Appendix B – Annotated Bibliography

Date	Paper Title
6 August, 2002	**Quantum learning and universal quantum matching machine**
	Early work examines a 'matching' problem where an input is evaluated to determine what state it most closely matches
	This is done to demostrate that quantum computers can perform maching learning, aka learning a model from training data and then matching unseen information
	The problem uses three different quantum states that are pure qunatum states with an unknown phase shift applied to them
	The template (training data) state is the tensor product of the three pure states with unknown phase shift applies
	For simplicity, the states have just two classifications, but they could be any number of states.
	Two solutions are offered, a semi-classical Bayesian classifier and a universal quantum matching machine
	For the semi-classical, strategy is to use state estimation on the templates, then use results to create classifier, then measure and apply classification
	The universal quantum strategy attempts to learn the state without the initial state estimation from the semi-classical method
	This is done by utilizing the probability distributions of the input features. The probability is used to create a score for each state that will be maximized
	The task is to find the positive eigenvalues resulting from comparing the template data to the new state data
	This information reveals how similar the new state is to one of the classifier states
	Paper also considers difference between only having a single copy of the new state and the theoretical infinite number of copies

	This is important because quantum computing has a 'no cloning rule' and states must be constructed, not copied. It may not be possible to create multiple copies of test data
	Understandably, scores were improved on multiple copies of the data rather than just a single instance
7 October, 2009	**Quantum Algorithm for Linear Systems of Equations**
	The study authors show that Quantum computing can be used to create a solvable system of linear equations.
	Given a hermitian matrix A and vector b, want to find a vector x such that Ax= b
	vector b can be encoded on quantum state |b>, then hamiltonian of A can be applied to |b> which is a phase transformation
	|b> can be decomposed into eigenvalues, then normalized
	If eigenvalues are too far apart, A cannot be inverted. ratio between smallest and largerst eigenvalues is k. singular values of A need to be between 1/k and 1.
	System results in vector x. However, in quantum you cannot read all values of x unless you run the process n times for the n values in vector x.
	Better approach is to use expectation value x^TMx where M is a linear operator.
	From M, features of vector x can be extracted.
	Why does this matter? A system of linear equations in machine learning is several equations that use the same variables and can be solved to find the values of the variables.
	Having a system of linear equations in a quantum system makes it is possible to minimize or optimize these given variables.
	Additionally, an exponential speedup of solve the SLE occurs compared to the classical SLE
	Overall runtime is O((log N)k^2s^2/e) where s is the number of values in a row of the spare matrix and e is error

	Clearly, if some constants such as the eigenvalue ratio k is large or the matrix is not sparse then the runtime will suffer, but poly O(log N) is a speedup over the linear N of the regular SLE
01 March 2014	**Quantum decision tree classifier**
	A decision tree classifier is a ML method for supervised learning where regression is used to split data into subgroups
	decision tree needs to create optimal tree splitting decisions with minimal error and then using the tree to accurately predict class of unseen objects
	The paper adapts classical decision tree algorthm into quantum computing, authors are designing a classifer that handles muliple class labels
	Separately, there is a quantum oracle (referred to as a quantum query) algorithm that returns values based on a predetermined function. The function does not learn and is not considered a decision tree
	Splitting a decision tree requires a metric to decide, here the quantum entropy or Von Neumann entropy is calculated and used.
	A quantum entropy of zero indicates a pure node which is the ideal splitting state.
	Quantum mechanics has a no cloning rule which does not allow data to be copied. Everytime a calculation is needed, a specific superposition must be created.
	For the decision tree classifier, the paper's authors are not strictly adhering to this. They are allowing multiple copies of training and test data to exist. The number of required copies is not calculated
	The Node Splitting algorithm takes the quantum entropy value of a random node and searches other nodes for a lower value. If none is found, the node with the lowest entropy is returned.
	The node splitting algorithm uses Grover's Algorithm for this process. Grover's is actually used in several of these decision tree classifier algorithms.
	Grover's uses a quantum oracle to differentiate between inputs and them amplifies their amplitudes to increase the probability of identifying the desired state.

	In order to partition data using attributes, the data distribution is predetermined. It is then sorted classically, as a quantum sorting is more expensive. O(n log n) classical vs O(2/3n^3/2) quantum
	Next the fidelity of the quantum state of the data distribution is measured. The fidelity measurement is the measurement of the closeness of the angle between quantum states.
	The data is then clustered by first comparing the fidelity of all pairs of states and creating a cluster for each found state. For all points grouped together, a centroid is calculated.
	The second step of the clustering algorithm is to take all points that were not assigned to a group and assign them to the group with the nearest centroid.
	Now that groups have been created, the decision tree may be constructed by selecting groups and calculating quantum entropy. The split occurs where the entropy is smallest
	Final output of the quantum decision tree is a sorted list by fidelity. Searching list can be done with Grover's algorithm
	Authors point out that Quantum model still lacks many features of classical decision trees (eg node pruning, training data with quantum noise)
	This work is entirely theoretical and was not tested on quantum hardware. The runtime of the algorithm benefits from the speedup of Grover's as it is used frequently
	That said, the state of quantum technology in 2025 is not such that this can be put into practice.
	To elaorate, The state of NISQ hardware is such that there would need to be tomography used to keep the data in an intermediate state or run into decoherence issues.
	The authors also acknowledged that they didn't look into errors which seem likely given the complexity of the algorithms versus the limited circuit depth that can work on today's hardware.
27 July 2014	**Quantum principal component analysis**
	Classical PCA is a technique that identifies variance within data by mapping it onto a lower dimension space

	Quantum Tomography is a method of measuring a quantum state to discover features of the quantum state.
	Tomography works with multiple copies of the quantum state to draw necessary conclusions
	However, instead of treating the state in tomography as a passive thing, we can convert it into an energy operator or Hamiltonian
	The quantum state or density matrix is used to calculate eigenvalues and eigenvectors
	Makes it possible to construct the large eigenvectors and eigenvalues of the state which are the principal compontents. Thus, quantum PCA has been performed.
	Would need one density matrix/copy of the state per dimension of PCA analysis that you are interested in.
	PCA process runs faster with a matrix that is well represented by just a few principle components. Holds true in both classical and quantum
	This is a roundabout way of saying that it works best with low rank matrices.
	The principal compnent matrix can be calculated in O(log n) when the data in the matrix is sparse.
	The quantum hardware can store the coefficents of the vectors in O(d) hardware/qubits where d is the dimensionality that we have reduced the analysis to.
	Further any matrix can be exponentiated in O(log d) time with this method. Authors need access to vectors A in AA^T to make this happen.
	Adding in another term, quantum PCA runs in O(R log d) time while classical PCA runs in polynomial (quadratic) time. This is the quantum speedup of quantum PCA.
	Paper is proving the theory behind the calculations. It did not run any simulations
	Quantum PCA still limited to a fraction of information available in classical PCA because of challenges in getting large amounts of classical data into quantum systems
25 September, 2014	**Quantum Support Vector Machine for Big Data Classification**

	Classical SVM is a supervised ML technique, splitting data into groups based on the best linear fit
	Quantum SVM implemented with O(log NM) for both training and running time where M is data points and N is features
	Utlizes least squares as SVM solving technique for solving with quantum phase estimation
	The quantum SVM process uses the quantum PCA matrix exponentiation method to speed up calculations
	For the process of translating the kernel matrix in SVM to quantum computing, the authors propose an oracle to return the kernel calculation
	An oracle is a 'black box' type of function that can receive a query and give an output but otherwise has no parameters for tuning.
	Speedup is achived through inner product evaluation of matrices, calulated in O(log N) in quantum environment, but polynomial time classically.
	The algorithm works by creating a quantum state that describes the hyperplane using the matrix inversion algorithm and the classifies a given quantum state vector
	When trained, the quantum SVM can act as a new computational basis for testing, with the basis aligned with the training labels.
	For classification, construct a state with equal probability of the test vector and the basis state, then measure the ancilla of state to construct success probability P
	If $P < 0.5$ then test vector is classified as +1, otherwise it is -1
	Quantum SVM may also provide some measure of privacy as the data from calculations doesn't have to be fully represented, only inner product
	Also, once the kernel matrix is generated the data is hidden from the user.
	As with quantum PCA, this work is done as a proof and is not implemented. It may be impossible to implement on today's NISQ hardware

	Similarly, it's good to recognize that there may be some privacy benefits, however, given that you cannot store quantum information for any period of time, that benefit may be short lived.
30 July 2018	**Quantum Computing in the NISQ era and beyond**
	Although it mentions QML/deep learning, this is not specifically about quantum machine learning. It does discuss what's feasible in quantum research given 2018 technology
	This work matters because machine learning algorithms have to run within the environment that was detailed here. Although developments have happened over last 7 years, framework remains relevant.
	Looks at the entanglement frontier, largely meaning that we're devloping tools (quantum hardware) to explore quantum systems.
	Utility of quantum devices will come from quantum complexity (gain from using superposition) and quantum error correction (ability of hardware to maintain accurate state)
	Potential of quantum algorithms? Possibly solving problems difficult for classical computers. Primarily Shor's factoring but not a lot else as of yet
	More potential-- Complexity allows speedup because superposition allows multiple problems to be factored at once
	More potential-- We can't (fully) model quantum systems on classical systems. That should be reason to be optomistic about potential
	Unlikely to solve classically NP-hard problems, most likely to speed up exhaustive search of them. May also improve on approximation algorithms of NP-hard problems
	Quantum challenges- systems need quantum particles isolated from other things to work, but must be connected externally to get input and send results.
	Error Correction necessary to scale up quantum computers from current state to more realiable state

	Current hardware state term coined NISQ - Noisy Intermediate-Scale Quantum- Noisy means control of qubits imprecise, intermediate scale means 50 to a few hundred qubits
	Best technologies for qubits are trapped ion and superconducting processors. Error rates still high on these 'best' technologies
	Notes that even though quantum computers are improving, classical are as well so quantum advantage over classical may continue to be a moving target
	Quantum annealing alternate technology with lots of qubits and low noise, however, untested at scale.
	On the machine learning topic, as of 2018 it was mostly theretical, needed more testing. Aside from purely focusing on speed ups, QML research may yield improvements in classical algorithms too. TBD.
	There isn't an easy way to move large amount of data from classical to quantum so encoding large amounts of date in qRAM/quantum registers has high overhead.
	Mentions HHL System of Linear Algebra article cited previously, calls it BQP-complete because any problem solved efficiently on a quantum computer can be encoded as a matrix inversion problem.
	Although nearterm improvements to hardware are expected, it is extremely unlikely nearterm that things will scale to the point that large circuits are feasible on quantum computers (correct for 7 years later)
26 August 2019	**Quantum convolutional neural networks**
	Given advances in classical ML, want to see if these can be translated to quantum computing
	Two quantum problems have the potential for investigation, Quantum Error Correction, applying optimal code given an unknown error model
	and also Quantum Phase Recognition, determining if an input quantum state belong's to a particular quantum phase.
	These are hard to accomplish without resorting to the difficult/expensive process of quantum tomography

	Additionally, state of quantum hardware limits depth / complexity of quantum algorithms
	Quick classical CNN summary- Convolution layers create feature map, pooling layers condense map to smaller grid, final layer is classification with some activation function
	Hyperparameters like number of layers, size of weight matrix are usually fixed for a given CNN
	Translating this to quantum, the convolution layer is where a single unitary operator is applied
	The for a pooling layer, some of the qubits are measured and the result specifies how much rotation is applied to nearby qubits
	quantum convolution and pooling are repeated until size is sufficiently small. For a fully connected layer, another unitary operator is applied.
	Final output is obtained by measuring a few qubits. The hyperparameters (num of layers) are fixed, the unitary operators are learned.
	One practical application of this is a multiscale entanglement renormalization ansatz (MERA).
	MERA takes an input state and adds a new qubit to and perfoms a unitary operation on all of the data.
	As MERA is adding qubits and the CNN convolution / pooling layers are shrinking the number of qubits, a circuit can be created where MERA runs in one direction and CNN the other.
	Athough the logical layout is symmetrical in the reverse, not every CNN step has to shrink / remove qubits although the MERA process does always add qubits.
	What can be done with the extra unneeded qubits? They can be read as part of error correction where the quantum error correction does syndrome detection
	It's at this point where we can be sure that this is a simulated environment. As of 2019 there was not a functional QEC model where it performed better than the physical error rate

	Even though it is simulated, a reasonable (for state of NISQ hardware) two layer QCNN with N=9 qubits was modeled. This system had 126 parameters.
	This model was used in simulations where x-axis, y-axis, and z-axis errors were correlated and uncorrelated. It was then compared to Shor's QEC error correction
	Shor's QEC assumes no correlation between errors and the QCNN performed the same when there was no error correlation.
	QCNN performed better than Shor's QEC when errors were correlated, however, there are no studies that suggest quantum errors are correlated. That doesn't prove they aren't, of course.
02 October 2020	**Quantum Algorithms for Feedforward Neural Networks**
	Starts with overview of classical neural networks, mentions importance of back propagation. Notes that neural networks require significant resources to train and run.
	Quantum challenges: neural network computations are very sequential, quantum computing is very parallel. Quantum computing also challenged by intermediate step calculations.
	Further: Strength of neural networks comes from non-linearity. Quantum relies heavily on linear algebra. Quantum requires many extra steps to handle non-linear calculations.
	If training a NN on quantum computer, parameters and training data must be encoded into quantum states. State preparation can be a bottleneck for quantum computers
	Not all bad. classical NNs use a lot of linear algebra despite non-linearity. Randomness is common in quantum computing and many classical algoritms make use of some randomness as they are optimized.
	Solution for non-linearity is to store vector inner product data in qRAM instead of quantum states. non-linear calculations can be performed on qRAM.
	Some things like a weight matrix is too large to be stored in qRAM, quantum states can be reconstructed indirectly
	Side note: There are several "Quantum Inspired" algorithms that have been developed to run on classical computers featuring "only" a polynomial slowdown in running time. Paper dryly notes how that might matter.

	Proposed quantum feed forward algorithm has not been proven to work better/faster than classical counterparts.
	Quantum training part of the process is stated to have polylogarithmic time. Paper also states that logarithmic term dominated by other factors, effectively is O(((TM)^1.5)N) where
	T is number of update iterations, M is size of minibatch, and N is total number of neurons in network.
	Quantum evaluation also has a logarithmic term that authors state is minimal leaving us O(N) where N is still the count of neurons.
	Contrast this to classical where training is O(TME) {new term E is edges} and evaluation is O(E). Paper also notes that classical NN have lots of edges.
	paper summarizes classical CNN calculations then contrasts with quantum where the inner products in forward and backward propagation steps are estimated (within an error tolerance) rather than calculated
	Returning to discussion of quantum training, it is a hybrid classical - quantum setup that only uses quantum to estimate inner products and to move data in and out of qRAM
	The paper discusses qRAM in a different way than most quantum papers which usually describe it as a quantum register or set of qubits. Here, the data is stored in a binary search tree.
	To expand, the paper assumes the qRAM is fully error corrected and notes that this is far off from what is available on today's NISQ hardware. Fault tolerant quantum computing would be a huge milestone.
	A presentation of the algorithms for forward and backward propagation are presented. They include the inner product estimation. The algorithms are then simulated classically.
	Because they are simulated classicaly, we do not get a speed up, hence the earlier side note. The quantum NN performs better than classical on the IRIS dataset when the error is 0.1. Much worse for other error values.

	The error estimation is compared to classical algorithms that introduce noise to effectively combat overfitting.
	Paper is very clear that classical algorithms outperform this method currently. The authors believe the inner product estimation where it effectively fights overfitting without additional steps is promising for future work.
	Finally, quantum feedforward neural networks are presented as theoretical, with the suggestion that the quantum inspired version is helpful for designing future algorithms.
07 December 2021	**Quantum deep reinforcement learning for clinical decision support in oncology: application to adaptive radiotherapy**
	Heathcare background: Radiation Therapy is a common cancer treatment, ~29% of cancer survivors in 2016 received some form of RT
	Radiation Therapy sessions (in terms of radiation dosages) are similar for patients with the same stage of cancer
	It is also known that patients do not all respond the same way to radiation therapy as a result of biological variation
	Therefore, a one size fits all approach to radiation will be suboptimal, personalizing radiation dosages is expected to imrpove outcomes
	Study examines treatment of lung cancer patients, they will all be treated the same for four of the six weeks of therapy with a custom plan for weeks 5 and 6.
	Reinforcement learning is a machine learning technique where the model is rewarded for making positive decisions. Tuning occurs over several iterations
	Reinforcement learning is here applied to radiation dosages using a Bayesian Network for feature selection
	Approach also incorporates prior knowledge into the process in addition to patient dosage information
	Deep-q reinforcement algorithm is used, specific configuration of deep-q for cancer treatment not discussed in paper.

	Output from Deep-q is encoded into quantum states. Highest state value selected and amplified, resulting in dosage measurement.
	Dosage is then simulated (rather than being used in actual practice) in reinforcement learning model which predicts the patients next state
	Quantum aspect of the model is designed to simulate the uncertainty in decision making process
	Authors state that their study is similar to a 2020 study using Quantum Reinforcement Learning involving gambling decisions. Main difference is Deep-q algorithm
	Compared 3 models, plain quantum reinforcement learning, quantum deep-q reinforcement learning simulated in qiskit and results run on actual IBM quantum circuits
	Had AI decision making, rewarded positive outcomes which were specific local tumor control and no raditaion induced pneumonitis.
	AI always selected highest score in these areas effectively making them confidence scores
	This was not supervised learning, so authors created a similarity score to compare typical practices and a self evaluation score to compare model to actual clinical decisions found in patient data
	When comparing model recommendations to actual recommendations, authors had to label some outcomes as 'unclear'
	This is due to the fact that patients experiencing pneumonitis in the actual real life results may or may not have been able to tolerate higher dosages simulated and recommended by model
	Quantum methods showed up to 10% improvement on clinical decisions
	Also, the simulator and the actual IBM quantum hardware performed differently. Authors had to design the circuit differently to avoid decoherence
	Authors also mention that they could not use the true Grover's quantum amplification method on the IBM hardware becuase of circuit depth/decoherence issues

10 January 2022	**Parameter estimation in quantum sensing based on deep reinforcement learning**
	Physical quantum systems require regular upkeep to maintain functionality. Systems make readings using sensors that must be regularly calibrated.
	Paper investigates the process of qunatum sensing and the methods to find the best system parameters
	Background: Sensing follows general workflow: 1) Prepare probe state, 2) Apply quantum transformation [Hamiltonian or Gate] 3) Measure new state
	Convential methods for determing sensing parameters are GRAPE and CRAB which are both sensitive to noise and are reportedly hard to engineer
	Machine learning has been previously used to create optimal parameters for time independent processes
	Authors are exploring whether some techniques can also be applied in time dependent setups (meaning where decoherence of data is an issue)
	The specific time dependent use case is time-dependent Hamiltonian evolution, where an unknown parameter influences strength of Hamiltonian
	Sensing is key to quickly determining unknown parameter
	In noiseless environment, this unknown parameter has an upper bound and reinforcement learning is used to treat the difference between upper bound and actual as reward term.
	There is an agent which feeds improved terms into the system to calculate the system parameters
	A classical deep reinforcement process' output is used to create a quantum ansatz used in an QVE / eigensolver algorithm
	The output creates better possible eigenvalues for the problem
	This is slightly different than the idea of using quantum computers to speedup or otherwise improve on the performance of classical algorithms. It's using classical algorithms to improve quantum computing.

10 May 2022	**Quantum–Classical Image Processing for Scene Classification**
	Image augmentation performed by a quantum computer.
	Satellite images are reviewed and enhanced through a hybrid ml and quantum model
	Only needed 4 qubits for this model which is good for NISQ hardware
	The research from this paper was used in the Q-SupCon model to augment image sets and increase the accuracy of classifications
05 January 2023	**Quantum Fourier Convolutional Network**
	Paper notes that convolutions and quantum have similarities at a surface level (both perform matrix operations in high dimension vector space) and look to see if quantum advantages can be applied to CNNs
	Direct convolution of a quantum state is not possible, however, authors propose hybrid approach as a workaround
	Training time in deep learning is a burden, anything that can reduce overall training time (such as speeding up computations) is worthy of exploration
	Replaces the discrete Fourier transform in CNN with quantum Fourier transform to achieve speedup over classical
	Simulated models (again, not actual) run faster with quantum CNN over classical
	In the construction of the quantum fourier transform, authors discuss 'qRAM' which appears to be a quantum register set aside for a quantum state to be saved. It's more like loaded and briefly held than truly 'saved'.
	In terms of background, qRAM which is also sometimes called a quantum register, is a series of physical qubits can be organized into a logical space.
	The overall process has the kernel being applied to one register and the input vectors (from perhaps an image) being applied to another register.
	The registers are then multiplied togetherin such a way that the result of the operation is the inverse quantum fourier. It is then sent to back to a classical computer to continue classical CNN computation

	The quantum foruier transform takes the same inputs as theclassical discrete fourier transform
	Needs n(n+1)/2 gates to perform QFT on n qubits. Quadratic can scale up quickly. 10 gates for n=4, 36 gates for n=8.
	Time complexity improves from O(n2^n) for classical to O(n^2) QFT.
	Backpropagation is a challenge in quantum circuits. Because of the parameterized circuits, gradients can be calculated by running cicuit again with shifted parameters
	Ran the QFCN as a simulation on the MNIST dataset. Noise was simulated as well to make it similar to real life quantum hardware
	In the simulations, the QFCN compares well with a control non-fourier Quantum CNN model constructed for comparison
	Authors show that it is possible to take advantage of quantum speedup in CNN calculations.
7 Feb 2023	**Quantum Recurrent Neural Networks for Sequential Learning**
	Authors want to design a Quantum Recurrent Neural Network that's realistic for the NISQ hardware environment that exists in today's quantum world.
	An approach to the quantum circuit where all QRNN circuits are run one after another, modeling the classical RNN directly will unfortunately have decoherence problems on today's hardware.
	The paper works around this issue by proposing a staggered QRNN circuit.
	It's helpful to review what a classical RNN is. It is similar to CNN, however R is for recurrent and is used for sequntial/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 output.
	That last layer output in the classical RNN implicitly contains output from all of the previous layers because the function works recursively in that manner.

	A common bottleneck for classical to 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, downside is that quantum circuits grow exponentially with datapoints
	Angle encoding of classical data takes the datapoints and represents them as angle rotations around the computational basis. Qubits are just O(n) of classical datapoints. This is more efficient.
	Because more qubits lead to more errors in an NISQ environment, angle encoding is used on data.
	Authors use three qubits for a single classical bit, intentionally redundant because of error issues with NISQ circuits
	An ansatz, which is a parameterized quantum circuit that has adjustable gates is used to perform the work of the QNN
	The values of the gates in the ansatz are what is trained by the model.
	QRNN ansatz must be implemented in a hardware efficient way using single qubit rotation gates and two qubit controlled rotation gates which can be decomposed into CNOT and Rz gates
	For measurement, paper has a reminder about quantum measurement, namely that upon measurement state collapses to one or zero in computational basis, must rerun multiple times
	Multiple runs will give us an estimate of expectation.
	In proposed model, first qubit of register is measured, then converted to prediction term yt
	Paper demonstates what closed system QRNN would look like, with a quantum register that takes a new datapoint and a second register that holds the ongoing value of the variables

	While quantum register with new data is reset prior to each input, the other register is not reset during the process. It has to hold the state for many iterations, which is unlikely to hold in an NISQ open system
	In the proposed staggered model, only two of the three qubits holding the state must remain for the next iterations, paired with one of the qubits that was previously used for new data.
	Each time a new datapoint is processed, the two registers shift by one qubit each. This means the state registers are all reset every 3 iterations, making that the longest time that a qubit must maintain coherence.
	Just as in classical deep learning, predictions are optimized versus real values. The authors used least squares
	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 qunatum computer, it requires running the quantum circuit twice with slightly different values and taking the difference between the loss functions
	Additional derivation steps must be computed on a classical computer.
	Performance of two QRNN systems (called plain and staggered) are then compared to a classical RNN that is configured similarly to the QRNN setup.
	The circuit was based on seven days of input data then used to generate a prediction. The RNN model was the same for comparison, the authors did not discuss if this was the optimal RNN configuration
	The two quantum RNN models performed similarly to the RNN on actual data and were even more accurate for a few features such as wind speed in meterolgical data.
	Having larger redundant registers of qubits was also studied. Prediction accuracy increased as quantum registers grew from 4 to 6 to 8 qubits. Again, the registers are holding redundant information
1 November 2023	**Quantum Computing and Visualization: A Disruptive Technological Change Ahead**
	Excellent and compact summary of quantum computing

	Article is about improving Bloch sphere represenetation by means of a QC sphere that shows weighted probabilities of superpositions
	Useful for introductory information, the rest of the content is not relevant to quantum machine learning.
11 December 2023	**A survey on the complexity of learning quantum states**
	The survey examines multiple different quantum computing problems and recent advances
	Learning here is on the broad side for quantum machine learning, for example, one section examines whether you can learn the physical state of qubits
	There is more discussion on advanced topics such as learning about classical functions encoded into models
	May want to examine some of the cited articles for further research.
08 January 2024	**Deep Q-learning with hybrid quantum neural network on solving maze problems**
	One option is to create a quantum walk algorithm and a variational quantum circuit (VQC) to encode classical agents with state information. Second option is to use a fully quantum VQC.
	Fully quantum model promising, but hampered by state of NISQ hardware
	Briefly discusses classical reinforcement learning using a reward signal, specifically Markov decision making process (MDP) investigated
	In MDP, agent observes state, then selects 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.
	quantum approach uses a 'model free' method of estimating state-action pairs and then using that estimate to update the value with a paramaterized (VQC).
	The model free method is referred to as Q-learning as the model estimates a Q-value or cumulative reward
	deep q-learning is approximating the q-value/q-function with a neural network

	Experiement constructed for quantum neural network to search 4x4 maze
	Hybrid quantum / classical environment because full quantum not yet practical thanks to noise and scalability issues
	classical agent updates quantum learning table by estimating quantum states and storing them in table
	The agent receives updates from deep neural network, the variational quantum circuit is used as the last layer of the DNN
	To prevent correlation between samples, the agent has past transitions in a table and a random past experience is chosen each time algorithm runs
	Loss function is Q-Loss, which MSE between predicted and target Q-values. For each step the agent caculates new gradients and then updates neural network weights.
	Circuit design uses two qubits which are manipulated by feature encoding and ansatz module to return a result
	Researchers ran the QNN in a simulated qiskit environment and compared results to traditional CNN
	Traditional CNN had a better win rate in 4x4 maze (89% vs 85%) and a much shorter training time (only 1/8th of qNN training).
	qNN improved on 5x5 maze but still was inferior to traditional CNN.
	Researchers suggest that implementing CudaQ for quantum GPU could improve process. Future work oppotunity.
	Although performance strictly worse than classical, study shows that the QNN is feasible, particularly if it can be effictive implemented on real quantum circuits
02 May 2024	**Quantum deep learning-based anomaly detection for enhanced network security**
	web security involves monitoring traffic, finding anomolies and mitigating them.
	Quantum machine learning has not commonly been used for this application. Want to explore quantum's ability to process correlated features

	Proposed frameworks explored include quantum auto encoding plus a one layer SVM, kNN, or Random Forest.
	Anomaly detection has challenges including class imbalance, diverse types of anomalies, and unexpected abrupt behaviors
	Previous work in QML anomaly detection used supervised learning. Here attempting unsupervised learning.
	Autoencoder are used to compress state details while maintaining key information. Noted that it is similar to PCA ML technique.
	The goal of the encoder in this context is to compress quantum information so it can be represented classically.
	Quantum qubits represent 2^n classical bits of information, explaining why it would be necessary to compress information to keep it managable on a classical system
	A six qubit quantum encoder/decoder is presented. When encoded, the full quantum information is conveyed on half (three) of the qubits.
	Authors expect the auto encoder to learn typical patterns so it may detect anomalies
	Using the encoder on its own to detect anomalies soley based on how much error was in the reconstruction of the quantum state was examined and discarded.
	For the proposed model, the classical data is first run through PCA to get it down to a few key dimensions represented by qubits. The expirements used 4 qubits for this
	After the classical PCA, then the quantum autoencoder is used and then one of the three models mentioned earler are applied. The quantum autoencoder is doing more compression from the classical input
	With the one layer SVM, both a fully quantum SVM and a classical SVM were examined.
	With the random forest approach, the input is fed through several variational quantum classifer (VQC) models. The VQC models are trained and a majority vote of the models is taken.
	For kNN, the quantum encoded data of the test state is compared to the nearest neighbors of M training states. Fidelity (similarity) is the metric used in kNN

	The article discusses different methods of constructing the SVM quantum circuit, both methods need 2^n gates per feature which can quickly become large
	For random forest, a feature map circuit and an ansatz circuit are designed.
	Test are performed on several different anomaly detection datasets. It's noted that as a succes metric, accuracy is not useful because of class imbalance in anomaly datasets. F1 score used for evaluation
	KNN method performed the best of proposed frameworks. Authors believe that it is because of the distance/fidelity metric used for anomaly detection.
	Expirements were simulated using pytorch and qiskit
	Tests were performed on actual quantum hardware, notable because many QML just do simulation. Still won't be replacing classical ML soon.
07 June 2024	**Study on The Effect of Encoding Method in Quantum Machine Learning**
	Encoding is important to consider in quantum computing because of all the data that must be encoded from classical to quantum
	Method of encoding data is important because of the quantity of data ingested by models and the provides an upper limit on performance of quantum models
	Enoding options include amplitude encoding and qubit rotation
	Previous work has shown that choice of encoding method results in different decision boundaries.
	Want to quantify the results from encoding. Have come up with distribution distance and distribution radius based on geographic distance in Hilbert space
	Distance conveys how well enoded the points are and the radius shows how close they are to the mean.
	Paper examines binary classification in supervised learning where data is appropriately split into one of two classes
	The mechanics of translating cartesian points onto a Bloch sphere representation are discussed.

	Cosine distance is preferred over euclidean distance because euclidean distances become similar in high dimension spaces.
	There is a relationship between distribution distance and distribution radius. Models preform better with a larger distribution distance and smaller distribution radius. (Well defined and tightly clustered)
	There is also a relationship with accuracy. With fixed radius, accuracy increases as distance increases. With fixed distance, accuracy increases as radius decreases.
	The authors do not state one encoding method is superior to another. The framework of distribution distance and radius can be used for evaluation in any specific instance.
	Only examined a single qubit. Haven't investigated multiple entangled qubits, possible topic for future work.
11 June 2024	**Learning Shallow Quantum Circuits**
	Advantages to using quantum shallow circuits: shallow quantum circuits more powerful than classical,hard to simulate on classical, works in NISQ environment
	No previous polynomial time algorithm defined for learning shallow quantum circuits. Challenges for learning include lots of bad local mimima and the fact that common algorithms like gradient descent fail.
	Parameterizing the gates, like making an ansatz, makes it possible to train a quantum circuit and make it possible to do Quantum Machine Learning.
	Deep neural networks in quantum are challenging because of 'barren plateaus' which is really gradients that become zero and drop out/cause learning to cease
	Therefore, shallow networks should have an advantage. They still have lots of local mimima during optimization.
	Quantum tomography is resource intesive and scales poorly, however, the best case for using it is when the scale is small like what you have in shallow networks
	Proposed technique is to have qubit inversions to disentangle. Can lead to complete circuit without difficult

	Setup: Let unitary U have n qubits. Take randomized dataset with N samples. Each N sample represents an input state and a pauli output measurement.
	Theory: When circuit is over finite gate set, U can be learned with O(log n) samples and poly (n) time
	Note that polynomial time has a large exponent. There is further theory that this can be refined b/c modern hardware uses geometrically local circuits
	with that, it becomes learned in O(log n) samples in O(n log n) time and learned circuit depth of (k+1)(2d + 1)+1. k and d are dimensions of the quantum lattice
	These theories can formalize the circuits as quantum celluar automata (QCM)
	Finally, circuits can be learned with O(1) queries in O(n) time. [NB, these are tightly bound big theta]
	Does this extend to higher depth circuits? No. Must use Grover's algorithm and then will be bounded by runtime/complexity properties of Grover's
	Why is small depth important? Because by using local inversions, we're brute force checking if UV' = U' [tensor prod] I for some n-1 qubit U'.
	The larger n is the larger the search space which will become impossible to manage at large n. Hence, switching to Grover's
	Also key to the process is the locality of the gates. Swap opeations are performed for the inverse. If the gates are not local, more swaps have to be completed, leading to (much) higher error.
	Finally one step to learning circuits it to disentangle them. Can be done by checking for inversions in O(nC) time where C is the maximum number of inversions for region.
	Given that larger depth models lead to high error rates on NISQ hardware, exploring this learning complexity for shallow circuits is useful for understanding tradeoffs in circuit design
20 June 2024	**Deep Ensemble learning and quantum machine learning approach for Alzheimer's disease detection**

	Alzheimer's disease is debilatating mental illness. Study investaged use of patient MRI scans to look for Alzheimer Disease markers. This data set is available through Kaggle.
	Model used combined VGG16 and ResNet50 to extract features. Features were then fed into QSVM model which gave classifications.
	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 state of the art.
	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.
	Flattened layers from each model are concatenated and fed into QSVM. Dataset was reduced to 5 dimensions so as to only need 5 qubits in the SQVM.
	Data augmentation was also performed on the dataset to combat class imbalance and overfittng.
	The classical SVM scores were lower than 'related work' state of the art SVM. Authors state this is related to what images they trained the models on, which were not MRI scans.
	Ran through IBM Qiskit simulator, doesn't appear to have actually run code on IBM hardware. Achieves proof of concept but doesn't have quantum advantage in simulator.
	ML + QSVM/Qiskit improved on accuracy/recall precision of classical SVM algoritrm.Metrics for QSVM ensemble model over 99% so they are acceptable.
	Training of classical deep learning models were discussed, however, there was no discussion of actual training qsvm training (even in qiskit).
	Although not evaluated on actual quantum hardware, given the small scope of a quantum SVM, this could plausibly be used, depending on how difficult it is to actually implement.
06 July 2024	**Shadows of quantum machine learning**

	Discusses 'shadow' models or models trained on quantum computers and evaluated (inference step) on classical computers
	This addresses access bottleneck in quantum computing at a tradeoff of speed at evaluation time.
	Typical ML workflow is training phase on training data followed by inference phase on test data. Quantum computing is needed at both steps for QML. Access to quantum computers very limited right now.
	Shadow tomography (recreating qunatum state) can be used to create a classical model after the training is complete.
	Seek answers to the question of whether there is still an advantage in a classical but quantum trained algorithm. Also, are there quantum algorthms that can't be converted to classical
	Shadow algorithm draws from linear algebra, where the trained f(x) function is a trace of the quantum density matrix for the parameter theta (generally a Pauli rotation)
	Shadow model is a flipped version of the quantum model. Specifcally, the quantum model has an encoding state followed by a paramerized function.
	The filliped version starts with the parameterized function followed by the encoding step followed by observations. A bit string is generated following this.
	When the model is evaluated multiple times, the set of bit strings is then used as input or "advice" to the classical algorithm along with the new test input during evaluation.
	There is not a discussion of the methods of shadow tomography as other papers cover this topic. It is noted that the choice of method may force limitations on shadow function.
	Have to consider what models can be flipped? The parameterized function has to be transformed into a density matrix. Adding a qubit can address the eigenvalues that must be calculated.
	Number of qubits involved in flipping is logarithmic because Pauli observable is 2^n.

	Although classical algorithms do not have a quantum advantage, unless a trapdoor function is given. The 'advice' generated earlier serves as a key for these trapdoor functions
	Specific trapdoor function in this case is the discrete cube root. If you consider how cryptographic functions work, you will have a general idea of how the advice calculation can be a key for the cube root.
	When determining if quantum models have shadow equivalents, it is helpful to rewrite quantum model equation as a fourier series.
	For high dimensional data fourier series can suffer from exponential sample complexity needing 2^n calls to converge.
15 August 2024	**Quantum deep generative prior with programmable quantum circuits**
	Parameterized quantum circuits (PQC) are used in Quantum Machine Learning
	Because of today's NISQ hardware, it's difficult to scale the depth and width of PQC circuits
	One challenge of exploring quantum algorithms is that classical simulators struggle to fully simulate them.
	The authors take on a computer vision task with classical image generation via GAN assited by PQC
	With ghost imaging, a low resolution image is created and the quantum algorithm must reconstruct it.
	Classical CNNs capture image statistics during processing that can be treated as a prior image
	Higher dimension images can be approximated by a learnable generator G that captures latent space Z
	Proposed: Quantum Deep Generative Prior (QDGP) using PQC to generate latent space with larger dimension thanks to quantum capabilities
	Images were created by taking an original image and transforming it into a degraded version. Ghost Imaging priors is using low light (as little as a few photons)

	Reasearch on classical methods to reconstruct degraded images with minimal samples is ongoing, however, they often need large datasets for training which is a challenge
	For experiments, QDGP utilizes a pretrained BigGAN. Tests involved images chosen to be outside of trained dataset
	With small number of samples, quantum models were competitive but inferior. With larger sample numbers quantum models performed better than classical.
	For high resolution images in tests, the pretrained GAN models all suffer, but quantum still is best among method comparison.
	Also ran tests with category transfer, colorization, inpainting, and super-resolution. Quantum excelled incolorization and inpainting, was similar to classical in the other tasks
03 September 2024	**Quantum Transfer Learning for Sentiment Analysis: an experiment on an Italian corpus**
	NLP has experience tremendous recent growth thanks to deep neural networks. These are resource intensive and quantum may be an alternative
	Current NISQ hardware limits Quantum NLP to relatively easy tasks or small datasets
	Recently, a Quantum Self-Attention network has shown promise.
	This work featuers a Variational Quantum Classifier (VQC) algorithm is used with a language model to make classifcations of sentiment
	Research looks at quantum embedding where quantum concepts of superposition and entanglement are used to capture semantic relationships between words.
	Sentiment analysis is a standard NLP task where the primary emotional tone in a text is identified. Combining sentient analysis with document retrieval makes the process more intellegent and powerful.
	Sentiment analysis work with Italian language is comparitively unresearched (or underresearched?) compared to English.
	Quantum approach is unsupervised machine learning using a density matrix generated from two sentiment dictionaries

	Word relations identified in quantum 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
	Electra is chosen as a transformer over Bert thanks to outperforming Bert on the capture of contextual word representations.
	Electra requires a large number of resoruces to train, so a pretrained Electra model was selected for the study.
	Numerical representations of sentences are encoded into quantum states using amplitude encoding. N features encoded into log(base 2) N qubits. Actual qubits used are 10 with 768 features encoded. 10 = log 1024
	Quantum states are loaded onto qubits and the run through a VQC. An optimizer finds the quantum parameters that best capture the target labels.
	Results were compiled by using a simulation with python and qiskit, not on actual hardware although quantum circuit wouldn't need to be deep based on description. No circuit diagream was provided.
	Training was run for only 8 Epochs to avoid overfitting. Training set is 4476 sentences and test set is 500 sentences. F1 score used to compare models.
	Electra with Quantum achieved better F1 score than other models. For classical comparison, Bert was compared with and without preprocessing.
	Authors do not explain why they did not use an Electra only classical baseline. There was no preprocessing for the Electra-Quantum model, perhaps the Bert preprocessing was intended to be state of the art? Unclear.
05 September 2024	**Photonic probabilistic machine learning using quantum vacuum noise**
	This is an exploration of noise generated by various computer hardware, focusing on quantum hardware
	Discusses how noise and uncertainty can be a positive in machine learning circumstances
	Use probabalistic photonic neurons for learning on MNIST dataset

	Bias noise can be tuned in this model
	It's a different angle to examine benefits of quantum computing
19 September 2024	**Buildung Continuous Quantum-Classical Bayesian Neural Networks for a Classical Clinical Dataset**
	Given NISQ limitations, compare simple quantum model to classical. Used Breast Ultrasound Imaging public data (BreastMNIST) for testing.
	Examines a hybrid Classical / Quantum approach to leverage existing quantum hardware as best as possible
	Classical CNN with quantum updating of stochastic weights, model is very simple, 2 layers, one convolutional and one fully connected
	Related work notes that there has been another hybrid quantum system designed to work with BreastMNIST dataset.
	Challenge in working with Bayesian Neural Networks is sampling the posterior distribution. Here the study uses variational inference to approximate the posterior.
	Use born machines to encode the probability distributions into quantum states. (amplitude encoding)
	Binary classifier using cross entropy created to discriminate between born machine distributions and actual distributions
	Study tried a number of different VQC methods. After trial and error, "Nikoloska" had stable results with z- rotation and x- rotation gates followed by CNOTS. z- and x- allowed full range of rotations.
	"Romero" test architecture did better than other architectures, doing it with y- rotation gates. My observation: Most NISQ don't use y- gates, all z- and x-
	Authors then created custom VQC circuits after analyzing results. Came up with four custom circuits using controlled rotation instead of CNOT gates.
	A bigger dataset would have been more useful in determining optimal VQC circuit. Many of the custom designs performed similarly.

	Classical model had better accuracy than any other model but gap may decrease with better quantum advancements
	Commendable effort went into making sure that design functioned within NISQ limitations. The VQC design was very creative, however, NISQ hardware doesn't feature all of the gates detailed in the paper.
	In other words, it was definitely simulated, however, the quantum circuit design is an excellent area to research.
19 November 2024	**The Quantum Imitation Game: Reverse Engineering of Quantum Machine Learning Models**
	If you have a given quantum model for machine learning, can you reverse engineer it to obtain potentially sensitive data? is the key issue
	Concerns are around the unitary transformations (which are reversable) composing the core of quantum computing
	Researchers believe that access to model via third party or cloud offering may reveal parameters of model
	Steps to secure models include dummy layers and increasing complexity
	Countermeasures may impact overall accuracy but can be accounted for in design
14 December 2024	**Optimizing quantum machine learning for proactive cybersecurity**
	Paper examines serveral cybersecurity datasets used in ML training to see if an advantadge is gained with the usage of quantum computing
	Solid methodology section and quantum circuit descriptions. The authors did a good job of making sure the work could be duplicated
	The ML datasets were scaled down in complexity for use with qunatum hardware
	Quantum algorithms performed nearly as well as classical algortirms in this application
	Quantum machines/circuits still have a number of limitations today such as maximum feasible qubits for features, dataset size on quantum hardware, hardware limitations such as decoherence

30 December 2024	**Hybrid quantum enhanced federated learning for cyber attack detection**
	quantum inspired federated averaging optimization part of a larger learning process to detect cyber attacks
	QIFA is claimed to use superposition and entanglement to achieve model optimization
	Paper states that quantum support vector machine is the chosen algorithm
	The quantum inspired section of the paper looks like a standard ML calculation with a noise term added to it. Definitely not qSVM
14 January 2025	**Q-SupCon: Quantum-Enhanced Supervised Contrastive Learning Architecture within the Representation Learning Framework**
	Large datasets are important for proper training of deep learning algorithms, however, some domains such as medical have privacy restrictions which make it challenging to construct a large dataset
	Curse of dimensionality' an issue where higher dimension models need more data to fight overfitting. Classical methods of dealing with limited data are being explored.
	Qunatum computing examined in hopes that the method of encoding quantum data will mitigate issues with overfitting.
	Both quantum autoencoders and quantum CNNs are sought as potential methods to address limited datasets
	Authors propose a quantum supervised contrastive learning model
	Experiment has 4 qubits for qunatum data augmentation / image processing using y-axis rotations and c-not gates. 2x2 bit blocks are selected from a larger gray scale image.
	Pixel values are encoded as angle rotations on the quantum circuit. A measurement occurs, resulting in one of two states. The c-not gate performs a bit flip if one of those two states exist.
	After the bit flip operation the value is scaled back to the 255 gray scale pixel values. This random creates more image samples to train the model.

	After the augmentation, there is an autoencoder layer. It uses a parameterized quantum circuit is used to compress the images into fewer qubits
	A swap test with new qubits is run comparing data to the autoencoded information. Measurements from this swap information become the feature vectors and are sent to classification
	Finally, a variational quantum classifier method is used on the feature vectors to determine classification.
	Two parts have to be trained, the classifier and encoder.
	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.
	Fully quantum supervised learning, tested on real quantum hardware, using trapped ion infrastructure. Was also modelled on IBM qiskit software.
	Addresses sparce data caused by privacy issues, makes training challenging
	small sample sizes (ranged 25-150) selected from MNIST, FMNIST, KMNIST datasets for training
	Test accuracy reached 80%, considerably better than classical ResNet CNN run on GPUs (ResNet had 20% accuracy)
21 January 2025	**Quantum machine learning with Adaptive Boson Sampling via post-selection**
	Paper explores the promise of photon based quantum computers which can transmit photons optically in circuits
	Big advantage is that the circuits are more robust to decoherence
	Still have challenges with designing qunatum gates for this type of hardware
	Performed SVM classification on 1-D and 2-D datasets to test hardware
29 January 2025	**Re-locative guided search optimized self-sparse attention enabled deep learning decoder for quantum error correction**
	Uses machine learning to aid in Quantum Error Correction

	Extracts features via classical CNN and outputs expected quantum states
	Proposed method is competitive with other current QEC techniques in terms of accuracy
	Authors show some possible speed up over other techniques
30 January 2025	**Superconducting quantum computing optimization based on multi-objective deep reinforcement learning**
	Method is heavily based on quantum hardware calibration for optimal performance
	Runs a qunatum state through a process with a reward term
	Deep neural network uses DQN network and Adam
	The optimization allows more complex models with more qubits
11 February 2025	**Nav-Q: quantum deep reinforcement learning for collision-free navigation of self-driving cars**
	Quantum reinforcement algorithms applied to self driving cars
	Central to self driving cars is real-time interpretation / optimization of camera data
	Clasiscal approach to this is partially observable Markov decision processes and Deep Reinforcement Learning.
	Substanial time and effort is put into designing a reinforcement learning agent.
	The idea is to have a quantum computer do the work of the reinforcement learning agent. Previous attempts have been in a simplified environment, not with real world data
	Previous works also suggest having a parameterized quantum circuit replacing or at the end of the classical deep learning model. This would require a quantum computer essentially in the car.
	There is simply no way this could work, as quantum chips are only found in large datacenters. They must be cooled to near absolute zero which is not feasible for your car.

	Model is therefore hybrid. The Quantum computing is part of the actor-critic reinforcement learning.
	The study also presents an new data uploading / encoding technique for converting classical data to quantum.
	Previous Quantum works in DRL have been assessed in OpenAI Gym environments
	Self driving background: Car has a view of objects within 50M in 360 degree range.
	Car creates a 2D map of objects using grid and birds eye view. Projects car states for 1M, 50M and 100M away. Pedestrians are also tracked.
	NavA2C is classical baseline to Quantum component. Combines path planner with modified RL agent. Agent is only responsible for speed action
	Briefly, the agent outputs the speed action, along with a reward value, the car velocity, and the previous speed action. The speed action and the steering wheel angle are passed to the self driving simulator
	DRL agent has four main components: Encoder, Actor, LSTM cell, and critic.
	For the Quantum model, the Quantum circuit is used in the critic as it is only needed for training. If a part of the algorithm was needed for testing the quantum components could not be placed in a car.
	The Quantum critic uses a paramaterize quantum circuit that takes an input of the hidden state from the LSTM cell.
	In the given example circuit, 24 classical parameters are optimized across 4 qubits.
	Output is taken from the quantum curcuir and linearly recombined as a fully connected layer.
	Simulator uses Rx, Ry, and Rz gates. May not translate into quantum hardware. Training also ran for many iterations 5000-10000 iterations
	Included a normalization layer that wasn't present in the classical RL agent.

	Study noted that the Quantum steps increased the time it took to train model. Had to decrease training from 4 qubits with 2 layers to 2 qubits with 1 layer. Also reduced LTSM state input to 6 dimensions
	Normalized Effective Dimension of Nav-Q 10x times higher than classical suggesting that more model space utilized meaning that more complex functions could be modeled.
	The experiments were run on a noise-less simulated quantum circuit. Results were promising but may not generalize well.
	When introducing noise, stability of learning dereases substantially.
	Not a clear winner in terms of performance. Classical model better at collision avoidance, hybrid model better at optimal route selection
20 February 2025	**Direct entanglement detection of quantum systems using machine learning**
	Utilizing machine learning neural network to determine entanglement of quantum system
	ML algorithm takes quantum state input and then outputs whether or not entanglement is present.
	Polynomial amount of data O(N^c) where c is a constant is needed to train model
	Method bypasses need for complex quatum full state tomography (method of pseudocopying data)