JCTC Journal of Chemical Theory and Computation

pubs.acs.org/JCTC Article

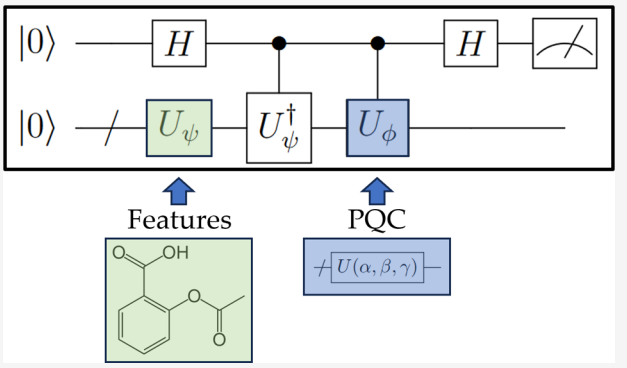
Quantum-to-Classical Neural Network Transfer Learning Applied to Drug Toxicity Prediction

Anthony M. Smaldone and Victor S. Batista*

Cite This: J. Chem. Theory Comput. 2024, 20, 4901−4908 Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information

ABSTRACT: Toxicity is a roadblock that prevents an inordinate number of drugs from being used in potentially life-saving applications. Deep learning provides a promising solution to finding ideal drug candidates; however, the vastness of chemical space coupled with the underlying $O(n^3)$ matrix multiplication means these efforts quickly become computationally demanding. To remedy this, we present a hybrid quantum-classical neural network for predicting drug toxicity utilizing a quantum circuit design that mimics classical neural behavior by explicitly calculating matrix products with complexity $O(n^2)$. Leveraging the Hadamard test for efficient inner product estimation rather than the conventionally used swap test, we reduce the number of qubits by half and remove the need for quantum phase estimation. Directly computing matrix products quantum mechanically allows for learnable weights to be transferred from a quantum to a classical device for further training. We apply our framework to the Tox21 data set and show that it achieves commensurate predictive accuracy to the model's fully classical $O(n^3)$ analogue. Additionally, we demonstrate that the model continues to learn, without disruption, once transferred to a fully classical architecture. We believe that combining the quantum advantage of reduced complexity and the classical advantage of noise-free calculation will pave the way for more scalable machine learning models.



INTRODUCTION

The quest for novel pharmaceuticals is fraught with the challenge of ensuring drug safety. With 90% of drug candidates failing in clinical trials and 30% of these failures being attributed to toxicity,[1] the cost to create a successful drug can exceed $2 billion USD,[2] making the early detection of toxicological properties in drug candidates crucial.

Machine learning (ML) has revolutionized the field of drug discovery,[3−5] offering a powerful tool to handle and interpret the complex data sets that are characteristic of pharmaceutical research. By leveraging algorithms that can learn from and make predictions on data, ML has enabled significant advancements in identifying new drug candidates[6−8] and predicting drug toxicity.[9−14] As ML continues to amaze researchers, they are always searching for new computational tools to bolster the tractability and performance of their models.

Quantum computing, with its inherent capability to handle certain complex computations more efficiently than classical computing, has emerged as a promising solution. Leveraging the principles of superposition and entanglement, quantum computing offers a new framework that potentially expands computational boundaries for ML. Recent work suggests that quantum ML (QML) models are more learnable and generalize better to unseen data than classical networks.[15−18] In pursuit of these potential advantages, researchers have built numerous QML models to address a range of chemical and biological problems.[19−28] More specifically, there have been several QML works that are trained to predict toxicity.[10,29,30] Despite the success of these QML models, the limitations that plague noisy intermediate-scale quantum (NISQ) devices, such as decoherence and gate errors, are still a sobering reality.

While noise is not a concern for classical computing, it is not without its bottleneck either. Every year, as developers create deep learning models more powerful than the last, we are seeing a swift increase in the number of parameters.[31] This escalation in parameter count introduces a consequential challenge: the need for significantly more computing power and the development of more efficient algorithms. Matrix multiplication lies at the heart of neural networks. The standard method of matrix multiplication has complexity $O(n^3)$, and while subcubic algorithms do exist such as the Strassen

Received: April 1, 2024
Revised: May 10, 2024
Accepted: May 13, 2024
Published: May 25, 2024

algorithm[32] $O(n^{2.8074})$ and the recording-setting algorithm by Williams et al.[33] $O(n^{2.3716})$, these methods suffer from numerical instability,[34] high memory usage, or have a complexity prefactor that makes them galactic algorithms, rendering them impractical for tractable computation.

In response to these computational challenges, this study introduces a hybrid quantum-classical neural network model designed to leverage the strengths of both quantum and classical computing paradigms and apply the framework to drug toxicity prediction. Utilizing the Hadamard test, our model employs quantum circuits that replicate the forward functionality of classical neural networks, allowing it to efficiently calculate discrete inner products. Assuming efficient state preparation, the swap test has been shown to be the most computationally favorable algorithm[35] for matrix multiplication with complexity $O(n^2)$ and has been suggested for use in a quantum neuron.[36−39] Our use of the Hadamard test achieves the same complexity, one quantum circuit for each dot product while using half the working qubits and without the need for quantum phase estimation. This QML approach allows for "classical weights" to be easily derived from the quantum model, which can then be transferred to a classical machine to be fine-tuned without the presence of noise and errors, which is characteristic of current NISQ devices. All previous works of quantum transfer learning are either classical-to-quantum[40−42] or simply use the initial quantum component as a feature extractor that is nonlearnable after transferring.[40] To the best of our knowledge, this framework of fine-tuning weights classically that were initially embedded in a quantum circuit is a novel contribution to this growing field of quantum transfer learning.

In an effort to realize these potential quantum advantages, we investigate our model's performance on drug toxicity prediction using the Tox21 data set[43,44] with a convolutional neural network (CNN). We present the effectiveness of a quantum CNN (QCNN), which replaces the filter with a quantum circuit to compute the dot product. Additionally, we illustrate that a classical convolution filter can seamlessly continue the training process initially started by its quantum counterpart. Furthermore, we demonstrate that both the hybrid quantum-classical model and fully classical equivalents exhibit comparable performance levels for drug toxicity prediction.

QUADRATIC MATRIX MULTIPLICATION

Swap Test. The swap test,[45] shown in Figure 1, has been suggested for use in a quantum neuron and as a convolutional filter.[36−39] This idea is manifested by encoding the input data into one working register of qubits and by applying a parameterized quantum circuit to another register. When the controlled swap between the two states of these registers is executed, the expectation value of the ancilla qubit will produce the squared modulus of the inner-product between the two states $|\langle\phi|\psi\rangle|^2$ with error ϵ in $O\left(\frac{1}{\epsilon^2}\right)$ repetitions. In this sense, the amplitudes of the parameterized qubit register can be thought of analogously as the "weights". These "weights" are being multiplied together with the input data and summed when the swap test is conducted and the ancilla is measured. However, since the swap test only reveals the fidelity between two quantum states, it is directionless. The lack of the dot product's sign is problematic for flexible ML and places additional difficulty on the learning procedure. A remedy to this is quantum phase estimation (QPE), but the extra ancilla qubits required in QPE place an additional overhead on the resources required to run this algorithm.

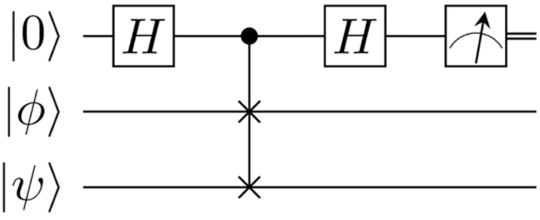


Figure 1. Swap test.

Hadamard Test. The Hadamard test, shown in Figure 2, is another cornerstone quantum algorithm[46] that has found use in calculating inner products for fields ranging from chemistry,[47] to physics,[48] to mathematics.[49] The test allows one to determine $\mathrm{Re}\langle\psi|U|\psi\rangle$ with error ϵ in $O\left(\frac{1}{\epsilon^2}\right)$ repetitions. However, by choosing the correct U, we can use this algorithm to produce the inner product between the two vectors of choice, $\mathrm{Re}\langle\psi|\phi\rangle$. Since this gives a direction with the inner product unlike the swap test, we may dispense with the need for QPE as well as the need for separate quantum registers for the input data and parametric "weights". Equations 1−4 show how U can be chosen for input data, $|\phi\rangle$, and "weights", $|\psi\rangle$.

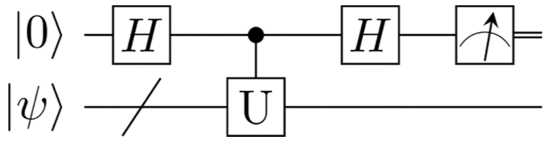


Figure 2. Hadamard test.

$$U_\psi|0\rangle^{\otimes n}=|\psi\rangle \tag{1}$$

$$U_\phi|0\rangle^{\otimes n}=|\phi\rangle \tag{2}$$

$$U = U_\phi U_\psi^\dagger \tag{3}$$

$$\mathrm{Re}\langle\psi|U|\psi\rangle = \mathrm{Re}\langle\psi|U_\phi U_\psi^\dagger|\psi\rangle = \mathrm{Re}\langle\psi|U_\phi U_\psi^\dagger U_\psi|0\rangle^{\otimes n} = \mathrm{Re}\langle\psi|\phi\rangle \tag{4}$$

Equation 4 shows that if U is chosen to be the product of the unitary matrix that prepares the quantum state whose amplitudes act as the "weights" and the adjoint of the unitary that prepares the input data, the real part of the desired inner product is obtained when performing the Hadamard test. This circuit is visualized in Figure 3.

We note that the fully complex-valued inner product $\langle\psi|\phi\rangle$ may easily be obtained. The Hadamard test may be slightly modified to include an $S^\dagger$ gate after the first Hadamard gate on

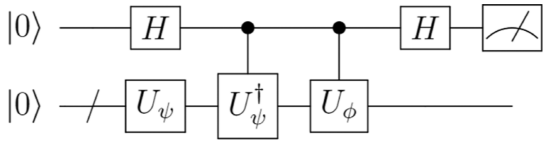


Figure 3. Quadratic matrix multiplication with the Hadamard test.

https://doi.org/10.1021/acs.jctc.4c00432
J. Chem. Theory Comput. 2024, 20, 4901−4908

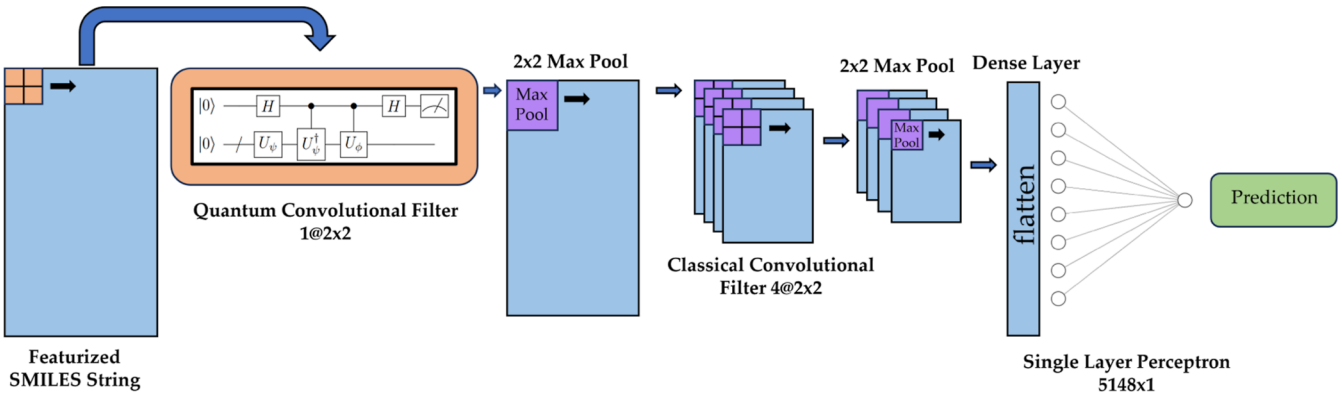


Figure 4. Quantum-classical neural network architecture.

the ancilla qubit to produce $\mathrm{Im}\langle\psi|\phi\rangle$. Summing the expectation values of these two different Hadamard tests yields the complex-valued inner product $\mathrm{Re}\langle\psi|\phi\rangle + i\mathrm{Im}\langle\psi|\phi\rangle = \langle\psi|\phi\rangle$.

As with the swap test, only one quantum circuit needs to be run for each resulting entry (or two circuits for complex-valued input data) when finding the product of two matrices and thus scales $O(n^2)$. While this method uses significantly fewer qubits than the swap test, it may come at the expense of deeper circuitry due to the controlled unitary gate $U_\psi^\dagger$ required to uncompute the input state. To this point, we note that both the swap test and the proposed Hadamard test method for multiplying arbitrary matrices with $O(n^2)$ complexity rely on the assumption of efficiently prepared quantum states.[35] As state preparation techniques become more refined, qubit control improves, and quantum devices become more coherent, the fulfillment of this assumption will also satisfy the concerns of the deeper circuitry possessed by this Hadamard test method. Concurrently, the computational overhead of more than double the number of qubits and the requirement for QPE associated with the swap test will not be alleviated as these are fundamental to the algorithm and are not dependent on quantum hardware.

■ ARCHITECTURE AND DATA SET

Hybrid Architecture. CNNs have been found to be successful for use in toxicity prediction tasks in the literature[50−52] and are very popular QML analogues of classical architectures.[53−57] The popularity of QCNNs partly stems from only needing to operate on small sections of input data at a time, rendering them feasible for NISQ devices. We employ a quantum-classical architecture where the first convolutional layer is replaced by a quantum circuit and is subsequently followed by classical pooling, a classical convolutional layer, and a fully connected layer, as depicted in Figure 4.

Tox21 Data Set. The Tox21 10k data set[43,44] is frequently used as a benchmark for evaluating ML models based on their binary predictive ability for toxicity. This data set comprises approximately 10,000 molecules represented as SMILES strings, each tagged with binary labels according to their response to 12 distinct assays. The Tox21 data set suffers from a significant imbalance, and although techniques such as SMOTE[58] can enhance performance, the primary objective of this study is to showcase the synergy between quantum and classical ML frameworks rather than attaining the highest level of predictive accuracy. We specifically choose to create a sparse binary feature map from each SMILES string in order to reduce overhead on the quantum component of the model. The computational benefit of binary features in a CNN is visually depicted in Figure 5. For each filter composed of w weights, there can exist a maximum of 2^w unique convolutions for binary data. Thus, for a single 2×2 filter, there are only 16 unique dot products that are required to fully stride across input data of any size. Additionally, the sparsity condition allows for the use of state-preparation techniques that do not scale exponentially,[59,60] which is a necessity along the road to achieving implementable quadratic complexity for matrix multiplication.

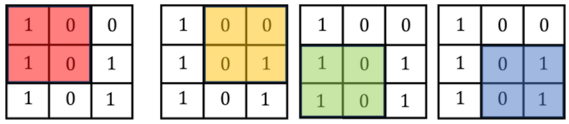


Figure 5. Binary data in a CNN. The red and green sections of this example input data will result in an identical output, and thus, only one of the two dot products needs to be computed. Since the number of unique dot products for binary data is 2^w, where w is the number of weights in a filter, for our input data of size 400×57 and a single 2×2 filter, there are only $2^{2\times2}$ different filters that will produce a unique dot product. Therefore, since the number of unique dot products is not dependent on the size of the input data, we only need to execute 15 circuits (the all-zero vector need not be computed) rather than the traditional 22,344 for each molecule. We note that the removal of nonunique calculations is applicable to both classical and quantum CNNs; however, as quantum circuits are inherently more expensive than classical computations in the NISQ era, this process of removing duplicates becomes particularly valuable for QCNNs.

In this study, we utilize the binary-encoded SMILES approach,[50] whereby each character within the SMILES string is depicted as a bit vector. Twenty-seven bits are used to encode the SMILES symbols and alphabet, and 30 bits are used to encode atomic properties. The vector of each character is joined with all other character vectors, forming a grid, and is padded to the longest string in the data set. The descriptors used to create this length 57 vector for each SMILES character are detailed in Table 1.

Quantum Transfer Learning. After training the quantum neural network (QNN), the real parts of the first column of the unitary matrix $U_{\phi'}$ that prepares the learned state $|\phi'\rangle$ are the classical weights that may be used to further train the model on a classical device. This is the case since $U_{\phi'}$ always operates on $|0\rangle^{\otimes n}$, which yields only the first column of $U_{\phi'}$. Since the input

https://doi.org/10.1021/acs.jctc.4c00432
J. Chem. Theory Comput. 2024, 20, 4901−4908

Table 1. Feature-Encoding Scheme for Each SMILES Character

feature type	element	bit(s)
symbol in SMILES	()	2
	[ ]	2
		1
	:	1
	=	1
	#	1
	\	1
	/	1
	@	1
	+	1
	-	1
	.	1
number in SMILES	atom charge (2−7)	6
	ring being (yes/no)	1
	ring end (yes/no)	1
atom type	C, H, O, N, others	5
others	surrounding hydrogen number (0−3)	4
	atom formal charge (−1, 0, 1)	3
	valence (1−5, other)	6
	ring atom (yes/no)	1
	degree (1−4, other)	5
	aromaticity (yes/no)	1
	chirality (R/S/others)	3
	hybridization (sp, sp^2, sp^3, sp^3d, sp^3d^2, unspecified, other)	7
total		57

data used in each quantum circuit must be L2 normalized to represent a valid quantum state, the corresponding classical component of the model must also work with normalized data to ensure a seamless transfer process. With regard to a CNN, this entails normalizing the data covered by a filter before taking the convolution with the weights and is shown in eq 5.

$|\mathbf{x}|_2$ is the normalized input data, x_k is the kth element of the n-dimensional input data vector, and $\mathbf{w}$ is the weight vector from the flattened convolutional filter.

$$\text{Output} = |\mathbf{x}|_2 \cdot \mathbf{w} = \left(\sqrt{\sum_{k=1}^{n} |x_k|^2}\right) \cdot \mathbf{w} \tag{5}$$

The workflow of training and transferring our Hadamard test-based model from a hybrid QNN to a fully classical network is demonstrated in Figure 6. The details of the quantum components of the hybrid architecture are shown in Figure 7.

METHODS

Hardware. All scripting was done in Python 3.9.18. The quantum simulation package PennyLane[61] version 0.35.0 was used, as well as PyTorch[62] version 2.1.0 with CUDA Toolkit[63] 11.8. An Intel i7−13700KF CPU, 12GB Nvidia GeForce RTX 3080Ti GPU, and 64GB of 3600 MHz CL18 RAM were used for all computations.

Data Set. The Tox21 data set was downloaded from the National Institute of Health.[44] RDKit version 2023.09.1 was used to load the SMILES string and calculate the atomic features shown in Table 1. SMILES strings with invalid valence shells and deuterium isotopes were discarded. 80% of each assay data set was used for training and 20% for the test set. To ensure that all binary-encoded SMILES files were of the same dimension, they were padded with zeros for a final shape of 400 × 57.

Machine Learning. Binary cross-entropy was used for the loss function, and weights were updated with the Adam optimizer.[64] A learning rate of 0.001 was used. Quantum weights were initialized from a uniform distribution between 0 and 2π. To ensure fair comparisons between the quantum and classical models, all other classical parameters were randomly initialized and copied to be identical between the models. The exponential decay rate for the first and second moment

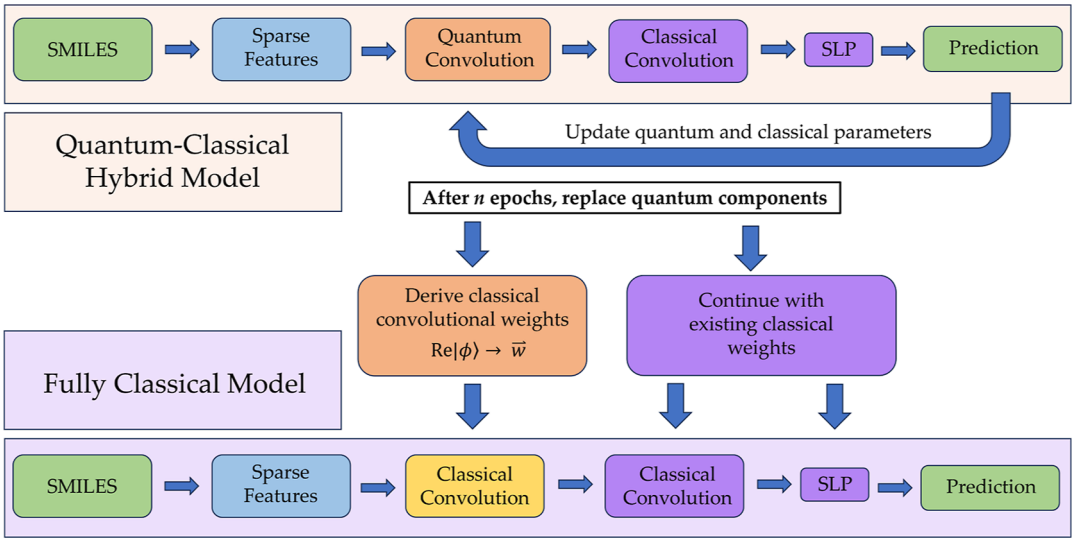


Figure 6. Quantum-to-classical neural network transfer pipeline. The convolutional weights are derived from the learned unitary matrix U_ϕ by operating this on the ground state, as shown in eq 2. The real part of the amplitudes of the evolved state $|\phi\rangle$ are identical to the normalized weights of a classical convolutional filter. The yellow classical convolutional block inherits the derived weights from the quantum state amplitudes. In order to maintain a mathematically exact learning transition between quantum and classical computations, this classical convolution L2-normalizes the input data covered by each filter. SLP indicates a single-layer perceptron. The exact architecture details are presented in Figure 4.

https://doi.org/10.1021/acs.jctc.4c00432
J. Chem. Theory Comput. 2024, 20, 4901−4908

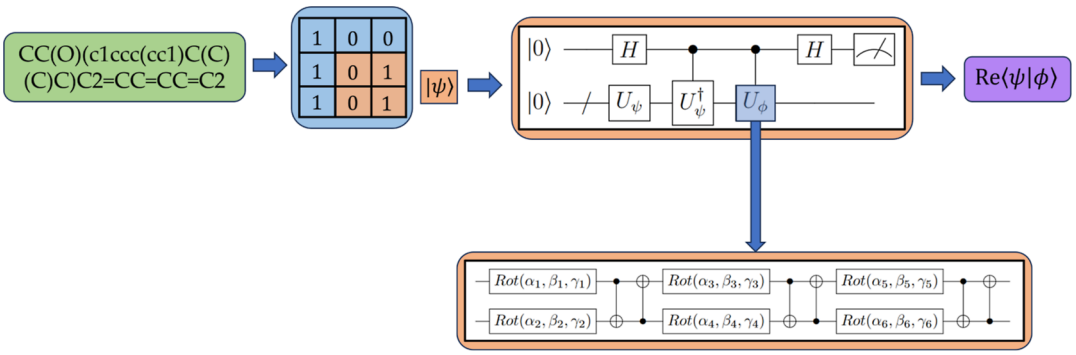


Figure 7. Variational quantum circuit used in the Hadamard test to calculate the real part of the inner product between an input data vector $|\psi\rangle$ and a learnable weight vector $|\phi\rangle$. The unitary matrix of the Rot gate is shown in eq 8.

estimations was set to 0.9 and 0.999, respectively, for all training. Epsilon was set to 1.0×10^{-8}. The bias component of the quantum convolution layer was trained classically. The output of each hidden layer in the model architecture was activated with ReLU. Given the imbalanced nature of the binary data set, we find it appropriate to measure performance with the area under the curve of the receiver operating characteristic (ROC−AUC).[65] This metric calculates the area under the curve of the plot of the true positive rate (eq 6) versus the false positive rate (eq 7). An ROC−AUC score of 1.0 indicates that the model predicts all samples correctly, while a score of 0.5 indicates that the model is no better than random chance.

$$\mathrm{TPR} = \frac{\text{true positive}}{\text{true positive} + \text{false negative}} \tag{6}$$

$$\mathrm{FPR} = \frac{\text{false positive}}{\text{false positive} + \text{true negative}} \tag{7}$$

Quantum Computing. Quantum states were prepared with the algorithm presented by Mottonen et al.[66] available in PennyLane; however, in practice, with this feature map, the algorithms by de Veras et al.[59] and Gleinig and Hoefler[60] would be better suited for use with NISQ devices. The variational component of the quantum circuit used three layers from PennyLane's *StronglyEntanglingLayers*, inspired by Schuld et al.[67] The unitary matrices comprising the general rotation gates are shown in eqs 8 and 9.

$$\mathrm{Rot}(\alpha, \beta, \gamma) = \mathrm{RZ}(\gamma)\mathrm{RY}(\beta)\mathrm{RZ}(\alpha) \tag{8}$$

$$\mathrm{RZ}(\alpha) = \begin{bmatrix} e^{-i\alpha/2} & 0 \\ 0 & e^{i\alpha/2} \end{bmatrix},$$

$$\mathrm{RY}(\beta) = \begin{bmatrix} \cos\left(\frac{\beta}{2}\right) & -\sin\left(\frac{\beta}{2}\right) \\ \sin\left(\frac{\beta}{2}\right) & \cos\left(\frac{\beta}{2}\right) \end{bmatrix} \tag{9}$$

The expectation value of the ancilla qubit in the computational basis was used as the input to the subsequent convolutional layer.

RESULTS AND DISCUSSION

We compare our results to a fully classical model with identical architecture except where the quantum convolutional filter is replaced by a classical filter of the same dimensions. It is shown in Table 2 that the hybrid QNN and fully classical neural network (CNN) perform similarly for each assay. On the overall data set, the quantum and classical models achieved ROC−AUC values of 0.692 and 0.689, respectively. These results demonstrate that the same accuracy as the fully classical $O(n^3)$ neural network can be achieved by the more efficient QNN with the $O(n^2)$ quantum component.

Table 2. Tox21 Test ROC−AUC

assay	QNN	CNN	QNN transferred to CNN
nr-ahr	0.619	0.632	0.619
nr-ar	0.725	0.764	0.727
nr-ar-lbd	0.800	0.847	0.800
nr-aromatase	0.724	0.722	0.726
nr-er	0.677	0.674	0.676
nr-er-lbd	0.641	0.623	0.640
nr-ppar-gamma	0.636	0.586	0.626
sr-are	0.701	0.705	0.701
sr-atad5	0.682	0.644	0.679
sr-hse	0.677	0.668	0.675
sr-mmp	0.722	0.698	0.721
sr-p53	0.692	0.689	0.691
average	0.692	0.689	0.691

Additionally, we demonstrate that the fully classical model is able to resume the training process started by the quantum model. As shown in Figure 8a,b, the training continues seamlessly after 5 epochs, where weights for the classical convolutional filter were derived from eq 2. The training curves for all assays are made available in the Supporting Information.

The first convolutional layer transferred from quantum training to classical training contains only a small portion of the model's total parameters. Despite this, the first layer of feature extraction plays a crucial role in the model's overall performance, and improperly computing the weights during the transfer procedure will have a demonstrable impact. To validate our framework of properly transferring these weights, we show the resulting model's performance after randomly initializing the weights from the first convolutional layer while copying the remaining learned weights. Epoch 6, the first epoch after transferring to a fully classical architecture, shows a spike in cross-entropy loss and a decrease in ROC−AUC in Figure 8c,d. Despite having so few parameters, this ablation study is indicative that our framework correctly transfers these weights to be fine-tuned classically.

https://doi.org/10.1021/acs.jctc.4c00432
J. Chem. Theory Comput. 2024, 20, 4901−4908

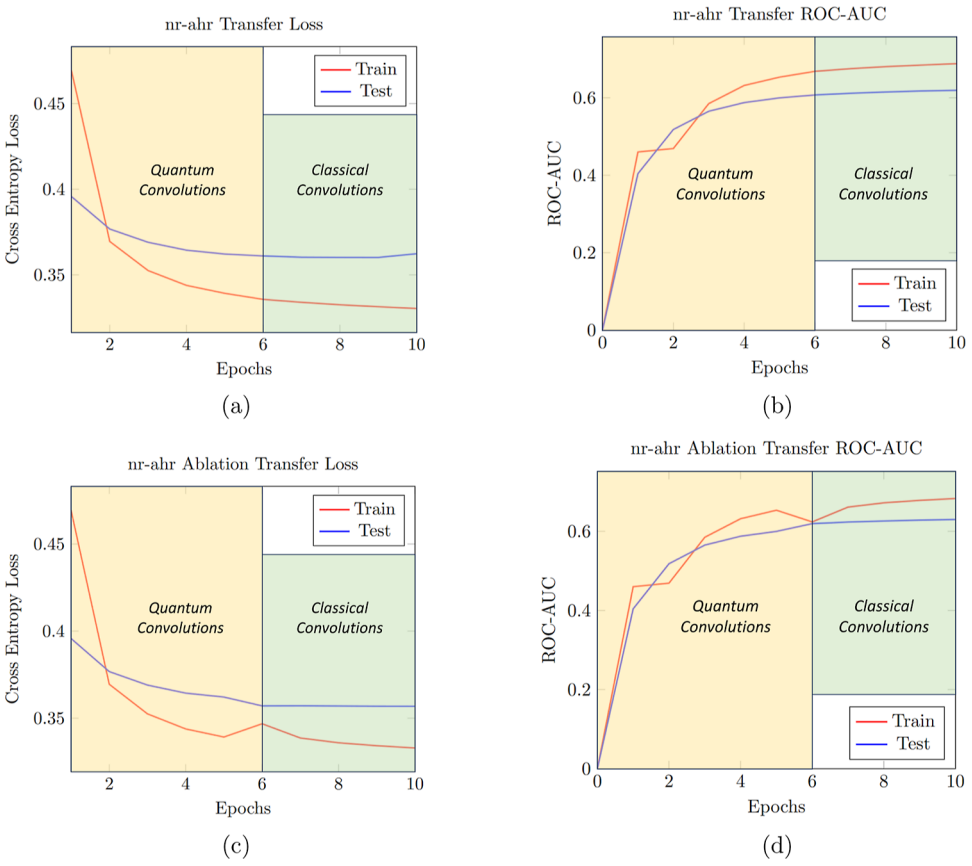


Figure 8. Model performance before and after weight transfer and ablation study results. (a) Transfer learning loss; (b) transfer learning ROC−AUC; (c) ablation study loss; (d) ablation study ROC−AUC. The yellow shading (epochs 1−5) indicates where the model was trained with quantum convolutions. The green shading (epochs 6−10) indicates where the model was trained with classical convolutions. The ablation study (c,d) shows the model's performance where the weights from the quantum layer were initialized to random values instead of deriving them from U_ϕ.

CONCLUSIONS

In summary, we present the first quantum-to-classical transfer learning framework in which weights initially learned from a quantum circuit can be fine-tuned classically. Assuming efficient state preparation, we show a theoretical quantum advantage by reducing the complexity of calculating matrix products from $O(n^3)$ to $O(n^2)$ using a modified version of the Hadamard test, which uses half the qubits of the previously proposed swap test for a quantum neuron and without the need for quantum phase estimation. We apply these methods to toxicity prediction using the Tox21 data set and show that our frameworks perform equivalently to a fully classical model with approximately the same number of parameters. As quantum devices improve and the assumption of efficient state preparation becomes fulfilled, we hope this framework will provide an implementable polynomial speedup for ML networks and consequently more efficiently navigate the chemical space to find life-saving pharmaceuticals.

ASSOCIATED CONTENT

Data Availability Statement

The data and code to reproduce the results presented here are available as open-source at https://github.com/anthonysmaldone/Quantum-to-Classical-Transfer-Learning.

Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acs.jctc.4c00432.

Parameter counts of the models and training curves of each Tox21 assay (PDF)

AUTHOR INFORMATION

Corresponding Author

Victor S. Batista − *Department of Chemistry, Yale University, New Haven 06511 Connecticut, United States;* orcid.org/0000-0002-3262-1237; Email: victor.batista@yale.edu

Author

Anthony M. Smaldone − *Department of Chemistry, Yale University, New Haven 06511 Connecticut, United States;* orcid.org/0009-0008-7265-0017

Complete contact information is available at: https://pubs.acs.org/10.1021/acs.jctc.4c00432

Author Contributions

A.M.S. designed the idea, A.M.S. developed the software, A.M.S. conducted the research, A.M.S. and V.S.B. analyzed the data, and A.M.S. wrote the article.

Notes

The authors declare no competing financial interest.

https://doi.org/10.1021/acs.jctc.4c00432
J. Chem. Theory Comput. 2024, 20, 4901−4908

■ ACKNOWLEDGMENTS

We acknowledge financial support from the National Institute of Health Biophysical Training Grant (1T32GM149438-01) [A.M.S.], as well as the financial support provided by the CCI Phase I: National Science Foundation Center for Quantum Dynamics on Modular Quantum Devices (CQD-MQD) under award no. 2124511 [V.S.B.]. Furthermore, we acknowledge seed funding from the Yale University through the National Science Foundation Engines Development Award: Advancing Quantum Technologies (CT) under award no. 2302908 and high-performance computer time from the National Energy Research Scientific Computing Center and from the Yale University Faculty of Arts and Sciences High Performance Computing Center.

■ REFERENCES

(1) Sun, D.; Gao, W.; Hu, H.; Zhou, S. Why 90% of clinical drug development fails and how to improve it? *Acta Pharm. Sin. B* **2022**, *12*, 3049−3062.
(2) Research and Development in the Pharmaceutical Industry | Congressional Budget Office. 2021; https://www.cbo.gov/publication/57126, (accessed 2024-04-08).
(3) Mullowney, M. W.; Duncan, K. R.; Elsayed, S. S.; Garg, N.; van der Hooft, J. J. J.; Martin, N. I.; Meijer, D.; Terlouw, B. R.; Biermann, F.; Blin, K.; et al. Artificial intelligence for natural product drug discovery. *Nat. Rev. Drug Discov* **2023**, *22*, 895−916.
(4) Sarkar, C.; Das, B.; Rawat, V. S.; Wahlang, J. B.; Nongpiur, A.; Tiewsoh, I.; Lyngdoh, N. M.; Das, D.; Bidarolli, M.; Sony, H. T. Artificial Intelligence and Machine Learning Technology Driven Modern Drug Discovery and Development. *Int. J. Mol. Sci.* **2023**, *24*, 2026.
(5) Volkamer, A.; Riniker, S.; Nittinger, E.; Lanini, J.; Grisoni, F.; Evertsson, E.; Rodríguez-Pérez, R.; Schneider, N. Machine learning for small molecule drug discovery in academia and industry. *Artif. Intell. Life Sci.* **2023**, *3*, 100056.
(6) Bian, Y.; Xie, X.-Q. Generative chemistry: drug discovery with deep learning generative models. *J. Mol. Model* **2021**, *27*, 71.
(7) Martinelli, D. D. Generative machine learning for de novo drug discovery: A systematic review. *Comput. Biol. Med.* **2022**, *145*, 105403.
(8) Tang, B.; Ewalt, J.; Ng, H.-L. In *Biophysical and Computational Tools in Drug Discovery*; Saxena, A. K., Ed.; *Topics in Medicinal Chemistry*; Springer International Publishing: Cham, 2021; pp 221−243.
(9) Semenova, E.; Williams, D. P.; Afzal, A. M.; Lazic, S. E. A Bayesian neural network for toxicity prediction. *Comput. Toxicol.* **2020**, *16*, 100133.
(10) Suzuki, T.; Katouda, M. Predicting toxicity by quantum machine learning. *J. Phys. Commun.* **2020**, *4*, 125012.
(11) Mayr, A.; Klambauer, G.; Unterthiner, T.; Hochreiter, S. DeepTox: Toxicity Prediction using Deep Learning. *Front. Environ. Sci.* **2016**, *3*..
(12) Nguyen, T.; Nguyen, G. T. T.; Nguyen, T.; Le, D.-H. Graph Convolutional Networks for Drug Response Prediction. *IEEE/ACM Trans. Comput. Biol. Bioinf.* **2022**, *19*, 146−154.
(13) Peng, Y.; Zhang, Z.; Jiang, Q.; Guan, J.; Zhou, S. TOP: A deep mixture representation learning method for boosting molecular toxicity prediction. *Methods* **2020**, *179*, 55−64.
(14) Jiang, J.; Wang, R.; Wei, G.-W. GGL-Tox: Geometric Graph Learning for Toxicity Prediction. *J. Chem. Inf. Model.* **2021**, *61*, 1691−1700.
(15) Du, Y.; Hsieh, M.-H.; Liu, T.; Tao, D. Expressive power of parametrized quantum circuits. *Phys. Rev. Res.* **2020**, *2*, 033125.
(16) Du, Y.; Hsieh, M.-H.; Liu, T.; Tao, D.; Liu, N. Quantum noise protects quantum classifiers against adversaries. *Phys. Rev. Res.* **2021**, *3*, 023153.
(17) Du, Y.; Hsieh, M.-H.; Liu, T.; You, S.; Tao, D. Learnability of Quantum Neural Networks. *PRX Quantum* **2021**, *2*, 040337.
(18) Caro, M. C.; Huang, H.-Y.; Cerezo, M.; Sharma, K.; Sornborger, A.; Cincio, L.; Coles, P. J. Generalization in quantum machine learning from few training data. *Nat. Commun.* **2022**, *13*, 4919.
(19) Gircha, A. I.; Boev, A. S.; Avchaciov, K.; Fedichev, P. O.; Fedorov, A. K. Hybrid quantum-classical machine learning for generative chemistry and drug design. *Sci. Rep.* **2023**, *13*, 8250.
(20) Batra, K.; Zorn, K. M.; Foil, D. H.; Minerali, E.; Gawriljuk, V. O.; Lane, T. R.; Ekins, S. Quantum Machine Learning Algorithms for Drug Discovery Applications. *J. Chem. Inf. Model.* **2021**, *61*, 2641−2647.
(21) Sajjan, M.; Li, J.; Selvarajan, R.; Sureshbabu, S. H.; Kale, S. S.; Gupta, R.; Singh, V.; Kais, S. Quantum machine learning for chemistry and physics. *Chem. Soc. Rev.* **2022**, *51*, 6475−6573.
(22) Mensa, S.; Sahin, E.; Tacchino, F.; Kl Barkoutsos, P.; Tavernelli, I. Quantum machine learning framework for virtual screening in drug discovery: a prospective quantum advantage. *Mach. Learn.: Sci. Technol.* **2023**, *4*, 015023.
(23) Kao, P.-Y.; Yang, Y.-C.; Chiang, W.-Y.; Hsiao, J.-Y.; Cao, Y.; Aliper, A.; Ren, F.; Aspuru-Guzik, A.; Zhavoronkov, A.; Hsieh, M.-H.; Lin, Y.-C. Exploring the Advantages of Quantum Generative Adversarial Networks in Generative Chemistry. *J. Chem. Inf. Model.* **2023**, *63*, 3307−3318.
(24) Domingo, L.; Djukic, M.; Johnson, C.; Borondo, F. Binding affinity predictions with hybrid quantum-classical convolutional neural networks. *Sci. Rep.* **2023**, *13*, 17951.
(25) Banerjee, S.; Yuxun, S. H.; Konakanchi, S.; Ogunfowora, L.; Roy, S.; Selvaras, S.; Domingo, L.; Chehimi, M.; Djukic, M.; Johnson, C. A hybrid quantum-classical fusion neural network to improve protein-ligand binding affinity predictions for drug discovery. 2023, arXiv:2309.03919. https://arxiv.org/abs/2309.03919.
(26) Dong, L.; Li, Y.; Liu, D.; Ji, Y.; Hu, B.; Shi, S.; Zhang, F.; Hu, J.; Qian, K.; Jin, X.; Wang, B. Prediction of Protein-Ligand Binding Affinity by a Hybrid Quantum-Classical Deep Learning Algorithm. *Adv. Quantum Tech* **2023**, *6*, 2300107.
(27) Vakili, M. G. et al. Quantum Computing-Enhanced Algorithm Unveils Novel Inhibitors for KRAS. 2024, arXiv:2402.08210. https://arxiv.org/abs/2402.08210.
(28) Hong, Z.; Wang, J.; Qu, X.; Zhu, X.; Liu, J.; Xiao, J. Quantum Convolutional Neural Network on Protein Distance Prediction. *2021 International Joint Conference on Neural Networks (IJCNN)*. 2021; pp 1−8, ISSN: 2161−4407.
(29) Albrecht, B.; Dalyac, C.; Leclerc, L.; Ortiz-Gutiérrez, L.; Thabet, S.; D'Arcangelo, M.; Cline, J. R. K.; Elfving, V. E.; Lassablière, L.; Silvério, H.; Ximenez, B.; Henry, L. P.; Signoles, A.; et al. Quantum Feature Maps for Graph Machine Learning on a Neutral Atom Quantum Processor. *Phys. Rev. A* **2023**, *107*, 042615.
(30) Bhatia, A. S.; Saggi, M. K.; Kais, S. Quantum Machine Learning Predicting ADME-Tox Properties in Drug Discovery. *J. Chem. Inf. Model.* **2023**, *63*, 6476−6486.
(31) Villalobos, P.; Sevilla, J.; Besiroglu, T.; Heim, L.; Ho, A.; Hobbhahn, M. Machine Learning Model Sizes and the Parameter Gap. 2022, arXiv:2207.02852. https://arxiv.org/abs/2207.02852.
(32) Strassen, V. Gaussian elimination is not optimal. *Numer. Math* **1969**, *13*, 354−356.
(33) Williams, V. V.; Xu, Y.; Xu, Z.; Zhou, R. New Bounds for Matrix Multiplication: from Alpha to Omega. 2023, arXiv:2307.07970. https://arxiv.org/abs/2307.07970.
(34) Miller, W. Computational Complexity and Numerical Stability. *SIAM J. Comput.* **1975**, *4*, 97−107.
(35) Shao, C. A Quantum Model for Multilayer Perceptron. 2018, arXiv:1808.10561. https://arxiv.org/abs/1808.10561.
(36) Stein, S. A.; Mao, Y.; Ang, J.; Li, A. QuCNN: A Quantum Convolutional Neural Network with Entanglement Based Back-propagation. 2022, , arXiv:2210.05443. https://arxiv.org/abs/2210.05443.
(37) Li, P.; Wang, B. Quantum neural networks model based on swap test and phase estimation. *Neural Networks* **2020**, *130*, 152−164.

https://doi.org/10.1021/acs.jctc.4c00432
J. Chem. Theory Comput. 2024, 20, 4901−4908

(38) Zhao, J.; Zhang, Y.-H.; Shao, C.-P.; Wu, Y.-C.; Guo, G.-C.; Guo, G.-P. Building quantum neural networks based on a swap test. *Phys. Rev. A* **2019**, *100*, 012334.
(39) Shao, C. Quantum Algorithms to Matrix Multiplication. 2018, arXiv:1803.01601. https://arxiv.org/abs/1803.01601.
(40) Mari, A.; Bromley, T. R.; Izaac, J.; Schuld, M.; Killoran, N. Transfer learning in hybrid classical-quantum neural networks. *Quantum* **2020**, *4*, 340.
(41) Kim, J.; Huh, J.; Park, D. K. Classical-to-quantum convolutional neural network transfer learning. *Neurocomputing* **2023**, *555*, 126643.
(42) Schuman, D.; Sünkel, L.; Altmann, P.; Stein, J.; Roch, C.; Gabor, T.; Linnhoff-Popien, C. Towards Transfer Learning for Large-Scale Image Classification Using Annealing-based Quantum Boltzmann Machines. 2023, , arXiv:2311.15966. https://arxiv.org/abs/2311.15966.
(43) Tice, R. R.; Austin, C. P.; Kavlock, R. J.; Bucher, J. R. Improving the human hazard characterization of chemicals: a Tox21 update. *Environ. Health Perspect.* **2013**, *121*, 756−765.
(44) National Institute of Health Tox21 Data Challenge 2014. https://tripod.nih.gov/tox21/challenge/data.jsp, (accessed 2024-02-29).
(45) Buhrman, H.; Cleve, R.; Watrous, J.; de Wolf, R. Quantum Fingerprinting. *Phys. Rev. Lett.* **2001**, *87*, 167902.
(46) Cleve, R.; Ekert, A.; Macchiavello, C.; Mosca, M. Quantum algorithms revisited. *Proc. R. Soc. London A* **1998**, *454*, 339−354.
(47) Wang, Y.; Mazziotti, D. A. Electronic excited states from a variance-based contracted quantum eigensolver. *Phys. Rev. A* **2023**, *108*, 022814.
(48) Miquel, C.; Paz, J. P.; Saraceno, M.; Knill, E.; Laflamme, R.; Negrevergne, C. Interpretation of tomography and spectroscopy as dual forms of quantum computation. *Nature* **2002**, *418*, 59−62.
(49) Aharonov, D.; Jones, V.; Landau, Z. A Polynomial Quantum Algorithm for Approximating the Jones Polynomial. *Algorithmica* **2009**, *55*, 395−421.
(50) Chen, J.; Cheong, H.-H.; Siu, S. W. I. BESTox: A Convolutional Neural Network Regression Model Based on Binary-Encoded SMILES for Acute Oral Toxicity Prediction of Chemical Compounds. *Algorithms for Computational Biology*; Springer, 2020; Vol. *12099*, pp 155−166.
(51) Yuan, Q.; Wei, Z.; Guan, X.; Jiang, M.; Wang, S.; Zhang, S.; Li, Z. Toxicity Prediction Method Based on Multi-Channel Convolutional Neural Network. *Molecules* **2019**, *24*, 3383.
(52) Limbu, S.; Zakka, C.; Dakshanamurthy, S. Predicting Dose-Range Chemical Toxicity using Novel Hybrid Deep Machine-Learning Method. *Toxics* **2022**, *10*, 706.
(53) Cong, I.; Choi, S.; Lukin, M. D. Quantum convolutional neural networks. *Nat. Phys.* **2019**, *15*, 1273−1278.
(54) Hur, T.; Kim, L.; Park, D. K. Quantum convolutional neural network for classical data classification. *Quantum Mach. Intell.* **2022**, *4*, 3.
(55) Chen, S. Y.-C.; Wei, T.-C.; Zhang, C.; Yu, H.; Yoo, S. Quantum convolutional neural networks for high energy physics data analysis. *Phys. Rev. Res.* **2022**, *4*, 013231.
(56) Herrmann, J.; Llima, S. M.; Remm, A.; Zapletal, P.; McMahon, N. A.; Scarato, C.; Swiadek, F.; Andersen, C. K.; Hellings, C.; Krinner, S.; et al. Realizing quantum convolutional neural networks on a superconducting quantum processor to recognize quantum phases. *Nat. Commun.* **2022**, *13*, 4144.
(57) Smaldone, A. M.; Kyro, G. W.; Batista, V. S. Quantum convolutional neural networks for multi-channel supervised learning. *Quantum Mach. Intell.* **2023**, *5*, 41.
(58) Chawla, N. V.; Bowyer, K. W.; Hall, L. O.; Kegelmeyer, W. P. SMOTE: Synthetic Minority Over-sampling Technique. *J. Acad. Ind. Res.* **2002**, *16*, 321−357.
(59) de Veras, T. M. L.; da Silva, L. D.; da Silva, A. J. Double sparse quantum state preparation. *Quantum Inf Process* **2022**, *21*, 204.
(60) Gleinig, N.; Hoefler, T. An Efficient Algorithm for Sparse Quantum State Preparation. In *2021 58th ACM/IEEE Design Automation Conference (DAC)*: San Francisco, CA, USA, 2021, pp 433−438.
(61) Bergholm, V. et al. PennyLane: Automatic differentiation of hybrid quantum-classical computations. 2022, arXiv:1811.04968. https://arxiv.org/abs/1811.04968.
(62) Paszke, A. et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. 2019, arXiv:1912.01703. https://arxiv.org/abs/1912.01703.
(63) *CUDA Toolkit - Free Tools and Training*. 2024. https://developer.nvidia.com/cuda-toolkit.
(64) Kingma, D. P.; Ba, J. Adam: A Method for Stochastic Optimization. 2017, arXiv:1412.6980. https://arxiv.org/abs/1412.6980.
(65) Hanley, J. A.; McNeil, B. J. A method of comparing the areas under receiver operating characteristic curves derived from the same cases. *Radiology* **1983**, *148*, 839−843.
(66) Mottonen, M.; Vartiainen, J. J.; Bergholm, V.; Salomaa, M. M. Transformation of quantum states using uniformly controlled rotations. 2004, arXiv:quant-ph/0407010. https://arxiv.org/abs/quant-ph/0407010.
(67) Schuld, M.; Bocharov, A.; Svore, K.; Wiebe, N. Circuit-centric quantum classifiers. *Phys. Rev. A* **2020**, *101*, 032308.