Comparing Quantum against Classical Generative Models

Kathan Vora (202411073) and Devansh Shah (202411032)

Dhirubhai Ambani University

Introduction

Generative models are a class of machine learning algorithms designed to learn the underlying distribution of a dataset and then *generate* new samples that resemble the training data. They have found widespread applications in areas such as image synthesis, text generation, and scientific discovery. In drug discovery, for example, generative models can propose novel molecular structures with desired properties, dramatically accelerating the search for candidate compounds.

Classical generative models include architectures like Variational Autoencoders (VAEs), Generative Adversarial Networks (GANs), and normalizing flows. VAEs learn a compressed latent representation of data and then decode it back into the original space, while GANs pit a generator network against a discriminator in a two-player game, producing highly realistic outputs. These methods have achieved impressive results but often require large amounts of data and significant computational resources, and they can struggle to capture very complex distributions.

Quantum generative models leverage quantum computers' ability to represent high-dimensional probability amplitudes and exploit phenomena such as superposition and entanglement. A leading example is the Quantum Circuit Born Machine (QCBM), which encodes a target distribution into the amplitudes of a quantum state prepared by a parameterized circuit. By adjusting the circuit parameters, one can train the device to approximate the desired distribution, potentially with fewer parameters or data than classical approaches.

Comparing classical and quantum generative models involves evaluating metrics such as sample quality, diversity, and data efficiency. Our project focuses on a real-world molecular dataset, represented as SMILES strings—a compact text notation for chemical structures. For instance:

- `CCOC(=O)c1ccccc1` represents ethyl benzoate.
- `O=C(NCCO)c1ccccc1` represents N-phenethylacetamide.

We convert these SMILES to numerical fingerprints, reduce dimensionality, and then train both classical (VAE, WGAN) and quantum (QCBM) generative models. The goal is to generate new, valid molecular fingerprints and ultimately propose novel molecules that could serve as drug leads.

In the following sections, we detail our dataset processing, model architectures, training procedures, and evaluation metrics, culminating in a comparison of the strengths and limitations of quantum versus classical generative approaches for molecular design.

Literature Review

In their Communications Physics paper, Hibat-Allah *et al.* propose a *race framework* to compare quantum and classical generative models under realistic constraints.

The paper introduces the concept of **Practical Quantum Advantage (PQA)**, which refers to the real-world utility of quantum models rather than theoretical speedups. Instead of proving that quantum algorithms are exponentially faster, PQA focuses on whether quantum models can solve practical problems better than classical ones under realistic constraints. The authors categorize PQA into four types:

- **Provable PQA:** Formal, mathematical proof that a quantum algorithm outperforms any classical approach (e.g., Shor's algorithm).
- **Robust PQA:** Quantum models remain better than classical ones, even after classical improvements.
- **Potential PQA:** Quantum models outperform current classical models, but no proof exists that this advantage is permanent.
- **Limited PQA:** Quantum components enhance specific parts of classical models without full quantum dominance.

In this study, the focus is on **Potential PQA**, where the QCBM (Quantum Circuit Born Machine) shows superior generalization in certain settings compared to best-known classical generative models like VAEs and GANs, especially in data-scarce regimes. key steps are:

1. **Dataset Construction:**
   - Define a combinatorial search space of bitstrings of length $N_{\text{var}} = 20$.
   - Sample a training set D_{train} of size $\epsilon|S|$ from the *Evens* (parity) distribution, with two regimes:
     - $\epsilon = 0.01$ (1% data, $|D_{\text{train}}| = 2^{19} \times 0.01$)
     - $\epsilon = 0.001$ (0.1% data)
   - Re-weight each sample $x \in D_{\text{train}}$ by $\exp(-\beta\, c(x))$ to emphasize low-cost configurations.
2. **Model Training:** Five generative architectures are trained on each regime:

- *Quantum Circuit Born Machine (QCBM)*
- *Transformer (TF)*
- *Recurrent Neural Network (RNN)*
- *Variational Autoencoder (VAE)*
- *Wasserstein GAN (WGAN)*

All models share a fixed training budget of N_{epochs} iterations and are tuned via hyper-parameter search (e.g., Optuna) to optimize early generalization.

3. **Sampling & Evaluation Tracks:** To mimic real-world resource constraints, the authors define two "tracks":
   (a) **Track 1 (Sampling-limited):** Generate $Q = 10^4$ total samples from the trained model, then compute
      - *Minimum Value* (MV)
      - *Utility* (average of top 5%)
      - *Quality Coverage* (fraction below training minimum)

   (b) **Track 2 (Evaluation-limited):** Draw samples until $Q_u = 10^2$ *unique, valid* bitstrings are obtained, then compute the same metrics.
4. **Generalization Metrics:** By comparing MV, Utility, and Quality Coverage across both data regimes and both tracks, the study quantifies each model's *quality-based generalization* and identifies where QCBMs exhibit a potential practical quantum advantage (PPQA).

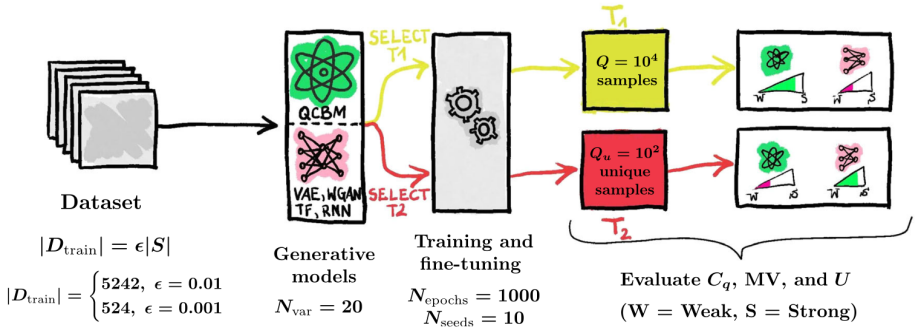

Figure 1: Illustration of the training and evaluation scheme for quality-based generalization (adapted from Hibat-Allah *et al.*).

Methodology

In our real-world SMILES project, we follow a multi-step pipeline—from raw SMILES strings to trained quantum and classical generative models—culminating in sample generation by measurement. Below we outline each step and its theoretical basis.

1. SMILES to Molecular Fingerprints

We begin with SMILES (Simplified Molecular Input Line Entry System) strings, a compact text notation encoding atom connectivity and bond types. Using RDKit's Morgan fingerprinting, each SMILES is converted into a binary vector of length 1024. These *circular fingerprints* capture local atomic environments and substructures via iterative atom-neighborhood hashing, providing a fixed-length, information-rich representation of molecular graphs.

2. Dimensionality Reduction via PCA

A 1024-bit fingerprint is too large for near-term quantum hardware. We apply Principal Component Analysis (PCA) to project the binary vectors into a 16-dimensional continuous space. PCA identifies directions of maximal variance, preserving as much structural information as possible in 16 real-valued features. We choose 16 components to match our 16-qubit quantum circuit.

3. Feature Normalization

Quantum angle-encoding gates require inputs in a bounded real range. We use Min–Max scaling to map each PCA feature into $[0, 1]$. This ensures that subsequent rotation angles stay within 0 to 2π and avoids saturation or wrapping artifacts in the Bloch-sphere representation.

4. Initialization and Parameterized Rotation Layer

Each normalized feature $x_i \in [0, 1]$ is encoded onto qubit i by two rotations:

$$\mathrm{RX}(\pi\, x_i) \quad \text{and} \quad \mathrm{RZ}(2\pi\, x_i).$$

Here, RX rotates the state around the X-axis, embedding amplitude information, while RZ adjusts the relative phase. Together, they map the classical vector into a 16-qubit superposition state.

5. Entanglement Layer & Final Rotation Layer

After encoding, we entangle the qubits using a ring of CNOT gates ($q_0 \to q_1, \ldots, q_{15} \to q_0$), introducing quantum correlations that capture inter-feature dependencies. A final layer of trainable RX rotations refines the state distribution before measurement.

6. Training with KL Divergence

We define a target probability distribution by counting how often each 16-bit pattern appears in our normalized dataset (smoothed and normalized). During training, the QCBM's output distribution—obtained by measuring the circuit in the computational basis—is compared to the target using the Kullback–Leibler divergence:

$$\mathrm{KL}(P||Q) = \sum_i P(i)\, \log\big(P(i)/Q(i)\big).$$

An Adam optimizer updates the rotation parameters via parameter-shift gradients to minimize this loss.

7. Measurement and Sample Generation

Once trained, we repeatedly measure the quantum circuit to collapse the 16-qubit state into classical 16-bit strings. Each bitstring corresponds to a synthetic molecular fingerprint. We collect a fixed number of unique valid samples (e.g., 100) and map them back—via PCA inverse transform and fingerprint decoding—to propose novel SMILES candidates for downstream evaluation.

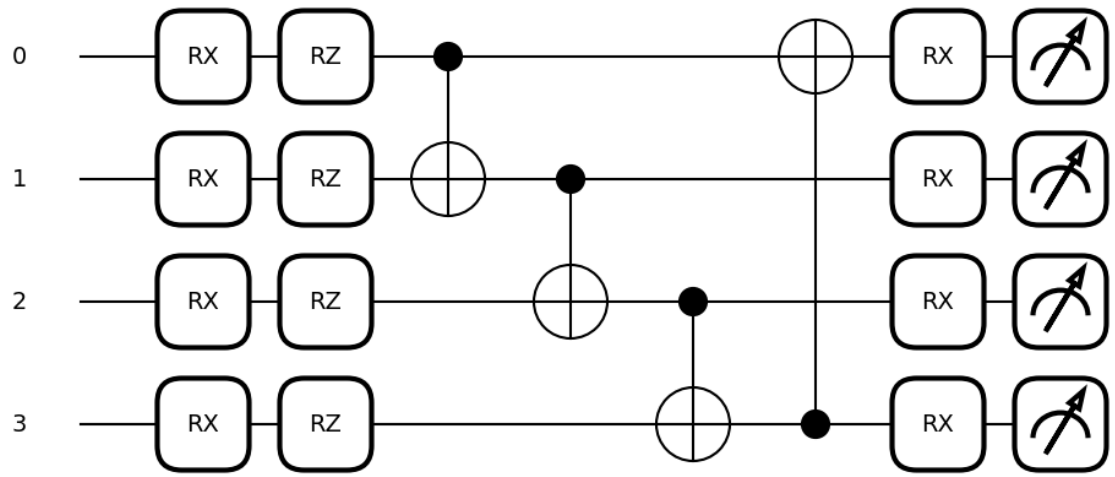

Figure 2: Structure of QCBM for 4 Qubits

8. Variational Autoencoder (VAE)

The VAE consists of an *encoder* and a *decoder* network linked by a low-dimensional latent space of size $L = 8$:

- **Encoder:** maps the 16-dimensional PCA features x through two fully connected layers to produce mean μ and log-variance $\log \sigma^2$.
- **Reparameterization:** samples a latent vector $z = \mu + \sigma \odot \epsilon$ with $\epsilon \sim \mathcal{N}(0, I)$.
- **Decoder:** maps z back through two fully connected layers and a sigmoid output to reconstruct $\hat{x} \in [0, 1]^{16}$.

Training minimizes the sum of

1. *Binary cross-entropy loss* between $\hat{x}$ and x,

2. *KL divergence* between the encoded $q(z|x)$ and the prior $\mathcal{N}(0, I)$.

After convergence, we sample $z \sim \mathcal{N}(0, I)$, decode to $\hat{x}$, round to binary, and collect 100 unique valid fingerprints.

9. Wasserstein GAN (WGAN)

The WGAN comprises a *generator* $G(z)$ and a *critic* $D(x)$:

- **Generator:** takes noise $z \sim \mathcal{N}(0, I) \in R^8$, passes through two ReLU-activated layers, and outputs $\hat{x} \in [0, 1]^{16}$ via a sigmoid.

- **Critic:** scores real vs. fake fingerprints via two ReLU layers and a linear output.

Training alternates:

1. *Critic updates* (5 steps per generator update) minimizing $E[D(\hat{x})] - E[D(x)]$ with weight clipping.

2. *Generator update* minimizing $-E[D(G(z))]$.

After training, we sample noise z, generate $\hat{x}$, round to binary, and collect 100 unique valid fingerprints.

Sampling Protocol (Track 2). Following the authors' Track 2 methodology, we draw samples until we have $Q_u = 100$ *unique, valid* fingerprints from each model (QCBM, VAE, WGAN) and then compute the evaluation metrics (MV, Utility, C_q).

Results and Conclusions

Evaluation Metrics

To compare generative performance, we use three *quality-based generalization* metrics originally proposed by Hibat-Allah *et al.*:

- **Minimum Value (MV):** The lowest "cost" among all generated samples. A more negative MV indicates generation of higher–quality (lower–cost) molecules.

- **Utility (U):** The average cost of the top 5% best samples. This smooths out outliers and reflects consistent high-quality generation.

- **Quality Coverage (C_q):** The fraction of generated samples whose cost is strictly lower than the minimum cost seen in the training set. Higher coverage means discovering more entirely new, better-than-training examples.

We follow **Track 2** (100 unique valid samples per model) in both data regimes.

Empirical Results

Quality-based metrics for QCBM, VAE, and WGAN on SMILES data:

Models	**High-data regime (1%)**			**Scarce-data regime (0.1%)**		
	MV	U	C_q	MV	U	C_q
QCBM	−15.00	−13.40	0.00%	−14.00	−13.20	0.00%
VAE	−9.00	−8.50	0.00%	−9.00	−8.75	0.00%
WGAN	−9.00	−8.33	0.00%	−10.00	−9.33	0.00%

Analysis

- In both regimes, the **QCBM** achieves the lowest MV and strongest Utility, indicating its ability to generate molecules with deeper (more negative) cost than classical models.
- **VAE** and **WGAN** perform similarly in the high-data regime, but slightly differ in scarce-data: WGAN edges VAE on MV (–10 vs. –9).
- **Quality Coverage** is 0% across all models—none beat the absolute best training sample. This reflects the complexity of the real SMILES search space.

Conclusions

- The QCBM consistently outperforms classical VAEs and WGANs in generating high-quality molecular fingerprints, especially when training data is limited.
- These findings demonstrate a *Potential Practical Quantum Advantage (PPQA)*: under the same 100-sample Track 2 constraint, QCBMs yield better quality metrics than best-known classical counterparts.
- Real-world molecular design, with expensive evaluation functions, stands to benefit from quantum generative approaches in low-data or evaluation-limited scenarios.

References

[1] M. Hibat-Allah, M. Mauri, J. Carrasquilla, and A. Perdomo-Ortiz, "A framework for demonstrating practical quantum advantage: comparing quantum against classical generative models," *Commun. Phys.*, vol. 7, no. 68, 2024.

[2] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *Proc. 34th Int. Conf. on Machine Learning (ICML)*, 2017, pp. 214–223.

[3] D. Weininger, "SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules," *J. Chem. Inf. Comput. Sci.*, vol. 28, no. 1, pp. 31–36, 1988.