Rob Grau

svb558

May 7, 2025

Deep Learning advancements in Quantum Machine Learning

Table of Contents

Introduction

Quantum Recurrent Neural Networks for Sequential Learning

Nav-Q: quantum deep reinforcement learning for collision-free navigation of self-driving cars

Q-SupCon: Quantum-Enhanced Supervised Contrastive Learning Architecture within the Representation Learning Framework

Quantum Transfer Learning for Sentiment Analysis: an experiment on an Italian corpus

Quantum deep learning-based anomaly detection for enhanced network security

Deep Q-learning with hybrid quantum neural network on solving maze problems

Quantum deep generative prior with programmable quantum circuits

Conclusion

References

Appendix A – Classification

Appendix B – Annotated Bibliography

Introduction

Quantum machine learning is a relatively new field of quantum computing with most major developments coming in the past 10-15 years. Quantum computing promises advantages over classical or non-quantum computing by way of several properties of quantum mechanics. Quantum superposition allows multiple states to be represented simultaneously allowing for parallel computation of problems. In comparison to a classical computer’s storage of information on a bit or, for clarity, a classical bit, a quantum computer stores information in a quantum bit or qubit. Additionally, quantum entanglement allows for the qubits to interact with each other opening up a range of possibilities that are not possible in classical computing.

John Preskill summarized some of the challenges facing quantum computing in a 2018 paper[4] that still hold true today. Quantum computing suffers from several drawbacks related to the quantum hardware necessary to perform operations. Notably, quantum particles need to be isolated from other particles in order to behave according to their quantum properties. However, it is necessary to interact with a quantum environment in order to process data or convey instructions. This conflict leads to quantum noise and state degradation, called decoherence, that limit the circuit depth and complexity that can be executed on a quantum computer.

Preskill also coined a term for the current hardware state, NISQ, which stands for Noisy Intermediate-Scale Quantum- Noisy means control of qubits imprecise and intermediate scale means that only 50 to a few hundred qubits can be successfully included in a program before the error rate becomes to large to receive a meaningful result. This makes functional error correction necessary to stabilize qubit storage into something that allows for a large scale. Finally, there isn't an easy way to move large amount of data from classical to quantum so encoding large amounts of date in quantum RAM has high overhead.

Even with these considerable challenges, quantum deep learning has been developed and more recently moved from early theoretical work into concrete experiments run on modern NISQ hardware.

Quantum Recurrent Neural Networks for Sequential Learning

In Yunan et al, [10] authors want to design a Quantum Recurrent Neural Network that's realistic for the NISQ hardware environment that exists in today's quantum world. If one were to take an approach to the quantum circuit where all QRNN circuits are run one after another, mirroring the classical RNN structure directly, then there will unfortunately will be decoherence problems on today's hardware. The key idea of the paper is a staggered quantum RNN circuit where the qubits are rotated to prevent decoherence.

It's helpful to momentarily review what a classical RNN is. The R stands for recurrent and is used for sequential / temporal data. The classical recurrent network uses a function that takes the input from the most recent item in the time series and combines it with the output from the last step/layer to create an updated state that may be transformed based on activations. After activations, you have the new output layer. The most recent layer’s output in the classical RNN implicitly contains output from all of the previous layers because of recursion in the RNN structure.

In regards to challenges with design of a quantum circuit, a common bottleneck for hybrid classical-quantum algorithms is the encoding of classical data into quantum qubits. Amplitude encoding is a commonly used method to encode exponential classical bits into quantum superpositions, the downside is that quantum circuits grow exponentially with more datapoints. Alternately, angle encoding of classical data takes the datapoints and represents them as angle rotations around the computational basis. With angle encoding, qubits can just be equal to the number classical datapoints. This is more efficient and because more qubits lead to more errors in an NISQ environment, efficiency is a requirement. Thus the authors chose to use angle encoding on the data. The authors took error rates into consideration, and actually used three qubits for a single classical bit. This is because of unreliable error correction to which the alternative is redundancy.

An ansatz design is used to perform the work of the quantum RNN. The functionality of an ansatz circuit is provided by a parameterized quantum circuit that has adjustable quantum gates. A quantum gate is a unitary operation transformation performed on a qubit. Because the gates are adjustable, the values of the gates in the ansatz are what is trained by the model.

For measurement, the process must be rerun multiple times. This is because a quantum state is represented by probabilities of the state belonging to a basis state. When quantum measurement occurs, the result is only one basis state. Only after multiple runs will an estimate of the probabilistic expectation be available.

Paper then discusses what a quantum QRNN would look like in a closed system or theoretical system where noise and decoherence were not factors. In this setup, you could model the qRNN directly after a classical RNN setup. In this design, a quantum register which is a set of qubits, takes a new input datapoint and a second register that holds the ongoing value of the variables. The ansatz is applied across all of the quantum bits and a measurement of one of the state qubits is made to create an output value. While quantum register with new data is reset prior to each input, the other register is not reset and holds its value throughout the process. In open systems or real world scenarios with NISQ hardware, it is not realistic to expect the state qubits to hold onto their state value for many iterations.

This leads us to the proposed staggered qRNN model, where after each iteration of the RNN process, the qubit register holding the system state shift by one qubit. For example, if there are six qubits, three for the state register and three for the new input register, then only two of the three qubits holding the state will remain for the next iteration, paired with one of the qubits that was previously used for new data. The remaining three qubits will be reset and used for new data. In this scenario, after three iterations the state qubits will have completely shifted to the qubits that were previously used for input data. In fact, the state will only have to be maintained for n+1 iterations where n is the number of qubits in the quantum register. This is the key advantage of the staggered qRNN over the plain qRNN.

There is another challenge to implementing classical machine learning methods on quantum computers and this comes in the form of optimizing a loss function. Unlike classical machine learning, backpropagation cannot be computed because that relies on saving the state of the environment, an impossibility in quantum computing

An approximate derivative can still be calculated on the quantum computer, it requires running the quantum circuit twice with slightly different values and using the difference for computing loss functions.

Researchers examined the performance of two QRNN systems (called plain and staggered) are then compared to a classical RNN that is configured similarly to the QRNN setup.

In the experiment, a circuit iteratively took seven days of input data which was then used to generate a prediction. The two quantum RNN models performed similarly to the classical RNN on actual data and were even more accurate for a few features such as wind speed in meteorological data. The study also varied the size of the qubit registers of qubits. Prediction accuracy increased as quantum registers grew from 4 to 6 to 8 qubits.

The experiments showed that a staggered quantum RNN model could be both accurate and possible to implement on modern hardware.

Nav-Q: quantum deep reinforcement learning for collision-free navigation of self-driving cars

A recent paper by Sinha, et al [7] discusses quantum reinforcement algorithms used in the self driving car domain. The authors note that they are not the first to consider quantum reinforcement learning however the earlier work have been simulated in simplified environments while demonstrating collision avoidance on self driving cars is more realistic test for quantum reinforcement learning.

Central to any work involving self driving cars is real-time optimization of on-board camera data. The classical approach to utilize partially observable Markov decision processes and Deep Reinforcement Learning. The proposed model is a hybrid quantum classical model where the key idea is to have a quantum computer do the work of the reinforcement learning agent. In particular, the authors use an actor-critic model for the agent. One particular appeal of this model is that the quantum circuit is only used in the critic because that algorithm is only needed for training. This is important because any part of the system that is used in testing will have to be part of the car design. Today’s quantum computers are still housed in large datacenters with carefully controlled environments. No vehicle comes close to being a feasible host for a quantum computer.

In the current classical training for self-driving cars, NavA2C is the baseline on which the authors built their quantum design. NavA2C combines the car’s path planner with a modified reinforcement agent. The deep reinforcement learning agent has four main components: an encoder, an actor, a critic and an LSTM cell. The NavA2C agent is responsible for calculating the speed action for the car. The Nav-q critic uses a parameterized quantum circuit that takes an input of the hidden state from the LSTM cell.

A parameterized quantum circuit is a common approach seen in quantum machine learning designs. At a general level, the system state is encoded onto the qubits of the quantum circuit and then it is run through what’s known as an ansatz which is a rotational gate where the angle can be set programmatically. The output of the circuit can then be taken and classified.

In the initial design of the Nav-q quantum circuit, 24 classical parameters from the LSTM cell are encoded onto 4 qubits across two layers. The output is taken from the quantum circuit and linearly recombined as a fully connected layer that feeds back into the LSTM cell. The authors noted the quantum process was actually quite slow for this design and they subsequently reduced the design to 2 qubits with 1 layer that only took 6 parameters from the LSTM cell hidden state.

After model refinement, training occurred and the designs were tested in standard CARLA driving model simulators. The quantum model performed well in optimal route model while the classical model outperformed the quantum model in terms of collision avoidance. Aside from simulator performance, the authors investigated the percentage of active parameters relative to the total number of parameters in the model, called the normalized effective dimension. The number was 10 times higher for the quantum model than the classical model, suggesting that the quantum model has a better ability to model more complex functions.

The quantum system still has drawbacks. Although the authors ran the Nav-Q model through a more complex driving simulation than previous works, it was still simulated in a noiseless quantum environment, which unfortunately is not how modern quantum hardware operates. When noise was introduced into the Nav-Q simulations, the stability of the learning rate decreased significantly and the time required to train the model increased.

Q-SupCon: Quantum-Enhanced Supervised Contrastive Learning Architecture within the Representation Learning Framework

Don et al. [6] addess dataset sparsity with a method called Q-SupCon, which is short for quantum contrastive learning. Contrastive learning is a method of comparing pairs of similar and dissimilar samples in order to learn from both the maximum and minimum values of similarity.

Deep learning networks have a need for large datasets to complete proper training, however, some domains such as healthcare have privacy restrictions which create barriers to collecting a large dataset. This presents the setup for the ‘curse of dimensionality' where a spare dataset lacks enough datapoints to be accurately represented with a lower dimension model, however higher dimension models require more data to fight overfitting.

When quantum computing can encode complex representations of data into a smaller space, the data behaves differently. This is why it investigated for use in contrastive learning as it could potentially simplify the process or combat overfitting. Researchers created a model with four separate quantum functions—data augmentation, encoder, projection head, and classifier.

The quantum data augmentation was developed as part of a previous work by [ref 8]. In this application, small parts of the image were sampled and encoded in the quantum circuit. Rotation gates are applied and a further not gate will flip bits to introduce noise. The noise is applied the original image to create more data samples.

The quantum autoencoding function employes a parameterized quantum circuit to compress the images into fewer qubits. Some of the bits are transformed into the feature vector that will be classified. Parameterized quantum circuits include angle rotation gates where the angle of the gate is trainable.

The third quantum feature is the projection head. It acts as a fully connected layer after the autoencoder. Notably, it is only utilized for training the autoencoder’s parameters and is not needed for the classifier.

The final component of the algorithm is a variational quantum classifier method is used on the feature vectors to determine classification. In training the VQC was optimized with a gradient method and researchers had to deal with gradient dropout that is sometimes called a barren plateau. An initial guess was used with cross entropy to deal with dropout.

The authors created a model that perform fully quantum supervised learning on actual quantum hardware, specifically using trapped ion infrastructure. This is impressive given the aforementioned state of NISQ hardware. Even so, some compromises had to be made on the evaluation datasets to make it possible to run. The MNIST, FMNIST, KMNIST datasets were used, however, image size was reduced to 8 x 8 pixels. The full datasets were not used either. The training datasets were reduced to 25-150 images per class and classes were randomly sampled multiple times. Images used for training in were not reused with a different random sample of classes.

After testing on similarly limited datasets of 25 images, test accuracy for Q-SupCon on the MNIST and FMNIST was 80% while KMNIST had a 60% accuracy. The classical ResNet baseline had a 20% accuracy on MNIST and FMNIST and a 40% accuracy on KMNIST. At 150 test samples, accuracy for both quantum and classical models was more than 90%.

These results are very promising given that barriers to data collection will likely remain in place. Quantum remains an expensive method for computation, however, this shows how it can be a benefit in spares data scenarios.

Quantum Transfer Learning for Sentiment Analysis: an experiment on an Italian corpus

In a paper by Buonaiuto et al. [5] , quantum computing is investigated to see if it offers any advantages to natural language processing (NLP). NLP is resource intensive and the hope is that quantum computing may be able to alleviate some of the resource pressures.

Within the field of quantum computing the task of sentiment analysis was selected for study. Sentiment analysis is a standard NLP task where the primary emotional tone in a text is identified. Additionally, if quantum computing can be used to combine the sentient analysis with document retrieval makes the retrieval process more intelligent and powerful. This research looks at quantum embedding where quantum concepts of superposition and entanglement are used to capture semantic relationships between words.

Quantum approach is unsupervised machine learning using a density matrix generated from two sentiment dictionaries. This work features a Variational Quantum Classifier (VQC) that is used with a language model to make classifications of sentiment. The variational quantum classifier is an ansatz with a single rotational gate followed by CNOT gates. During training, an optimizer finds the quantum parameters that best capture the target labels. This is a relatively straightforward implementation of a VQC.

The workflow is as follows: Text is fed into a classical NLP model called Electra. Electra is used to extract work vectors. The vectors are then encoded in to a quantum circuit and sent to the VQC. After passing through the VQC, word relations identified in quantum circuit are then used as inputs to a classical LSTM. High dimension word vectors are encoded into a probablistic space with a generative tensor network to classify text.

The current NISQ hardware limits quantum NLP to relatively easy tasks or small datasets. In the case of this experiment, the training and test datasets were reduced to 4476 sentences for training and 500 sentences for test. On the other hand, the dataset had 768 features and they were all encoded into 10 qubits. That is possible because quantum superposition allows an exponential number of classical data points to be represented. The log (base 2) of 768 falls between 9 and 10, so the 768 features were represented with just 10 qubits. The extra information for the unneeded features 769 through 1024 were filled in with zeros. Finally, although this experiment used a simulator for the quantum circuit, the number of qubits and the design of the VQC are sufficiently compact that it would have been feasible to run this on modern quantum hardware.

The results of the experiment were measured using an F1 score. The hybrid quantum circuit achieved an F1 score of 0.77 while the classical baseline achieved an F1 score of 0.75. Some other notes, the classical baseline algorithm used preprocessing of the language data to get a higher F1 score where as the quantum algorithm did not do any pretraining. The author’s classical baseline was not exactly the same as quantum model, so it’s hard to extrapolate too much. It does show that there are positives in the quantum model that may indeed come from the quantum properties.

Quantum deep learning-based anomaly detection for enhanced network security

In Hdaib et al, [2] three different quantum approaches to machine learning are compared to see what is most effective at detecting web traffic anomalies. Additionally, a quantum encoder is employed to compress the classical data to a format that can be processed by the quantum algorithms. The three proposed algorithms are Random Forest, kNN, and a one layer SVM.

Web security involves monitoring traffic, finding anomalies and mitigating them. Quantum machine learning has not commonly been used for this application. The authors want to find out if quantum's ability to process correlated features through entanglement will improve on classical approaches to web security. The previous work in quantum anomaly detection used supervised learning. The proposed algorithms are forms of unsupervised learning.

The quantum encoder in this application is not simply converting classical information into quantum. It has a parameterized quantum circuit that compresses the state but also can be trained on the parameters similar to the principal component analysis machine learning method where data is projected onto lower dimension vectors. After the state is compressed, fresh qubits are introduced to reconstruct the state. The encoder can be used on its own to detect anomalies based on how much error was in the reconstruction of the quantum state.

Subsequent to the encoder circuit, three different machine learning approaches are used by way of different quantum circuit design. The random forest has another parameterized quantum circuit, however this ansatz is more complicated than just a single rotation gate like many other PQC are. It uses multiple gates to map the feature map of the input onto the quantum qubits. For quantum kNN, the quantum circuit uses a swap test to measure proximity of the datapoints. For three dimensional proximity such as the measurements for quantum particles the angle between the two distance vectors is measured by taking the inner product of the two vectors. A quantum swap test followed by a measurement performs this function. The one layer SVM design uses single z-axis rotation gates followed by 2 qubit z-axis rotation gates. Finally, a binary classification is performed on the data (with all of the different algorithms) where the outcome is either an anomaly in web traffic or not.

For the experiments, the web anomaly datasets KDD99, IOT-23, and CIC IOT 23 were tested. Prior to the work, data preprocessing was performed. Of note, there is a class imbalance challenge with web anomaly detection, so undersampling was used to reduce the majority class without eliminating anything from the minority class. Normalization, one hot encoding and PCA reduction was also used. The experiments were run on IBM superconductor quantum hardware, however, the parameter optimization was first computed with a qiskit simulator and pytorch. For completeness, the models were compared against a classical baseline.

Because of the class imbalance issue, accuracy scores were not very revealing, so F1 scores were used instead of evaluation. The quantum kNN framework achieved the highest F1 score of any model at 98.26% on the CIC IOT 23 data. The quantum SVM performed well on the network flow data in the KDD99 dataset where its 97.19% F1 score was the best of any other model on the KDD99 data. The quantum models all outperformed their classical counterparts.

Despite the encouraging results, the authors note that the quantum experiments were limited by the amount of preprocessing and quantum circuit modification that had to be performed to feed the real world data into the quantum circuits. Additionally, the authors shared information on the autoencoder training relative to the number of qubits. At 4 total qubits with 2 latent qubits, training time was 1.64 hours with a fidelity of 99.6%. With a small increase to 6 qubits and 4 latent qubits, training time jumped to 7.85 hours with a drop in fidelity to 86.1%. The scaling problem is immediately evident.

Deep Q-learning with hybrid quantum neural network on solving maze problems

In Deep-Q learning with hybrid neural networks by Chen, et al [1] use quantum deep learning to solve maze problems. These problems are solved with reinforcement learning using a reward signal, specifically the Markov decision making process (MDP). Solving a maze problem classically can be computationally expensive and the hope is that quantum computing will reduce the overall complexity of the task.

In MDP, an agent observes the state and then selects an action and receives a reward based on the desirability of the action. The process repeats at the next state and the agent is expected to optimize reward. One option researchers have is to create a quantum walk algorithm and a variational quantum circuit (VQC) to encode classical agents with state information. The second option is to use a fully quantum model. The fully quantum model is promising but hampered by state of NISQ hardware.

The author’s eventual quantum approach is a hybrid one that uses a 'model free' method of estimating state-action pairs and then using that estimate to update the value with a parameterized (VQC). The model free method is referred to as Q-learning as the model estimates a Q-value or cumulative reward.

In the proposed hybrid model, a classical agent updates a quantum learning table by estimating quantum states and storing them. The agent selects the highest Q-score value available to it and gets a reward value from the environment. Those are sent to the DNN. To prevent correlation between samples, the agent may also select a random past experience. This is an effort to prevent overfitting. On the DNN side, the neural network takes the input and outputs a range of actions and Q-score.

In terms of quantum structure, a variational quantum circuit is used as the last layer of the DNN. The quantum circuit design uses two qubits which are manipulated by feature encoding and ansatz module to return the result. The feature encoding with z-axis gates and the ansatz alternates between y-axis rotation gates and cnot gates for entanglement.

For the experiment, the algorithm had to search 4x4 and 5x5 mazes. As is typical with maze problems, a path had to be created that went from the entry cell to the exit cell without touching any walls. A classical CNN was also run in the same environment as a baseline.

The experiment results favored the classical CNN. The classical CNN had a better win rate in 4x4 maze (89.94% vs 85.19%) and a much shorter training time (only 1/8th of qNN training). The quantum CNN improved its win rate on 5x5 maze (93.13%) but still was inferior to traditional CNN which had a 94.87% win rate and faster training time.

The researchers did not utilize NVIDIA’s cuQuantum platform but are hopeful that future work with it may yield training advancements for their structure. In the meantime, the quantum environment remains feasible but strictly worse than classical.

Deep Ensemble learning and quantum machine learning approach for Alzheimer’s disease detection

A novel application of quantum deep learning was presented by Belay et al [3]. in which an ensemble approach is applied to quantum machine learning. The research utilized a set of publicly available ADNI dataset of MRI scans for Alzheimer's disease patients. In this task, deep learning checks for disease markers in the brain scans and classifies the images.

Two classical deep learning models (VGG16 and ResNet50) are run separately and instead of a final classification layer, the output from each is flattened, concatenated, and fed into a quantum SVM classifier. There are a number of deep neural networks that have been created to solve this problem in related work section. A 2023 model reportedly achieved 99.68% accuracy, which is considered state of the art.

The model was run a number of times with different input models. The VGG16 and ResNet50 models were run separately and as an ensemble without the SVM classifier. They were also run with both a classical and quantum SVM classifier. The parts of the models before SVM were all run in the same configuration. VGGNet has nine convolutions, two batch normalizations, three max-pooling, two dropouts, and one flattened layer. ResNet 50 model has one convolution, one max-pooling, one average polling, one batch normalization, one activation, two identity blocks, three conv blocks, and one flattened layer.

Results for the classical algorithms without any SVM were unremarkable, with VGGNet having a 90.11% accuracy rate and the ensemble 90.58%. ResNet50 alone scored only 75.04%. With the classical SVM the accuracy results were 85.24% for VGGNet, 82.24% for ResNet50, and 86.78% for the ensemble model which was actually a step backwards. The authors did note that the classical SVM scores were lower because they used pretrained models that were not trained on the ANSI MRI images. With the quantum SVM the results were much better with 95.65% accuracy for VGGNet, 91.56% accuracy for ResNet50, and 99.89% accuracy for the ensemble model.

It should be noted that to fight class imbalance, data augmentation was performed and the quantum models were all simulated so they did not actually achieve this result in a modern NISQ hardware environment. Even with that, it remains an encouraging result for the use of quantum ensemble models.

Quantum deep generative prior with programmable quantum circuits

In Quantum deep generative prior with programmable quantum circuits by Xiao et al, [9] the authors discuss the role of quantum computers in generative tasks. They investigate what is possible to accomplish in the near future with NISQ hardware. One common approach in quantum machine learning is to utilize a parameterized quantum circuit (PQC). It's difficult to scale the depth and width of PQC circuits with hardware limitations.

The authors decide to test classical image generation via GAN assisted by a PQC. The authors design a range of computer vision tasks to be completed such as ghost imaging where a low resolution image is created and the quantum algorithm must reconstruct it.

For this process, classically CNNs capture image statistics during processing that can be treated as a prior image. Higher dimension images can be approximated by a learnable generator (the GAN) that captures a latent space. The idea is that a quantum deep generative prior (QDGP) using PQC to generate a latent space will be able to capture a larger dimension thanks to quantum capabilities.

For the experiment, Images were created by taking an original image and transforming it into a degraded version. These ghost Imaging priors can be created using extremely low light with as little as a few photons. Research with a classical solution to this problem is ongoing, however, larger datasets are often needed for training in this situation. As with many applications in data science, it is not always possible to have a robust dataset such as that. The sparse training data was simulated in the tests.

Additionally, both classical and quantum models were tested with both pretrained GANs and random initialization. A classical non-GAN model using a simple loss function was used as a baseline. Models were scores on peak signal to noise ratio and structural similarity metrics. As one would expect, both quantum and classical models performed better with more training samples. Initially, the random GANs (both quantum and classical) scored higher but as the sample space increased, the pretrained models overtook the random models in the metric scoring. The authors believe this is because the pretrained models need more data to fine tune their existing parameters. Both classical and quantum models are competitive with each other but at the highest sample rates, the quantum models perform the best. Finally, the non-GAN classical model performed the worst and was not competitive with any model which demonstrated the utility of the GAN.

In addition to the ghost imaging test discussed above, tests were conducted with category transfer, colorization, inpainting, and super-resolution. Quantum performed better than classical models in colorization and inpainting while being competitive in the other tasks.

The authors attribute the better performance of the quantum models in these tasks to the higher dimensionality created by quantum encoding of data.

Conclusion

The recent papers show that what was simply theoretical ten to fifteen years ago is possible to create and test on quantum hardware. The hardware environment of today’s quantum systems are still limiting to the overall field as experiments were all scaled down to function within current systems operating requirements. For true advancement, quantum error correction will need to improve circuits enough that they can run quantum RAM (qRAM) that several papers reference.

References:

1. Chen, HY., Chang, YJ., Liao, SW. *et al.* Deep Q-learning with hybrid quantum neural network on solving maze problems. *Quantum Mach. Intell.* **6**, 2 (2024). <https://doi-org.libweb.lib.utsa.edu/10.1007/s42484-023-00137-w>
2. Hdaib, M., Rajasegarar, S. & Pan, L. Quantum deep learning-based anomaly detection for enhanced network security. *Quantum Mach. Intell.* **6**, 26 (2024). <https://doi-org.libweb.lib.utsa.edu/10.1007/s42484-024-00163-2>
3. Jenber Belay, A., Walle, Y.M. & Haile, M.B. Deep Ensemble learning and quantum machine learning approach for Alzheimer’s disease detection. *Sci Rep* **14**, 14196 (2024). <https://doi-org.libweb.lib.utsa.edu/10.1038/s41598-024-61452-1>
4. Preskill, John, Quantum Computing in the NISQ era and beyond detection. Quantum 2, 79 (2018). <https://quantum-journal.org/papers/q-2018-08-06-79/>
5. Giuseppe Buonaiuto, Raffaele Guarasci, and Massimo Esposito. 2024. Quantum Transfer Learning for Sentiment Analysis: an experiment on an Italian corpus. In Proceedings of the 2024 Workshop on Quantum Search and Information Retrieval (QUASAR '24). Association for Computing Machinery, New York, NY, USA, 25–30. <https://doi-org.libweb.lib.utsa.edu/10.1145/3660318.3660325>
6. Asitha Kottahachchi Kankanamge Don and Ibrahim Khalil. 2025. Q-SupCon: Quantum-Enhanced Supervised Contrastive Learning Architecture within the Representation Learning Framework. ACM Transactions on Quantum Computing 6, 1, Article 6 (March 2025), 24 pages. <https://doi-org.libweb.lib.utsa.edu/10.1145/3660647>
7. Sinha, A., Macaluso, A. & Klusch, M. Nav-Q: quantum deep reinforcement learning for collision-free navigation of self-driving cars. *Quantum Mach. Intell.* **7**, 19 (2025). <https://doi-org.libweb.lib.utsa.edu/10.1007/s42484-024-00226-4>
8. Aram W. Harrow, Avinatan Hassidim, and Seth Lloyd, Quantum Algorithm for Linear Systems of Equations. Phys. Rev. Lett. 103, 150502 – Published 7 October, 2009. <https://doi-org.libweb.lib.utsa.edu/10.1103/PhysRevLett.103.150502>
9. Xiao, T., Zhai, X., Huang, J. *et al.* Quantum deep generative prior with programmable quantum circuits. *Commun Phys* **7**, 276 (2024). https://doi-org.libweb.lib.utsa.edu/10.1038/s42005-024-01765-9
10. Yanan Li, Zhimin Wang, Rongbing Han, Shangshang Shi, Jiaxin Li, Ruimin Shang, Haiyong Zheng, Guoqiang Zhong, Yongjian Gu, Quantum recurrent neural networks for sequential learning, Neural Networks, Volume 166, Pages 148-161, (2023) <https://doi.org/10.1016/j.neunet.2023.07.003>
11. Masahide Sasaki and Alberto Carlini, Quantum learning and universal quantum matching machine, Phys. Rev. A 66, 022303 – Published 6 August, 2002. <https://doi-org.libweb.lib.utsa.edu/10.1103/PhysRevA.66.022303>
12. Lloyd, S., Mohseni, M. & Rebentrost, P. Quantum principal component analysis. *Nature Phys* **10**, 631–633 (2014). <https://doi.org/10.1038/nphys3029>
13. Patrick Rebentrost, Masoud Mohseni, and Seth Lloyd, Quantum Support Vector Machine for Big Data Classification, Phys. Rev. Lett. 113, 130503 – Published 25 September, 2014. <https://doi-org.libweb.lib.utsa.edu/10.1103/PhysRevLett.113.130503>
14. Iris Cong, Soonwon Choi, Mikhail D. Lukin, Quantum Convolutional Neural Networks, Nature Physics (2019). <https://doi.org/10.1038/s41567-019-0648-8>
15. Jonathan Allcock, Chang-Yu Hsieh, Iordanis Kerenidis, and Shengyu Zhang. 2020. Quantum Algorithms for Feedforward Neural Networks. ACM Transactions on Quantum Computing 1, 1, Article 6 (December 2020), 24 pages. <https://doi.org/10.1145/3411466>
16. Marcello Benedetti, Erika Lloyd, Stefan Sack and Mattia Fiorentini, Parameterized quantum circuits as machine learning models, Quantum Science and Technology, Volume 4, Number 4, Published 13 November 2019. <https://doi.org/10.1088/2058-9565/ab4eb5>
17. Jerbi, S., Gyurik, C., Marshall, S.C. *et al.* Shadows of quantum machine learning. *Nat Commun* **15**, 5676 (2024). <https://doi-org.libweb.lib.utsa.edu/10.1038/s41467-024-49877-8>
18. Sha, M., Rahamathulla, M. Quantum deep learning in Parkinson’s disease prediction using hybrid quantum–classical convolution neural network. *Quantum Inf Process* **23**, 383 (2024). <https://doi-org.libweb.lib.utsa.edu/10.1007/s11128-024-04588-3>
19. Hu, F., Khan, S.A., Bronn, N.T. *et al.* Overcoming the coherence time barrier in quantum machine learning on temporal data. *Nat Commun* **15**, 7491 (2024). <https://doi.org/10.1038/s41467-024-51162-7>
20. Alona Sakhnenko, Julian Sikora, and Jeanette Lorenz. 2024. Buildung Continuous Quantum-Classical Bayesian Neural Networks for a Classical Clinical Dataset. In Proceedings of Recent Advances in Quantum Computing and Technology (ReAQCT '24). Association for Computing Machinery, New York, NY, USA, 62–72. <https://doi-org.libweb.lib.utsa.edu/10.1145/3665870.3665872>
21. Archisman Ghosh and Swaroop Ghosh. 2024. The Quantum Imitation Game: Reverse Engineering of Quantum Machine Learning Models. In Proceedings of the 2024 Workshop on Attacks and Solutions in Hardware Security (ASHES '24). Association for Computing Machinery, New York, NY, USA, 48–57. <https://doi-org.libweb.lib.utsa.edu/10.1145/3689939.3695783>
22. Rosa-Remedios, C., Caballero-Gil, P. Optimizing quantum machine learning for proactive cybersecurity. *Optim Eng* (2024). <https://doi-org.libweb.lib.utsa.edu/10.1007/s11081-024-09934-z>
23. Zoufal, C., Lucchi, A. & Woerner, S. Quantum Generative Adversarial Networks for learning and loading random distributions. *npj Quantum Inf* **5**, 103 (2019). <https://doi.org/10.1038/s41534-019-0223-2>
24. Hoch, F., Caruccio, E., Rodari, G. *et al.* Quantum machine learning with Adaptive Boson Sampling via post-selection. *Nat Commun* **16**, 902 (2025). <https://doi-org.libweb.lib.utsa.edu/10.1038/s41467-025-55877-z>
25. Shinde, U.U., Bandaru, R. Re-locative guided search optimized self-sparse attention enabled deep learning decoder for quantum error correction. *Sci Rep* **15**, 3615 (2025). <https://doi.org/10.1038/s41598-025-87782-2>
26. Liu, Y. Superconducting quantum computing optimization based on multi-objective deep reinforcement learning. *Sci Rep* **15**, 3828 (2025). <https://doi.org/10.1038/s41598-024-73456-y>
27. Huang, Y., Che, L., Wei, C. *et al.* Direct entanglement detection of quantum systems using machine learning. *npj Quantum Inf* **11**, 29 (2025). <https://doi.org/10.1038/s41534-025-00970-w>
28. Choi, S., Salamin, Y., Roques-Carmes, C. *et al.* Photonic probabilistic machine learning using quantum vacuum noise. *Nat Commun* **15**, 7760 (2024). https://doi-org.libweb.lib.utsa.edu/10.1038/s41467-024-51509-0

Appendix A – Classification of Papers

Primary Sources

Secondary Sources

Foundational Reference Sources