambda$

*goal:*

$$p_\lambda(\mathbf{x}) \approx p(\mathbf{x})$$

*loss function:*

$$\mathcal{L} = \langle \log p_\lambda(\mathbf{x}) \rangle_p$$

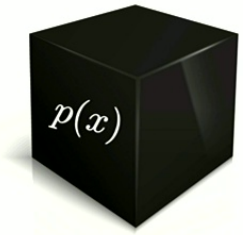
*training:*

$$\lambda' = \lambda - \eta \nabla \mathcal{L}$$

*… inference*

# Generative models: density estimation

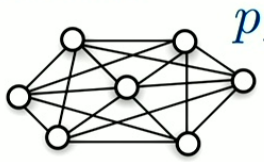
In the typical setting, generative models *learn* to approximate a target probability distribution that underlies a dataset:




$$\mathbf{x}_1 = (1,0,0,1,1,1,0,0,0,0,\cdots,1)$$
$$\mathbf{x}_2 = (1,1,1,0,1,1,0,1,1,1,\cdots,1)$$
$$\mathbf{x}_3 = (0,1,1,0,0,1,0,1,0,1,\cdots,0)$$
$$\vdots$$

Use this data to *train* the optimal parameters in some representation of the unknown target distribution/state

*a model:*



$p_\lambda$

*goal:*

$$p_\lambda(\mathbf{x}) \approx p(\mathbf{x})$$

*loss function:*

$$\mathcal{L} = \langle \log p_\lambda(\mathbf{x}) \rangle_p$$

$$\mathcal{L} = F_\lambda = \langle H_{\text{target}} \rangle_\lambda - TS(p_\lambda)$$

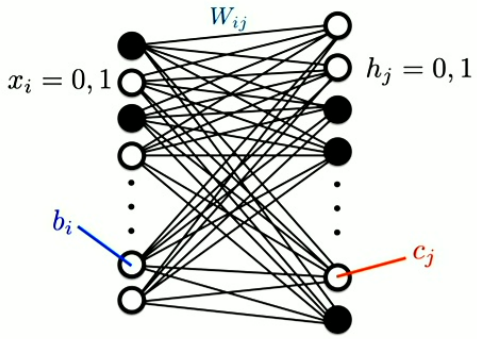
*training:*

$$\lambda' = \lambda - \eta \nabla \mathcal{L}$$

*… inference*

# Computational physics perspectives

## Energy-based (RBM)



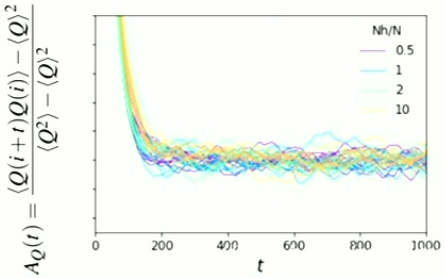

$$p_\lambda(\mathbf{x}, \mathbf{h}) = \frac{1}{Z_\lambda} e^{-E_\lambda(\mathbf{x}, \mathbf{h})}$$

$$p_\lambda(\mathbf{x}) = \sum_{\mathbf{h}} p_\lambda(\mathbf{x}, \mathbf{h})$$

$$E_\lambda(\mathbf{x}, \mathbf{h}) = -\sum_{ij} W_{ij} x_i h_j - \sum_i b_i x_i - \sum_j c_j h_j$$

$\lambda = \{W, b, c\}$ model parameters

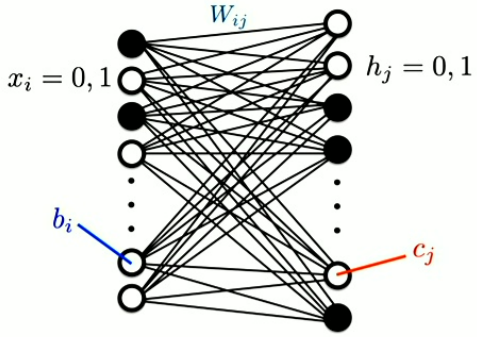
- Requires MCMC sampling
- Generated samples are autocorrelated




Restricted Boltzmann machines in quantum physics
RGM, G. Carleo, J. Carrasquilla & J. I. Cirac
Nature Physics 15, 887 (2019)

# Computational physics perspectives

## Energy-based (RBM)



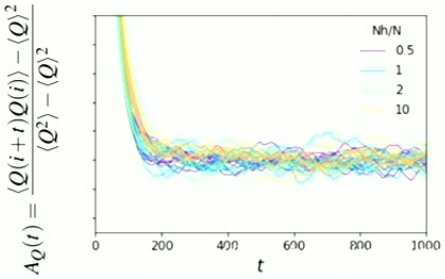

$$p_\lambda(\mathbf{x}, \mathbf{h}) = \frac{1}{Z_\lambda} e^{-E_\lambda(\mathbf{x}, \mathbf{h})}$$

$$p_\lambda(\mathbf{x}) = \sum_{\mathbf{h}} p_\lambda(\mathbf{x}, \mathbf{h})$$

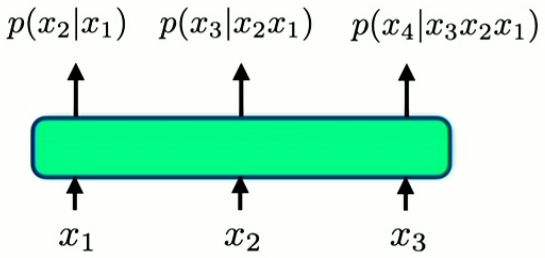
$$E_\lambda(\mathbf{x}, \mathbf{h}) = -\sum_{ij} W_{ij} x_i h_j - \sum_i b_i x_i - \sum_j c_j h_j$$

$\lambda = \{W, b, c\}$ model parameters

- Requires MCMC sampling
- Generated samples are autocorrelated

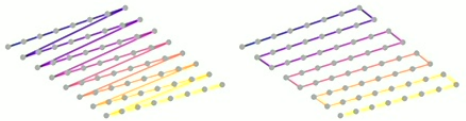



## Autoregressive




$$p_\lambda(x_1, \cdots x_N) = \prod_{i=1}^{N} p(x_i | x_1, \ldots, x_{i-1})$$

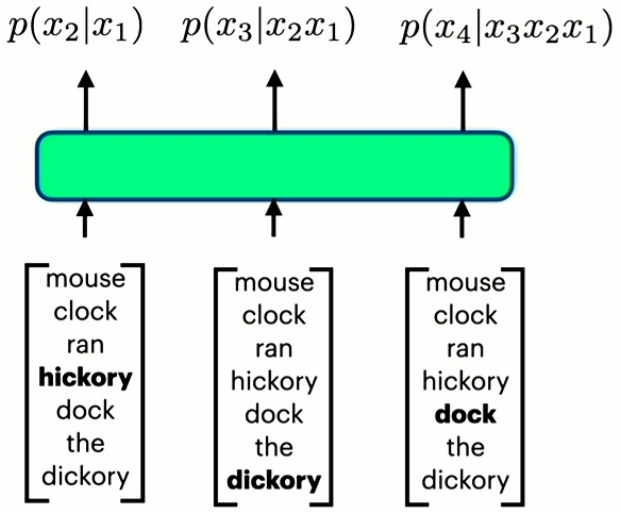
- Joint distribution is normalized
- Requires the definition of a sequence
- Generated samples are uncorrelated, inference time scales: $\mathcal{O}(N), \mathcal{O}(N^2)$

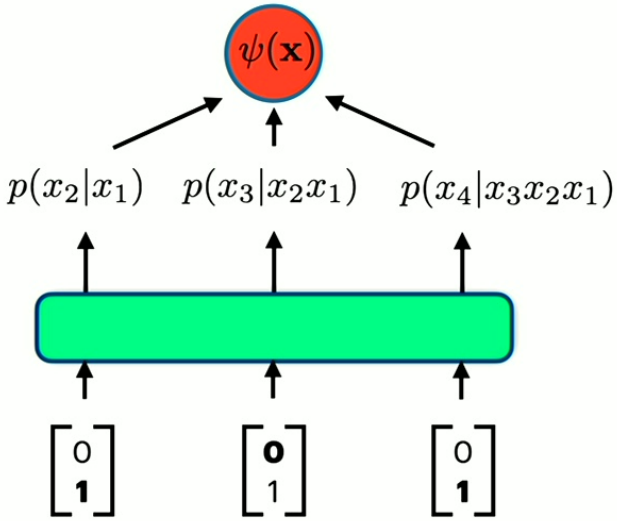


Autoregressive model path dependence near Ising criticality
Yi Hong Teoh, RGM, arXiv:2408.15715

# Sequences: languages and qubits

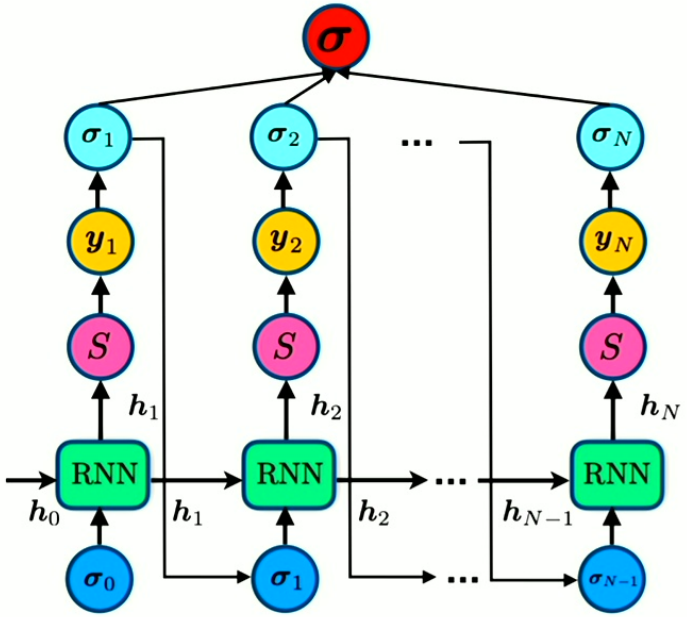
Autoregressive models produce the conditional probability of each *token* (word or qubit) in a sequence



# Recurrent Neural Networks

Lipton, Berkowitz, Elkan, arXiv:1506.00019
Hibat-Allah, Ganahl, Hayward, RGM, Carrasquilla, arXiv:2002.02973




$$\boldsymbol{y}_n = \left[p(\sigma_n = 0|\boldsymbol{\sigma}_{<n}), p(\sigma_n = 1|\boldsymbol{\sigma}_{<n})\right]$$

$$\boldsymbol{y}_n \equiv S\left(U\boldsymbol{h}_n + \mathbf{c}\right) \qquad S(v_j) = \frac{\exp(v_j)}{\sum_i \exp(v_i)}$$

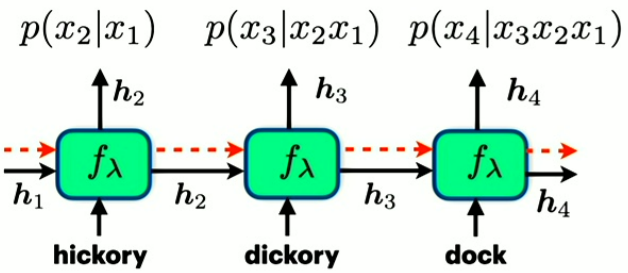
$$\boldsymbol{h}_n = f\left(W[\boldsymbol{h}_{n-1}; \boldsymbol{\sigma}_{n-1}] + \boldsymbol{b}\right)$$

# Recurrent Neural Networks

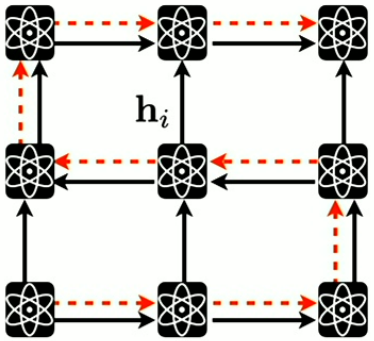
Lipton, Berkowitz, Elkan, arXiv:1506.00019
Hibat-Allah, Ganahl, Hayward, RGM, Carrasquilla, arXiv:2002.02973

- Long-range correlations (context) passed through a hidden state vector

language




qbit arrays



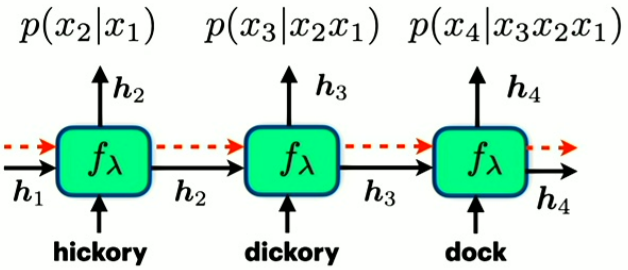

$$\mathbf{h}_i = f_\lambda(\mathbf{h}_{i-1}, x_{i-1})$$

# Recurrent Neural Networks

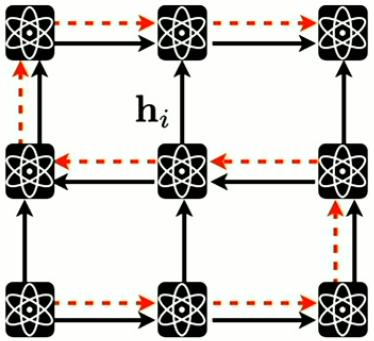
Lipton, Berkowitz, Elkan, arXiv:1506.00019
Hibat-Allah, Ganahl, Hayward, RGM, Carrasquilla, arXiv:2002.02973

- Long-range correlations (context) passed through a hidden state vector

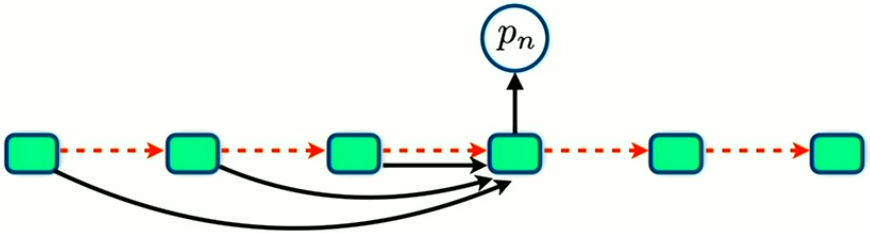
language




qbit arrays

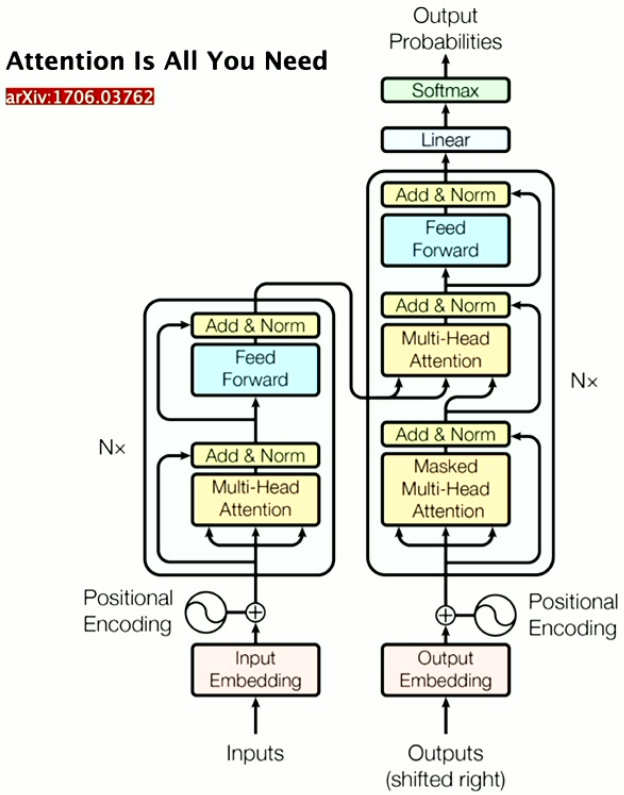


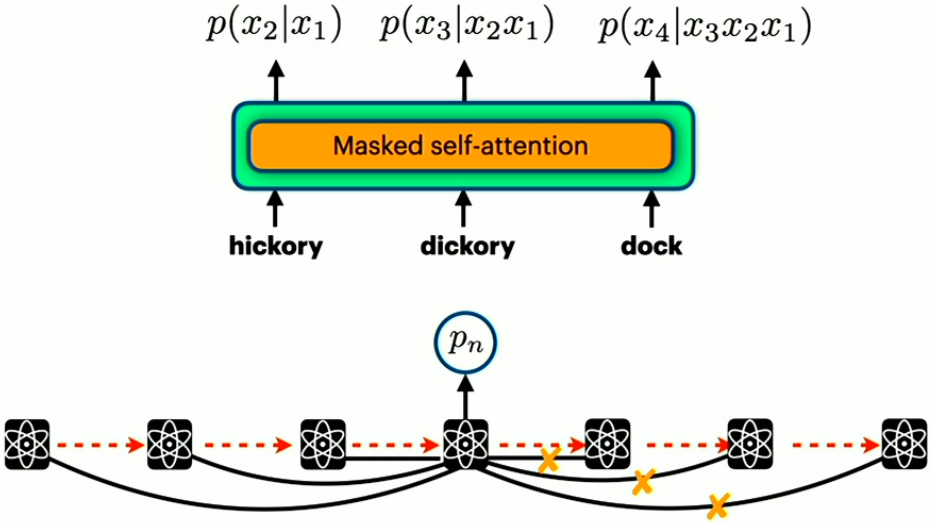

$$\mathbf{h}_i = f_\lambda(\mathbf{h}_{i-1}, x_{i-1})$$



# The Transformer

- The basis of modern Large Language Models (LLMs)



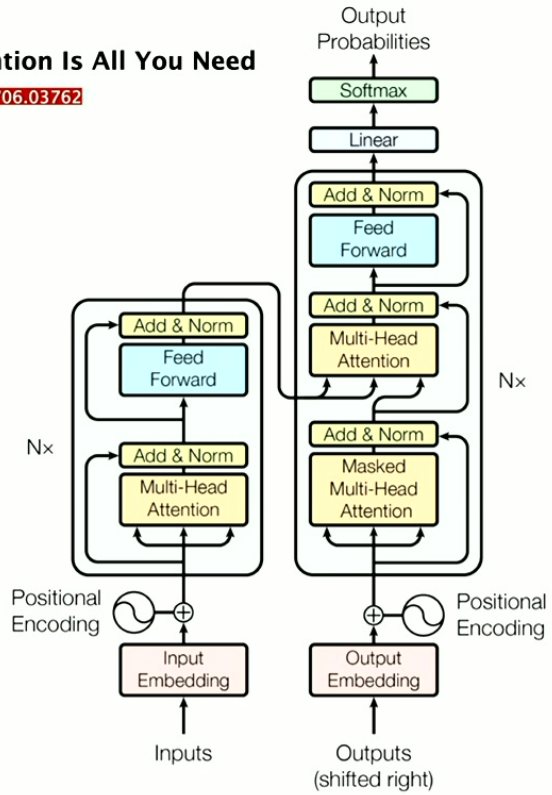

The self-attention and its correlation matrix are useful to introduce correlations between qubits separated at any distance in the quantum system. This is analogous to a two-body Jastrow factor which induces pairwise long-distance correlations between the bare degrees of freedom (i.e. spins, qubits, electrons) in a wavefunction.

Probabilistic Simulation of Quantum Circuits with the Transformer,
J. Carrasquilla, Di Luo, F. Pérez, A. Milsted, B. Clark, M. Volkovs, L. Aolita, Phys. Rev. A 104, 032610 (2021)

# The Transformer

**Attention Is All You Need**

arXiv:1706.03762




arXiv > quant-ph > arXiv:2306.03921

Quantum Physics

*[Submitted on 6 Jun 2023 (v1), last revised 16 Mar 2024 (this version, v2)]*

**Variational Monte Carlo with Large Patched Transformers**

Kyle Sprague, Stefanie Czischek

arXiv > cond-mat > arXiv:2406.00091

Condensed Matter > Disordered Systems and Neural Networks

*[Submitted on 31 May 2024]*

**Transformer neural networks and quantum simulators: a hybrid approach for simulating strongly correlated systems**

Hannah Lange, Guillaume Bornet, Gabriel Emperauger, Cheng Chen, Thierry Lahaye, Stefan Kienle, Antoine Browaeys, Annabelle Bohrdt

arXiv > cond-mat > arXiv:2407.21502

Condensed Matter > Quantum Gases

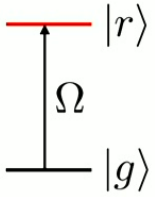
*[Submitted on 31 Jul 2024]*

**Interpretable correlator Transformer for image-like quantum matter data**

Abhinav Suresh, Henning Schlömer, Baran Hashemi, Annabelle Bohrdt

# Rydberg Blockade Hamiltonian

Jaksch, Cirac, Zoller, Rolston, Cote, Lukin, Phys. Rev. Lett. 85, 2208 (2000)
Lukin, Fleischhauer, Cote, Duan, Jaksch, Cirac, Zoller, Phys. Rev. Lett. 87, 037901 (2001)
Fendley, Sengupta, Sachdev, Phys. Rev. B 69, 075106 (2004)




$$H = \Omega \sum_i \sigma_i^x - \Delta \sum_i n_i + \sum_{i<j} V_{ij} n_i n_j$$

$$V(R) = \frac{\Omega}{(R/R_b)^6}$$

$$\sigma^x = |g\rangle\langle r| + |r\rangle\langle g| \qquad n = |r\rangle\langle r|$$

This Hamiltonian is stoquastic:

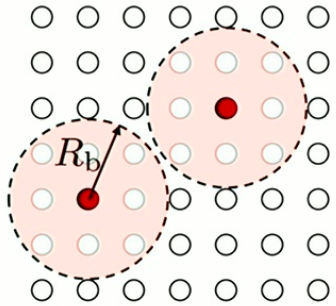
- Perron-Frobenius implies the groundstate is real positive

$$\psi_\lambda(z) = \sqrt{p_\lambda(z)}$$

- Allows for QMC with no sign problem

Stochastic series expansion quantum Monte Carlo for Rydberg arrays
Ejaaz Merali, Isaac J. S. De Vlugt, Roger G. Melko
SciPost Phys. Core 7, 016 (2024) · published 5 April 2024

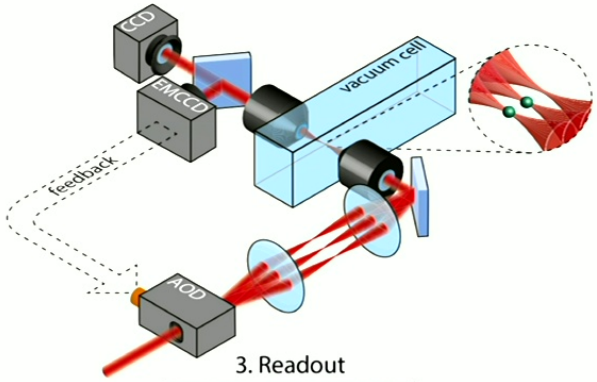
Browaeys, Lahaye, Nature Physics 16, 132 (2020)



# State prep and readout

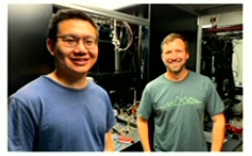
Endres et. al. Science 354, 1024 (2016)

Ebadi et. al. arXiv:2012.12281
Nature 595, 227 (2021)

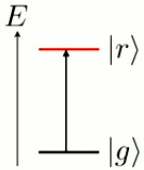


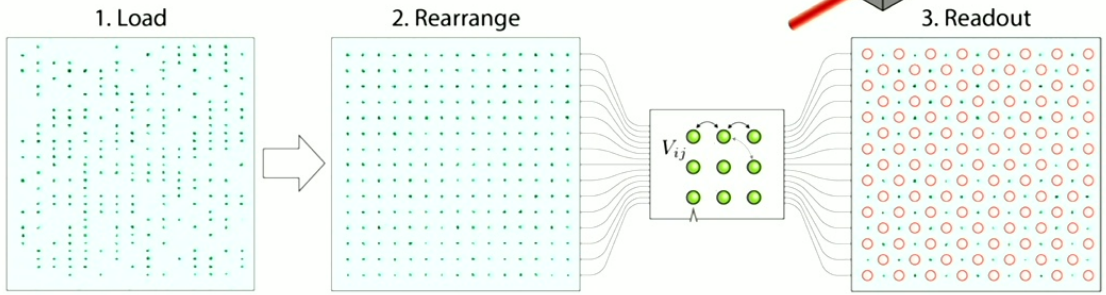

HARVARD UNIVERSITY

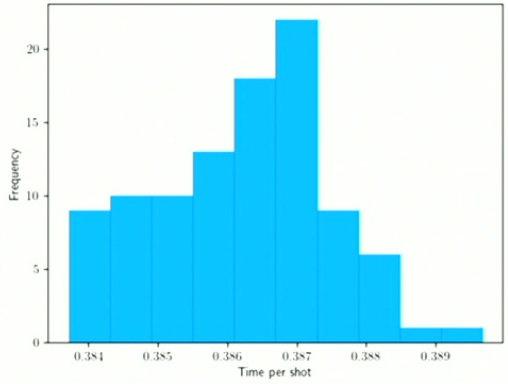
|QuEra⟩



$^{87}\mathrm{Rb}$




Single-atom resolved fluorescent imaging provides projective measurements of $|g\rangle$

Measurements are destructive

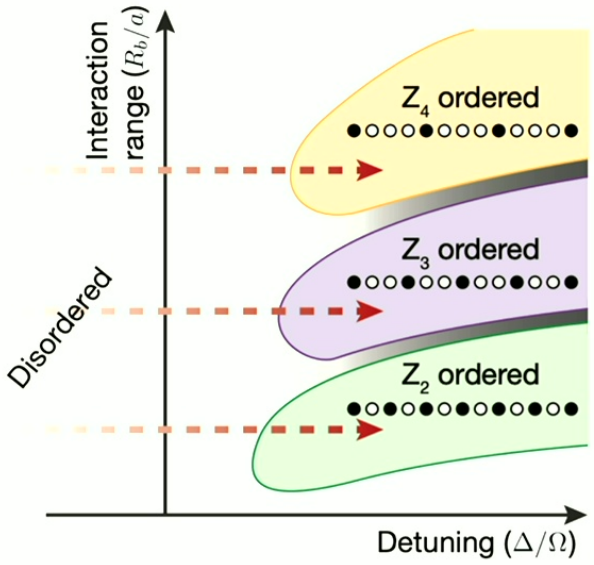
# 1D atom array

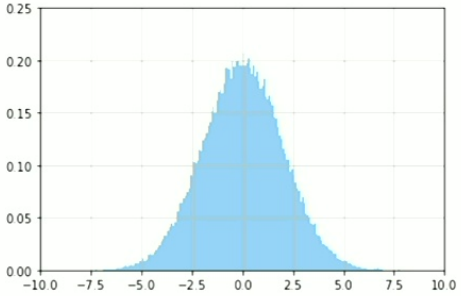
nature

**Probing many-body dynamics on a 51-atom quantum simulator**

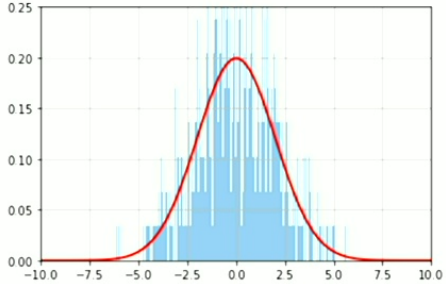
Hannes Bernien, Sylvain Schwartz, Alexander Keesling, Harry Levine, Ahmed Omran, Hannes Pichler, Soonwon Choi, Alexander S. Zibrov, Manuel Endres, Markus Greiner, Vladan Vuletić & Mikhail D. Lukin

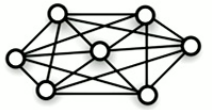
*Nature* **551**, 579–584 (2017)



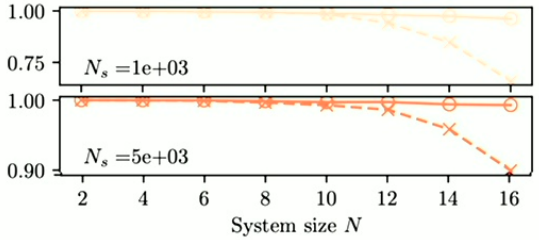



$$p(x) \approx \frac{1}{||\mathcal{D}||} \sum_{\mathbf{x}_k \in \mathcal{D}} \delta_{\mathbf{x},\mathbf{x}_k}$$



$p_\lambda(x) \approx$ 

Fidelity improvements for RBMs vs. frequency-distribution reconstructions, for a selection of dataset sizes Ns.



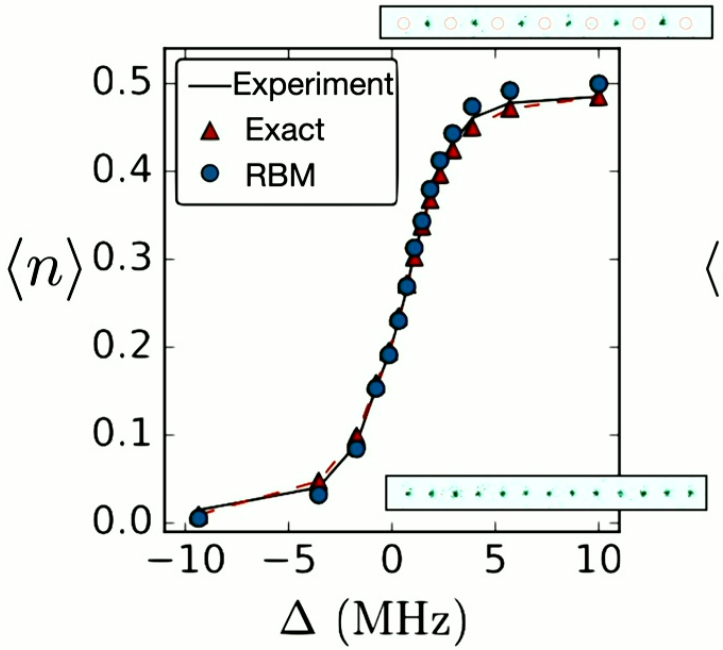


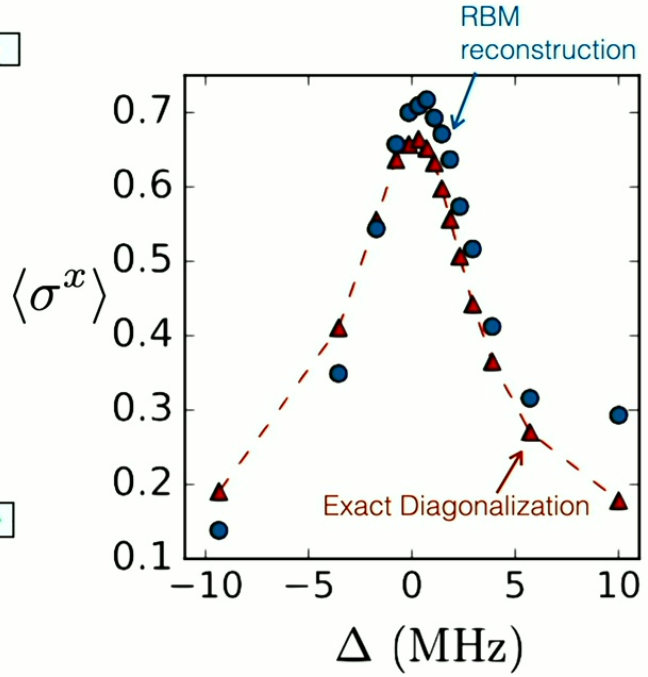
G. Torlai B. Timar

# Data-driven state reconstruction

Torlai, Timar, van Nieuwenburg, Levine, Omran, Keesling, Bernien, Greiner, Vuletić, Lukin, RGM, Endres, Phys. Rev. Lett. 123, 230504 (2019)

- 8 atoms, 3000 shots
- one-dimensional AFM chain
- 3000 projective measurements per detuning parameter
- RBM trained a energy-based model, produced estimators




Stoquastic Hamiltonian:

$$\psi_\lambda(z) = \sqrt{p_\lambda(z)}$$

$$\langle \mathcal{O} \rangle = \sum_{zz'} \psi_\lambda(z)\psi_\lambda(z')\mathcal{O}_{zz'}$$

$$= \sum_z \psi_\lambda^2(z) \sum_{z'} \frac{\psi_\lambda(z')}{\psi_\lambda(z)} \mathcal{O}_{zz'}$$

"local" estimator

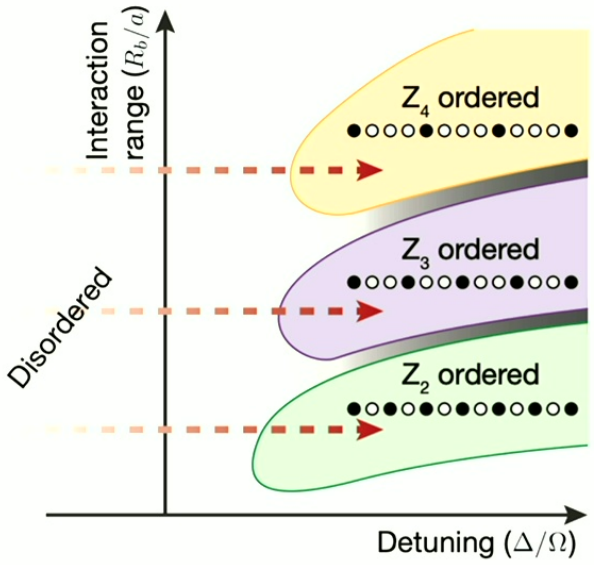
# 1D atom array

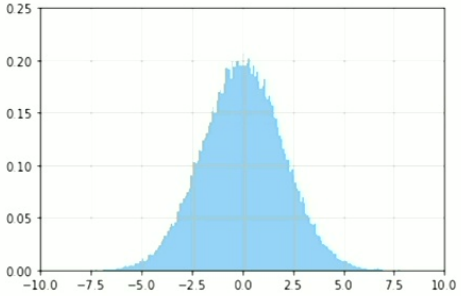
nature

**Probing many-body dynamics on a 51-atom quantum simulator**

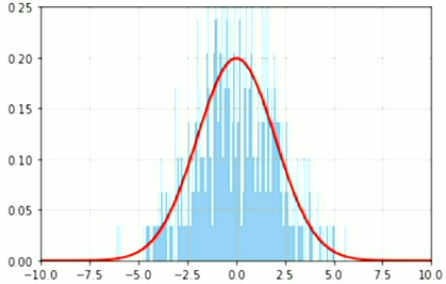
Hannes Bernien, Sylvain Schwartz, Alexander Keesling, Harry Levine, Ahmed Omran, Hannes Pichler, Soonwon Choi, Alexander S. Zibrov, Manuel Endres, Markus Greiner ✉, Vladan Vuletić ✉ & Mikhail D. Lukin ✉

*Nature* **551**, 579–584 (2017)

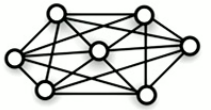


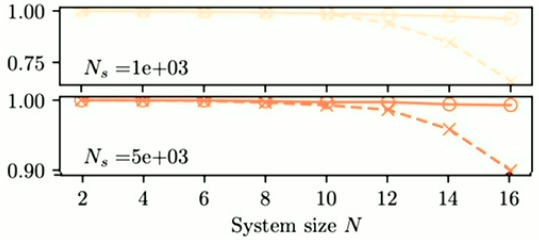
$$p(x) \approx \frac{1}{||\mathcal{D}||} \sum_{\mathbf{x}_k \in \mathcal{D}} \delta_{\mathbf{x},\mathbf{x}_k}$$



$$p_\lambda(x) \approx$$



Fidelity improvements for RBMs vs. frequency-distribution reconstructions, for a selection of dataset sizes Ns.



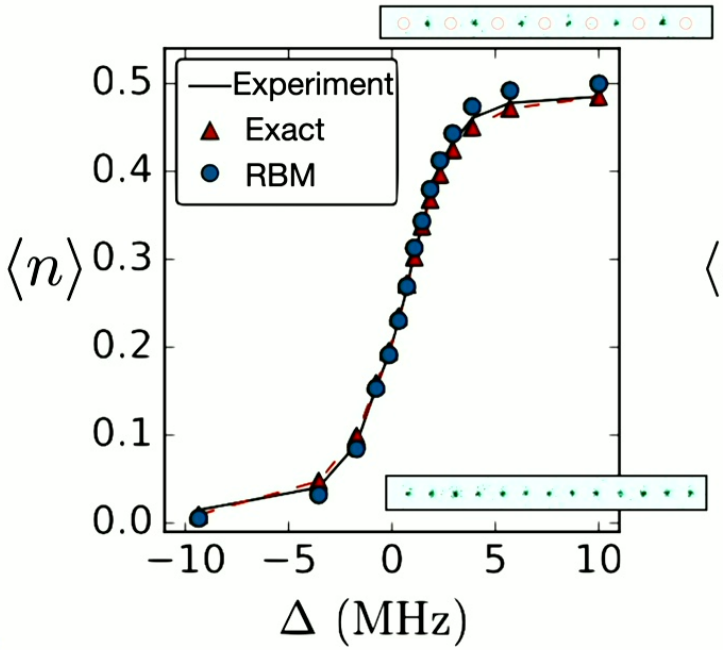

G. Torlai

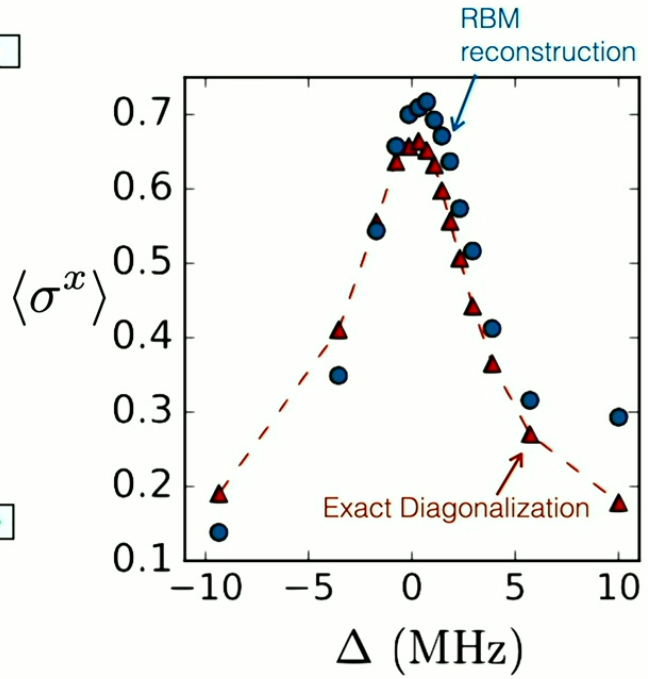
B. Timar

# Data-driven state reconstruction

Torlai, Timar, van Nieuwenburg, Levine, Omran, Keesling, Bernien, Greiner, Vuletić, Lukin, RGM, Endres, Phys. Rev. Lett. 123, 230504 (2019)

- 8 atoms, 3000 shots
- one-dimensional AFM chain
- 3000 projective measurements per detuning parameter
- RBM trained a energy-based model, produced estimators




Stoquastic Hamiltonian:

$$\psi_\lambda(z) = \sqrt{p_\lambda(z)}$$

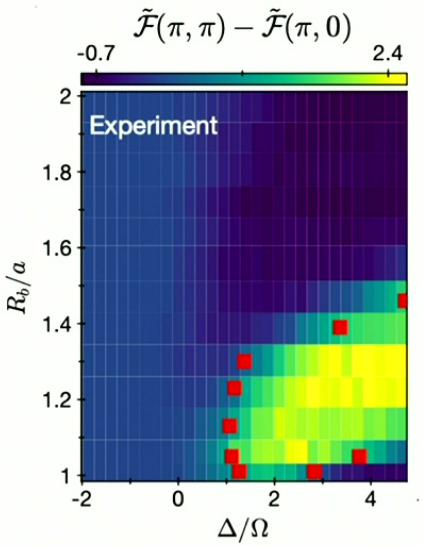
$$\langle \mathcal{O} \rangle = \sum_{zz'} \psi_\lambda(z)\psi_\lambda(z')\mathcal{O}_{zz'}$$

$$= \sum_z \psi_\lambda^2(z) \sum_{z'} \frac{\psi_\lambda(z')}{\psi_\lambda(z)} \mathcal{O}_{zz'}$$

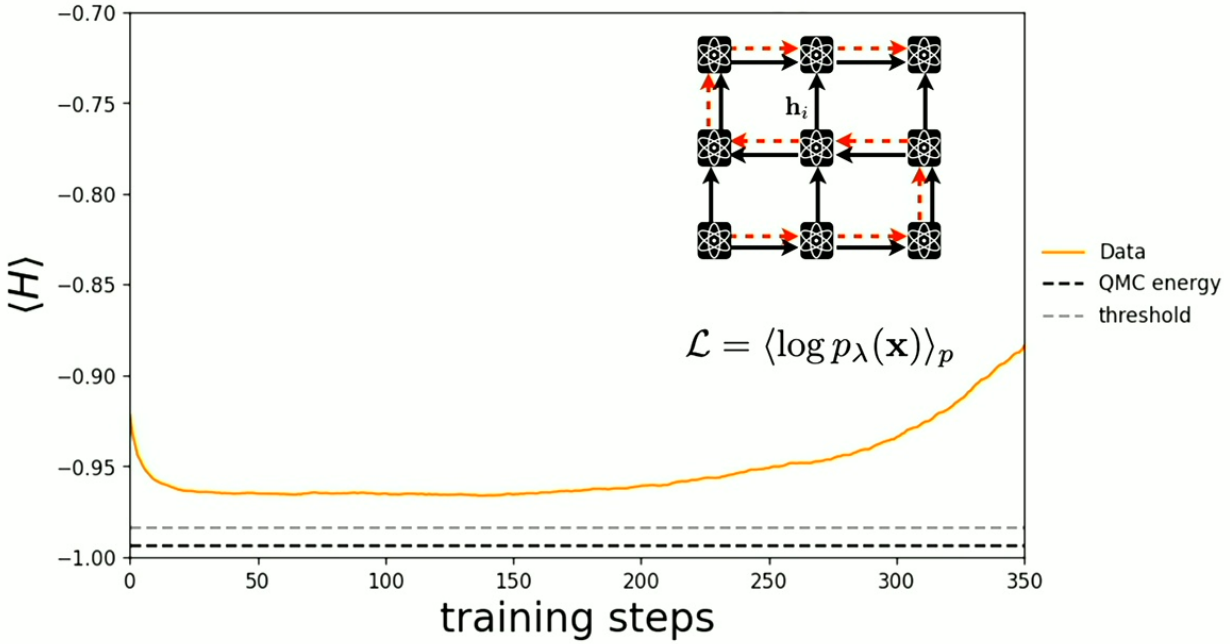
"local" estimator

# 2D square lattice atom array

- 16x16 lattice, near the disordered-checkerboard transition
- 1000 projective measurements per detuning parameter
- Used to train a 2D RNN wavefunction



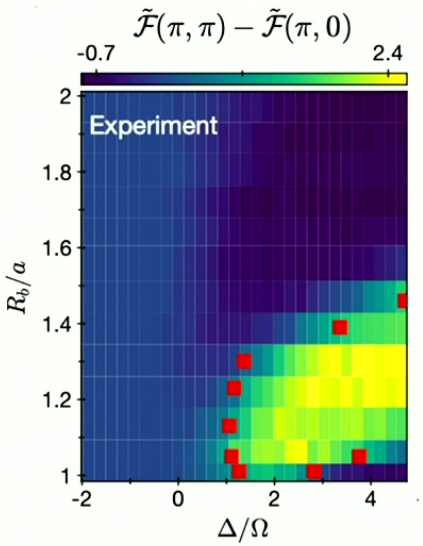

Ebadi et. al. arXiv:2012.12281



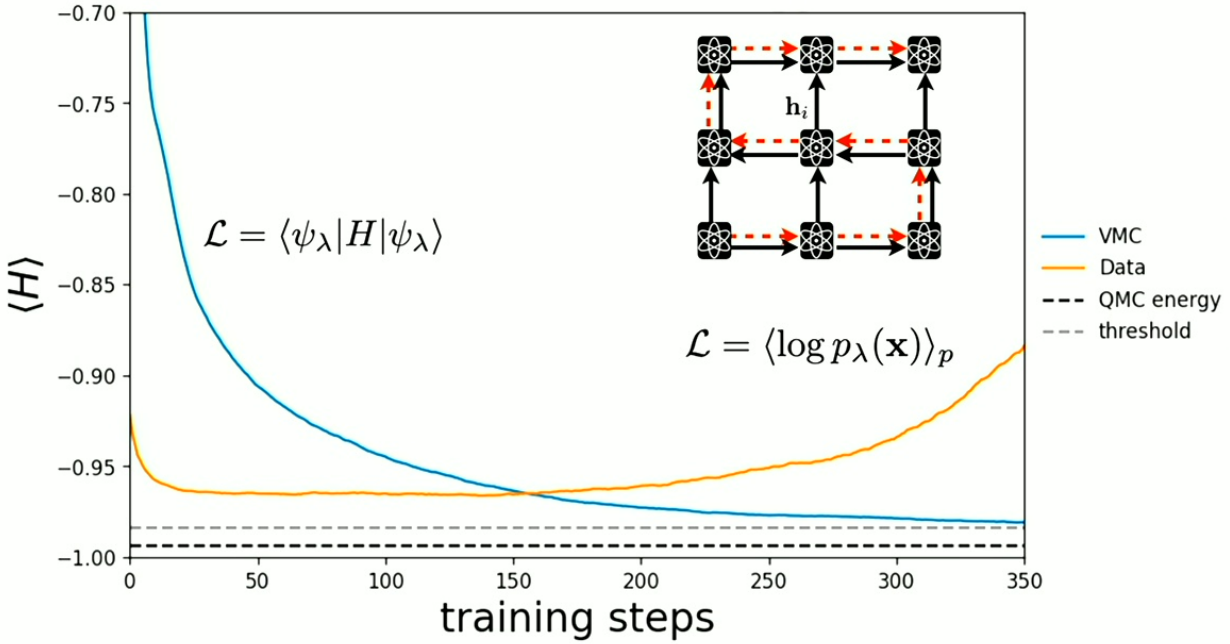

Moss, Ebadi, Wang, Semeghini, Bohrdt, Lukin, RGM, arXiv:2308.02647

# 2D square lattice atom array

- 16x16 lattice, near the disordered-checkerboard transition
- 1000 projective measurements per detuning parameter
- Used to train a 2D RNN wavefunction



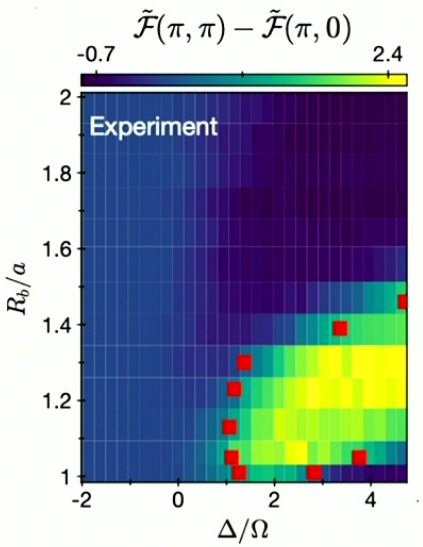

Ebadi et. al. arXiv:2012.12281



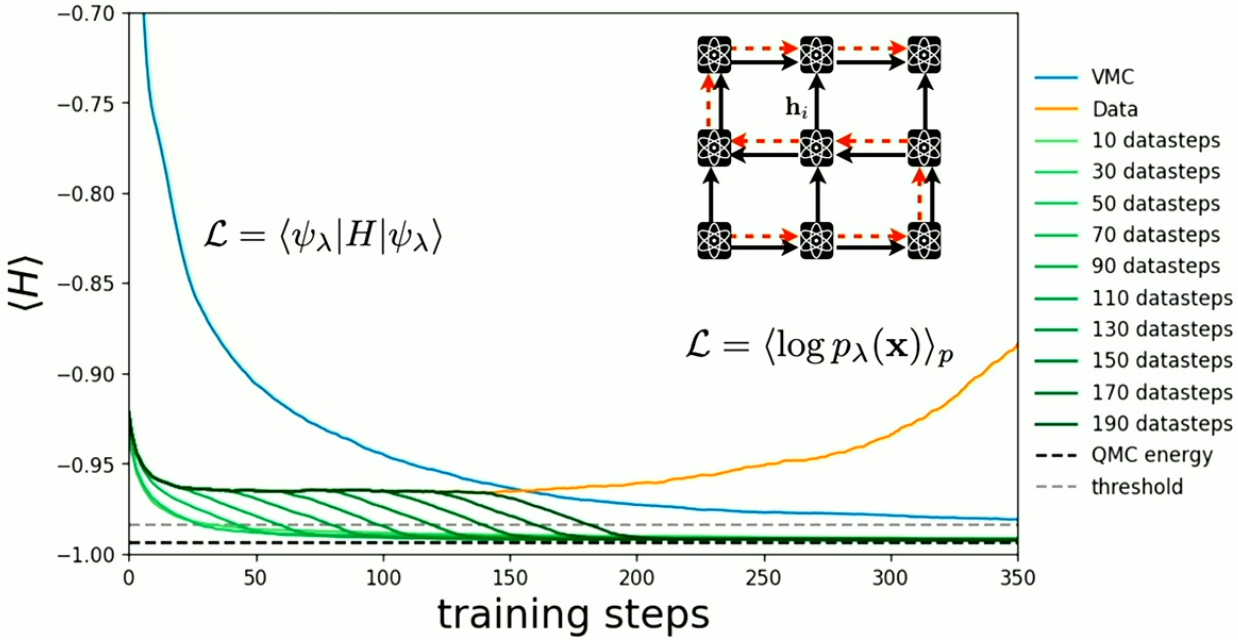

Moss, Ebadi, Wang, Semeghini, Bohrdt, Lukin, RGM, arXiv:2308.02647

# 2D square lattice atom array

- 16x16 lattice, near the disordered-checkerboard transition
- 1000 projective measurements per detuning parameter
- Used to train a 2D RNN wavefunction




Ebadi et. al. arXiv:2012.12281

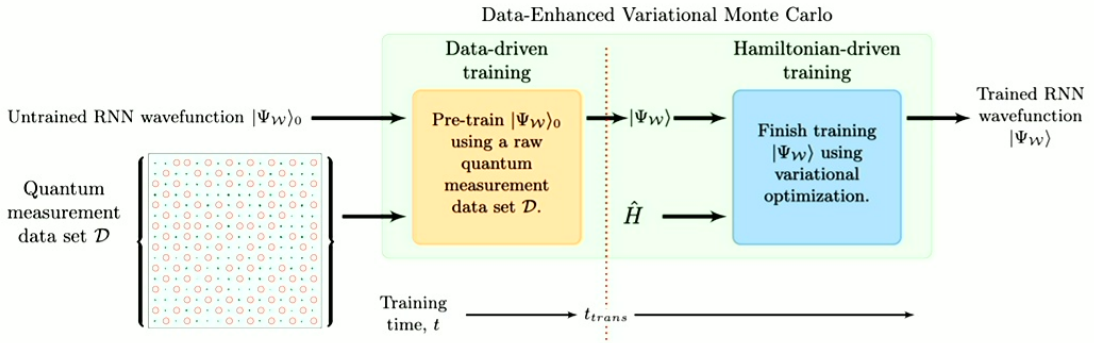


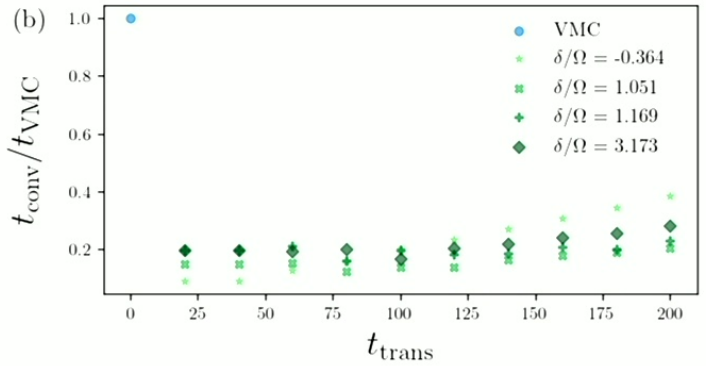

Moss, Ebadi, Wang, Semeghini, Bohrdt, Lukin, RGM, arXiv:2308.02647

# Time to solution improvements

Moss, Ebadi, Wang, Semeghini, Bohrdt, Lukin, RGM, arXiv:2308.02647

- Early-phase training moves the gradient descent algorithm into a parameter subspace that can more easily be optimized by later-phase variational training.



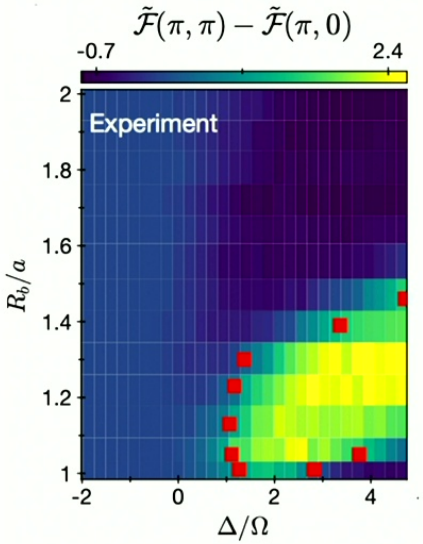




- A strategy for hybrid simulation of non-stoquastic Hamiltonians

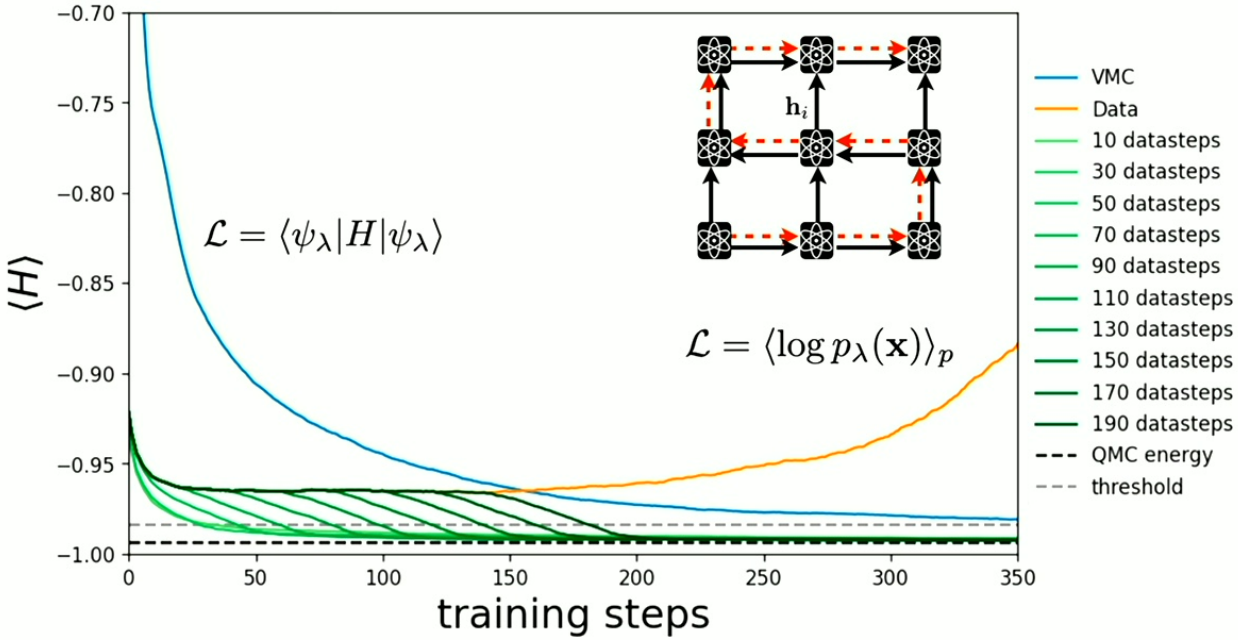
$$H = \sum_{i \neq j} J_{ij}\sigma_i^z\sigma_j^z + J'_{ij}(\sigma_i^x\sigma_j^x + \sigma_i^y\sigma_j^y)$$

# 2D square lattice atom array

- 16x16 lattice, near the disordered-checkerboard transition
- 1000 projective measurements per detuning parameter
- Used to train a 2D RNN wavefunction

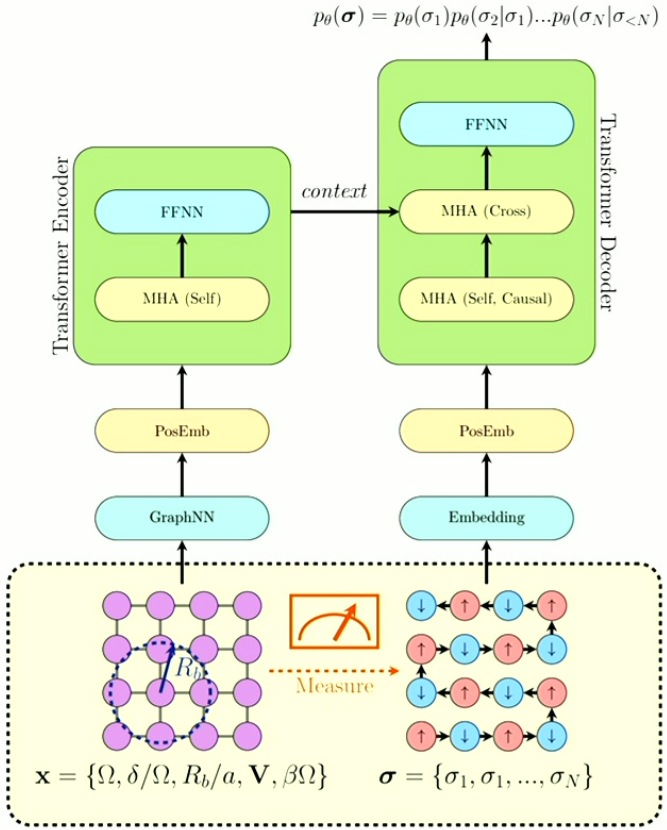



Ebadi et. al. arXiv:2012.12281




Moss, Ebadi, Wang, Semeghini, Bohrdt, Lukin, RGM, arXiv:2308.02647

# RydbergGPT




$$H = \Omega \sum_i \sigma_i^x - \Delta \sum_i n_i + \sum_{i<j} V_{ij} n_i n_j$$

arXiv > quant-ph > arXiv:2405.21052

Quantum Physics

[Submitted on 31 May 2024]

**RydbergGPT**

David Fitzek, Yi Hong Teoh, Hin Pok Fung, Gebremedhin A. Dagnew, Ejaaz Merali, M. Schuyler Moss, Benjamin MacLellan, Roger G. Melko

| Parameter | Value |
|---|---|
| Neural network architecture | |
| $d_{\text{ff}}$ | 128 |
| $d_{\text{model}}$ | 32 |
| $d_{\text{graph}}$ | 64 |
| num heads | 8 |
| num blocks encoder | 1 |
| num blocks decoder | 3 |
| num graph layers | 2 |
| trainable params | 66562 |
| Training hyperparameters | |
| batch size | 1024 |
| optimizer | AdamW |
| dropout | 0.1 |
| learning rate | 0.001 |
| learning rate schedule | Cosine annealing warm start |
| $T_0$ | 1 |
| $T_{\text{mult}}$ | 2 |
| $\eta_{\text{min}}$ | 0.00001 |
| dataset buffer | 50 |

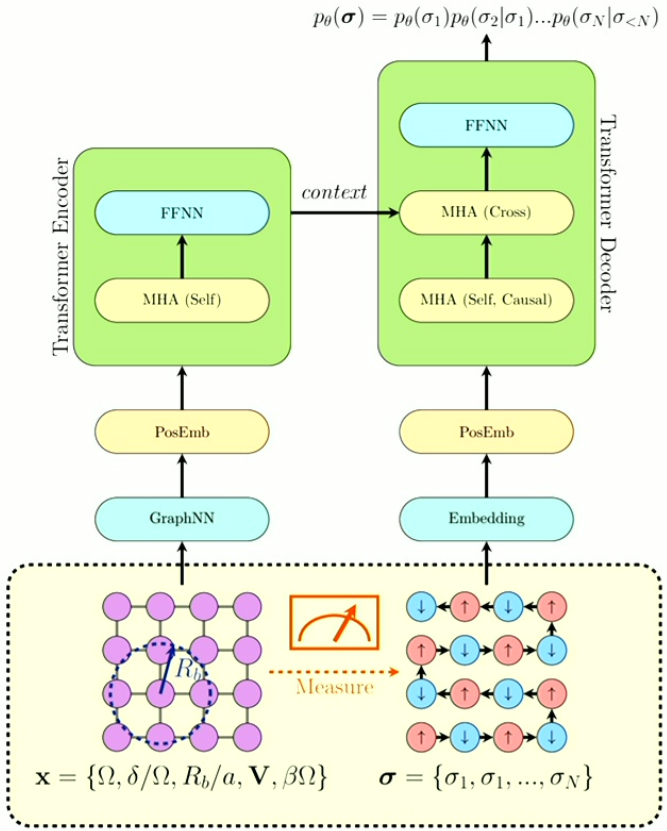
PIQuIL / RydbergGPT

RydbergGPT

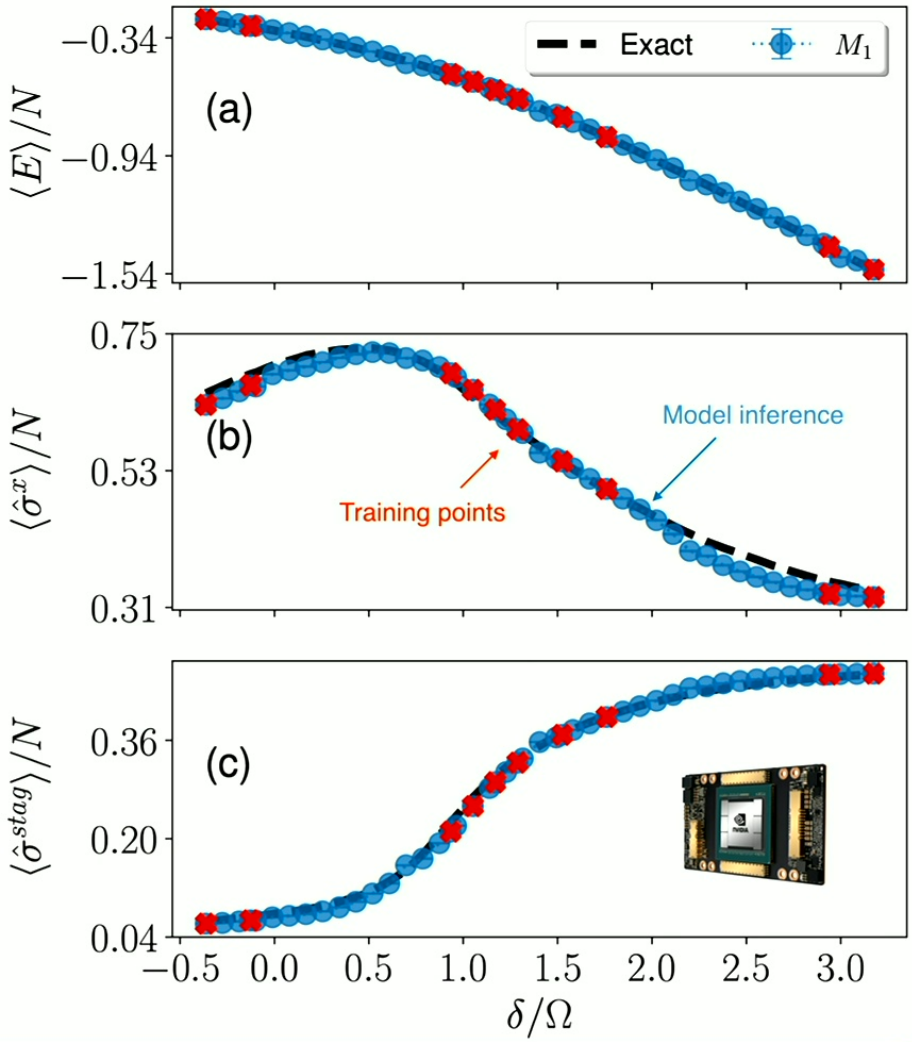
PENNYLANE

https://pennylane.ai/datasets/other/rydberggpt

# RydbergGPT

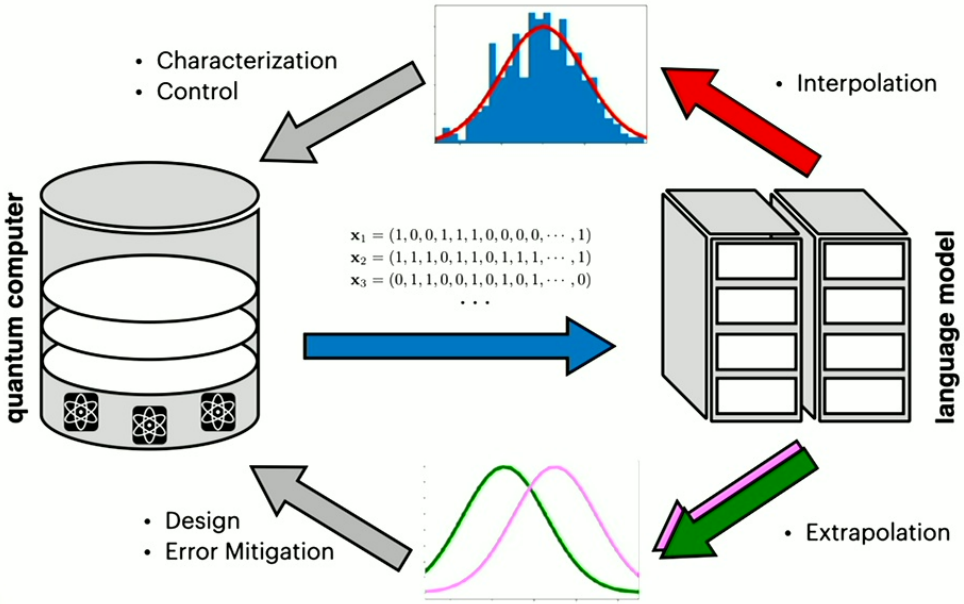



$$H = \Omega \sum_i \sigma_i^x - \Delta \sum_i n_i + \sum_{i<j} V_{ij} n_i n_j$$



# Discussion: language models for quantum simulation

- LLMs are increasingly demonstrating *scale* & *emergence*
- Quantum measurement data is well suited to train generative models
- Large models will be a flexible tool to aid quantum computer design & control
- It would be interesting to *scale* LLMs using quantum data… but can we afford to?

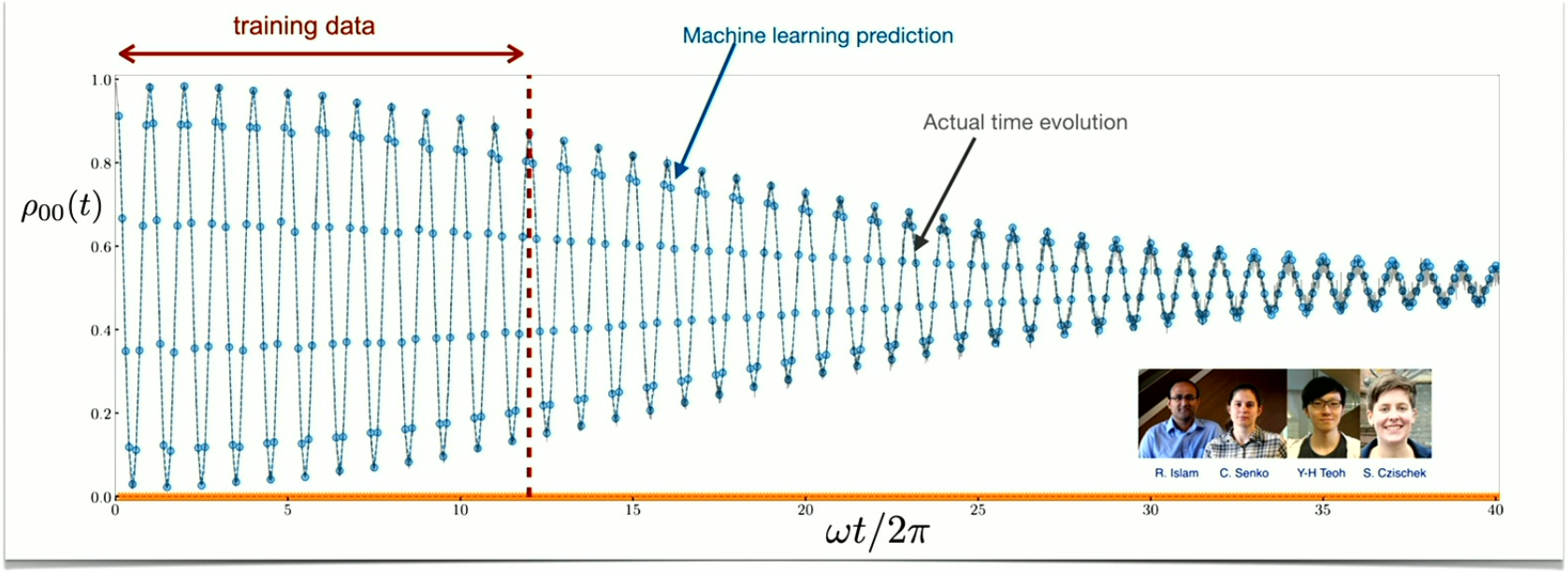



Open Quantum Design

https://openquantumdesign.org

# Many more applications of LMs to quantum

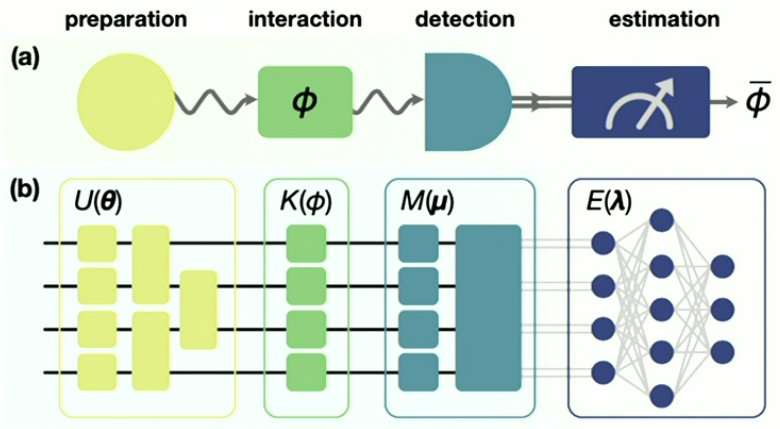
- Learning Hamiltonian parameters from data
- **Evaluating state preparation protocols**
- 
- 
- 



# Many more applications of LMs to quantum

B. MacLellan

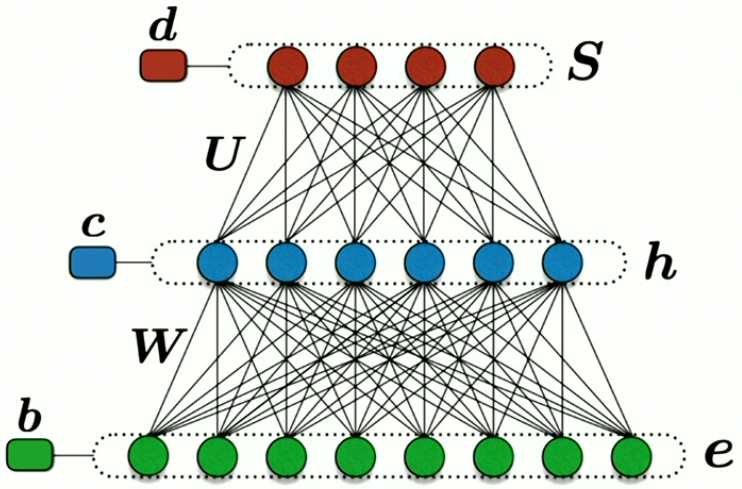
- Learning Hamiltonian parameters from data
- Evaluating state preparation protocols
- Variational Monte Carlo
- **Variational quantum sensing**
- Quantum error correction



# Many more applications of LMs to quantum

- Learning Hamiltonian parameters from data
- Evaluating state preparation protocols
- Variational Monte Carlo
- Variational quantum sensing
- **Quantum error correction**

G. Torlai

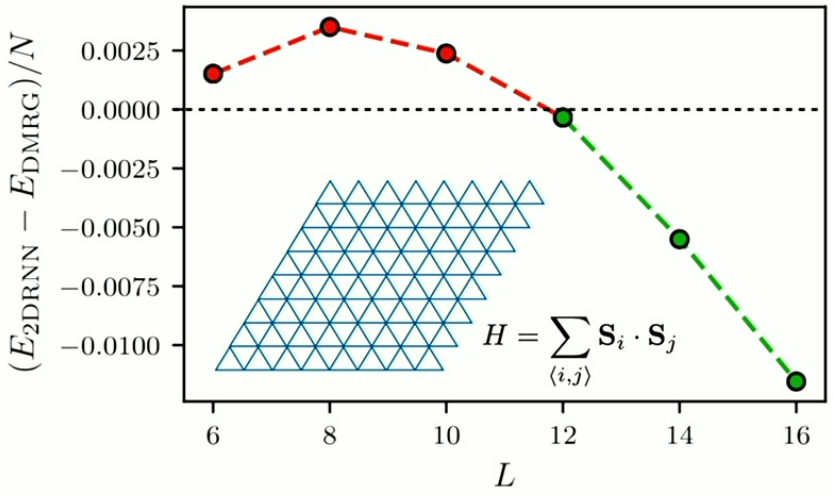



G. Torlai, RGM, Phys. Rev. Lett. 119, 030501 (2017)

# Many more applications of LMs to quantum

M. Hibat Allah

- Learning Hamiltonian parameters from data
- Evaluating state preparation protocols
- **Variational Monte Carlo**
- Variational quantum sensing
- Quantum error correction



# E.g. quantum error correction

Wang, Liu, Shao, **Dantong Li,** Gu, Pan, Ding, Han, arXiv:2311.16082

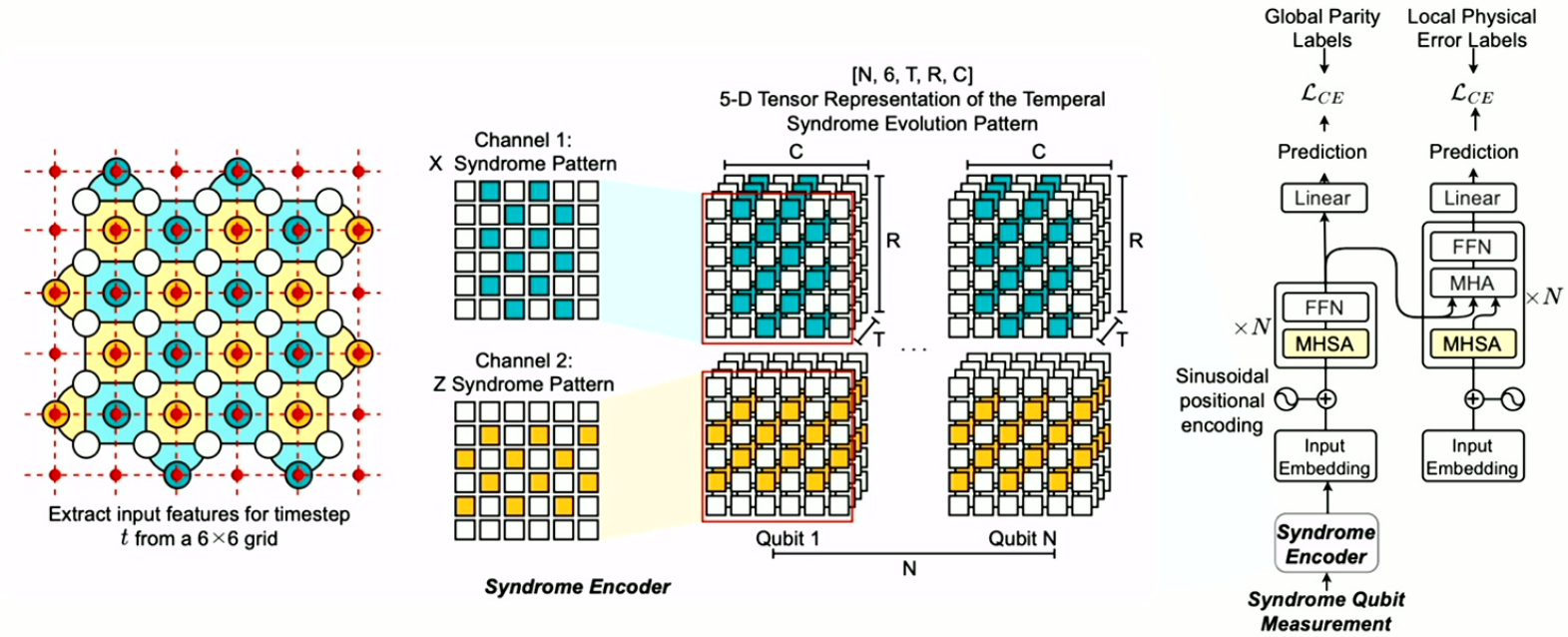
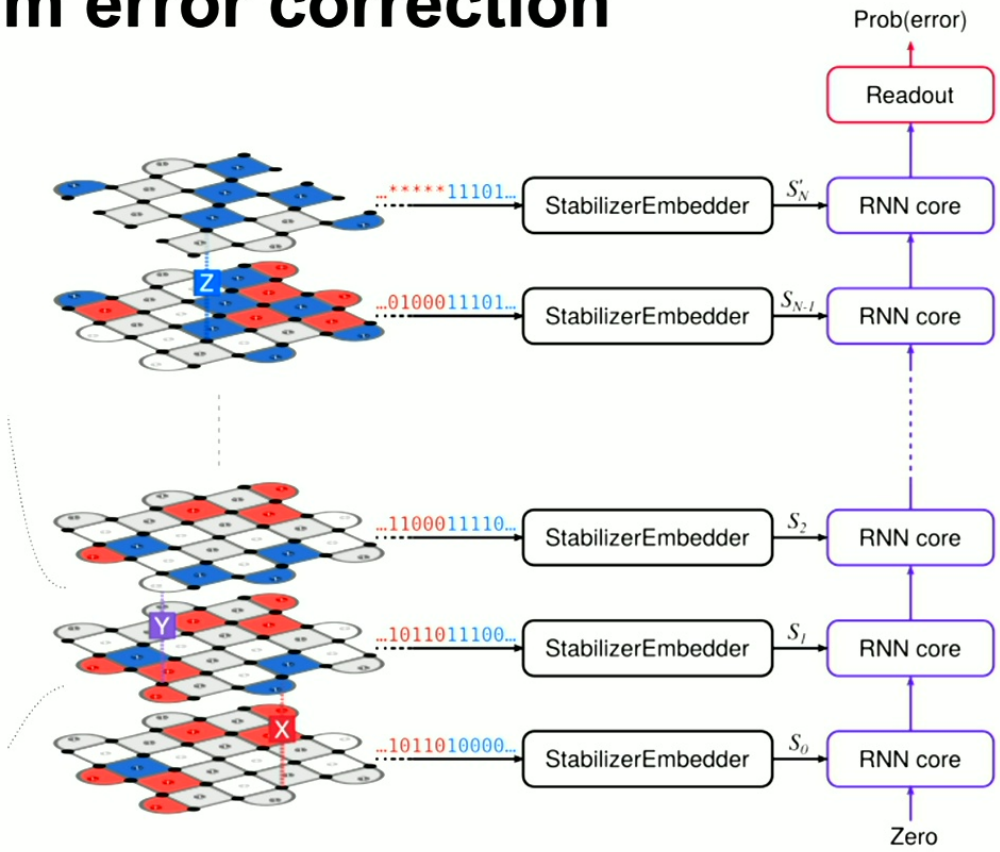



**Figure 6: Transformer model architecture. The input of the syndromes will be encoded by a $(D+1)$ cubic grid. The input will go through the transformer encoder with self attention and FFN layers. Then the transformer decoder will produce the physical error predictions by processing the positional encoding of data qubits with size $D$ cubic.**

# E.g. quantum error correction




Learning to Decode the Surface Code with a Recurrent, Transformer-Based Neural Network
Google DeepMind & Google Quantum AI, arXiv:2310.05900