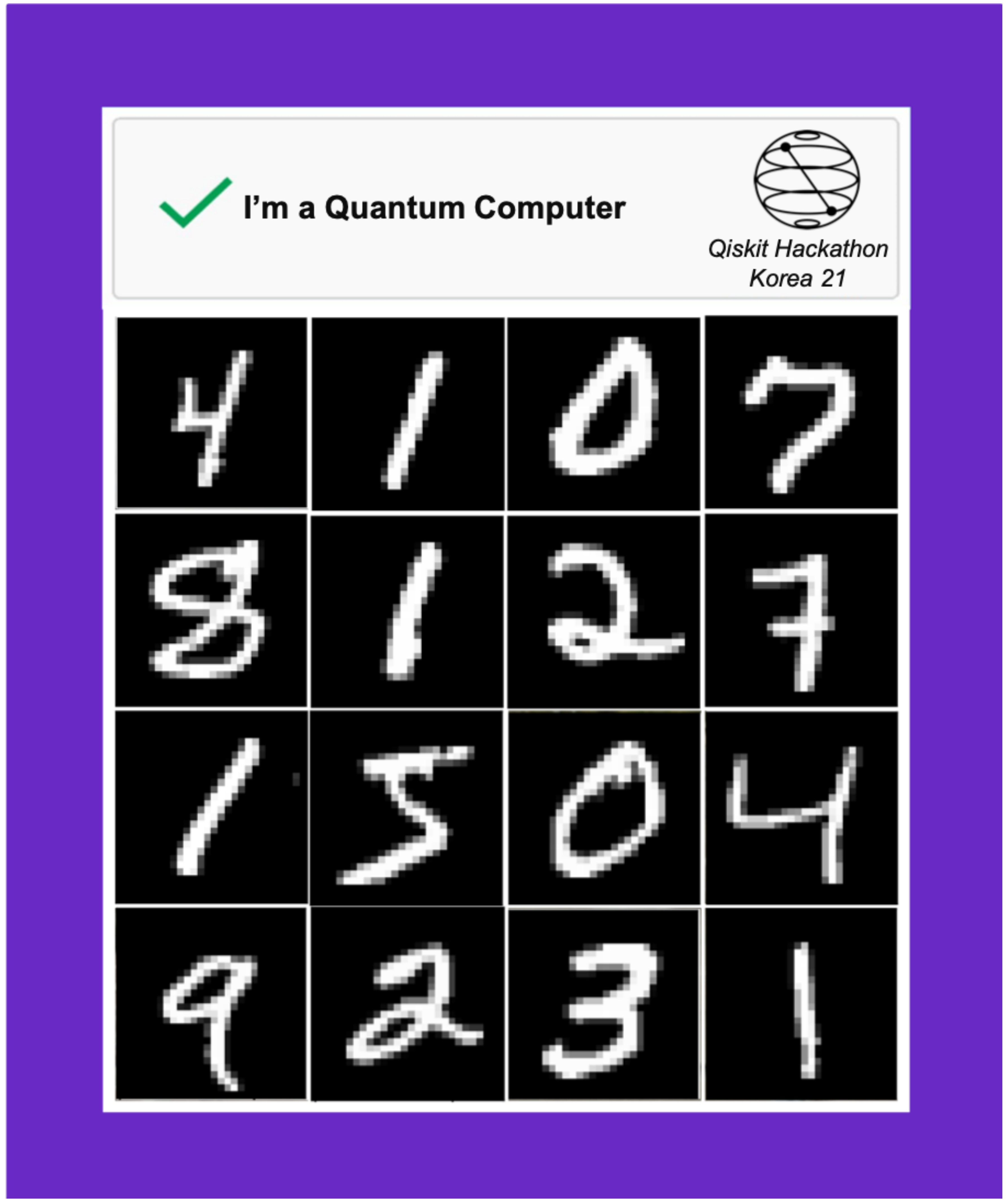
Exploring Hybrid quantum-classical Neural Networks with Pytorch and Qiskit

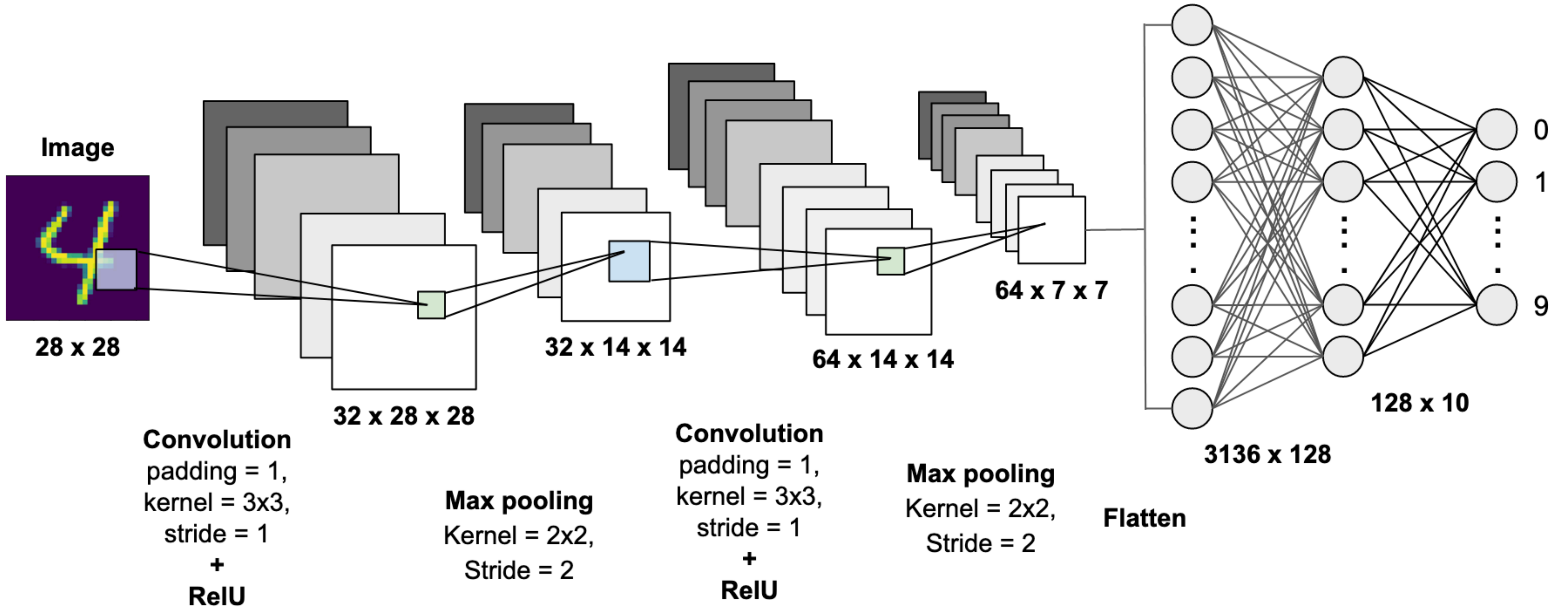
Team “Quanputing” #12
Jaehoon Hahm, Yoon Kwon, Dohun Kim, Yunseo Kim, Daeheon Yoon, Eunchan Lee
Kifumi Numata, Anna Phan

Motivation



Everything is Quantum Mechanical.

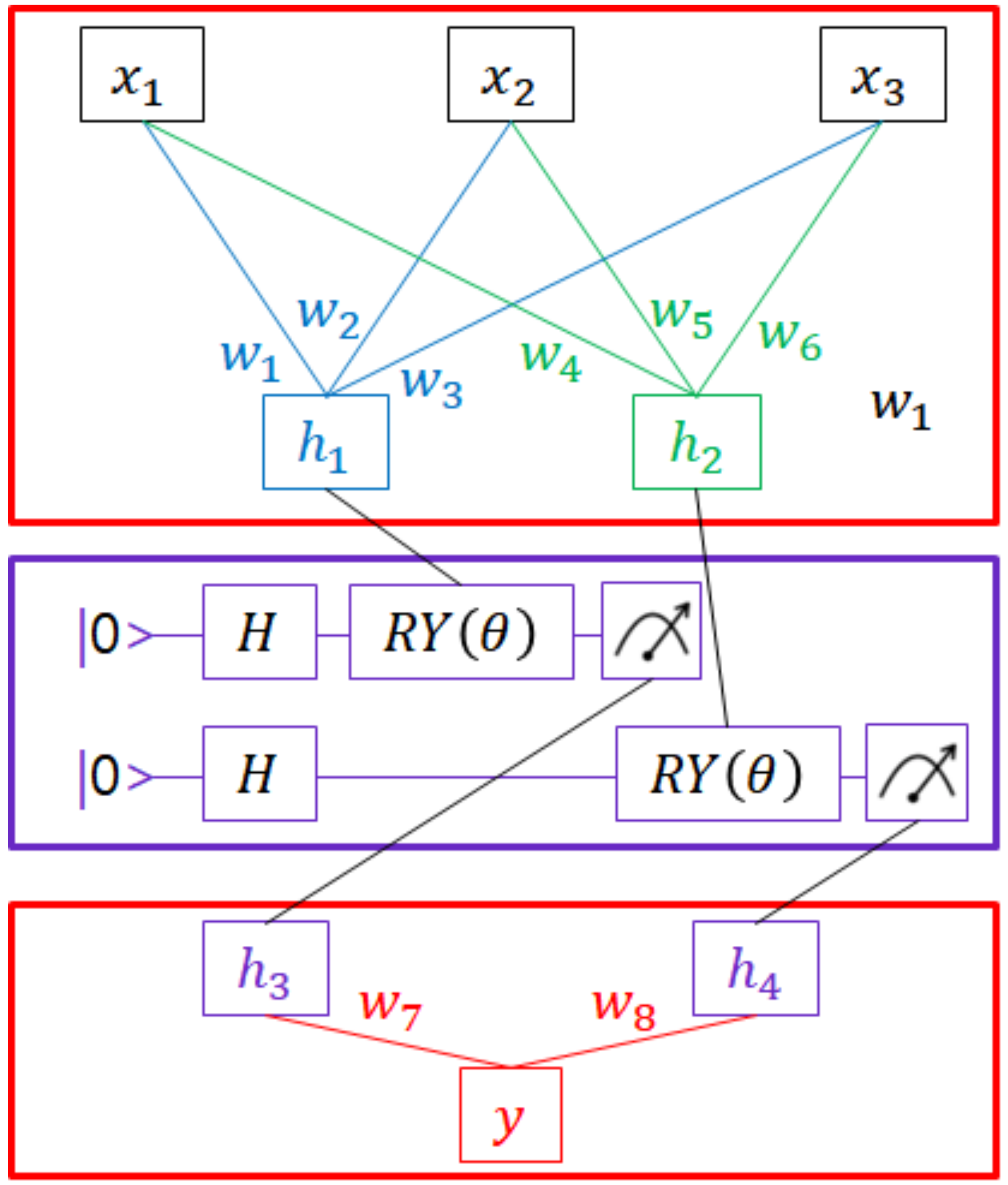
If input is essentially quantum data, then *quantum neural network* would be natural!



Can we make a neural network consist of quantum circuits?

Using this Hybrid quantum-classical Neural Networks, can we successfully classify handwritten digits(MNIST)?

Hybrid Neural Network



$$h_1 = \sigma(x_1 w_1 + x_2 w_2 + x_3 w_3)$$
$$h_2 = \sigma(x_1 w_4 + x_2 w_5 + x_3 w_6)$$
$$y = \sigma(h_3 w_7 + h_4 w_8)$$

Classical layer
By
PyTorch

Quantum Layer
By
Qiskit

Classical Layer
By
PyTorch

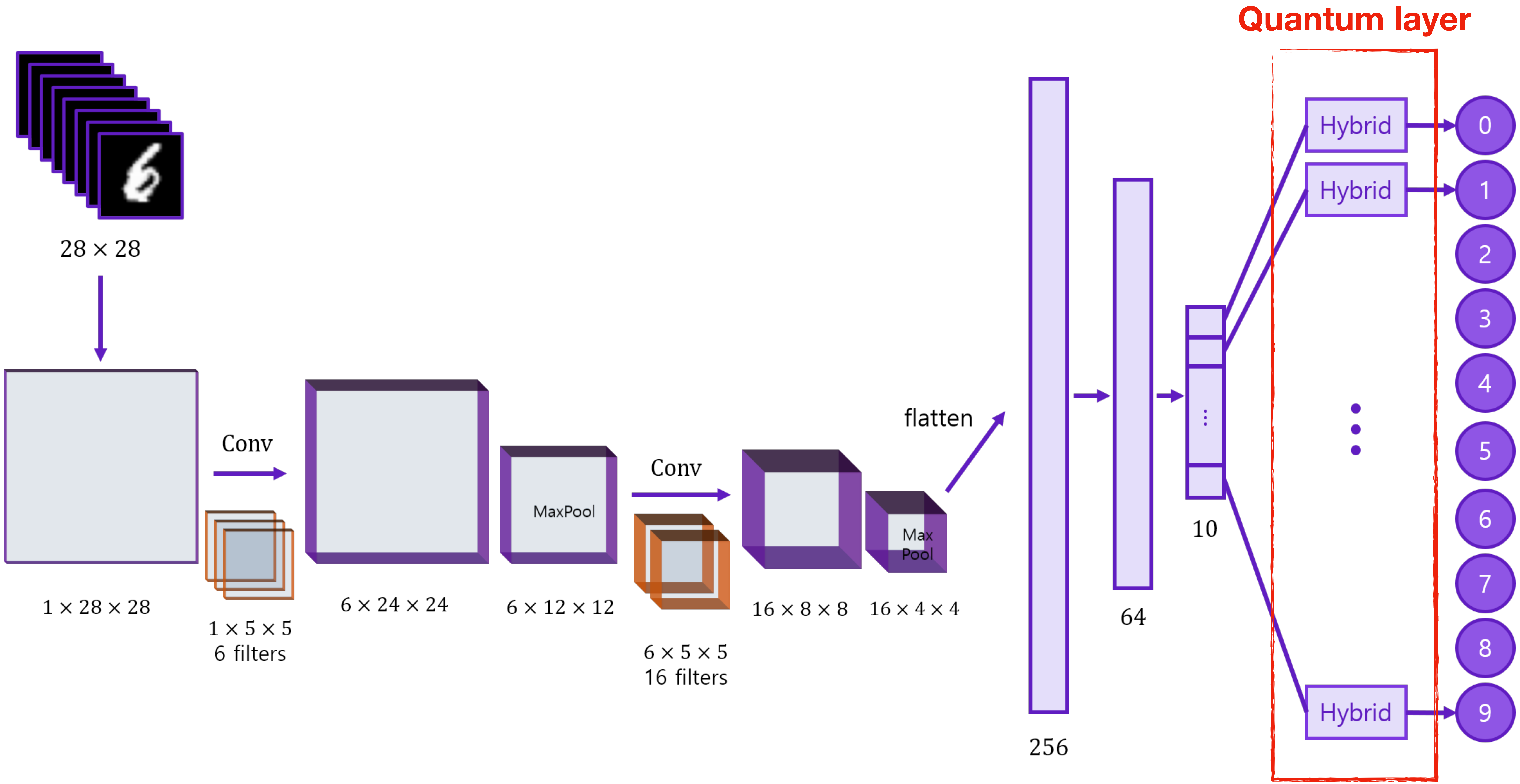
$$\nabla_\theta \boxed{\text{Quantum Circuit }(\theta)} = \boxed{\text{Quantum Circuit }(\theta + s)} - \boxed{\text{Quantum Circuit }(\theta - s)}$$

Gradient Descent of quantum circuit is achieved by the parameter shift rule.

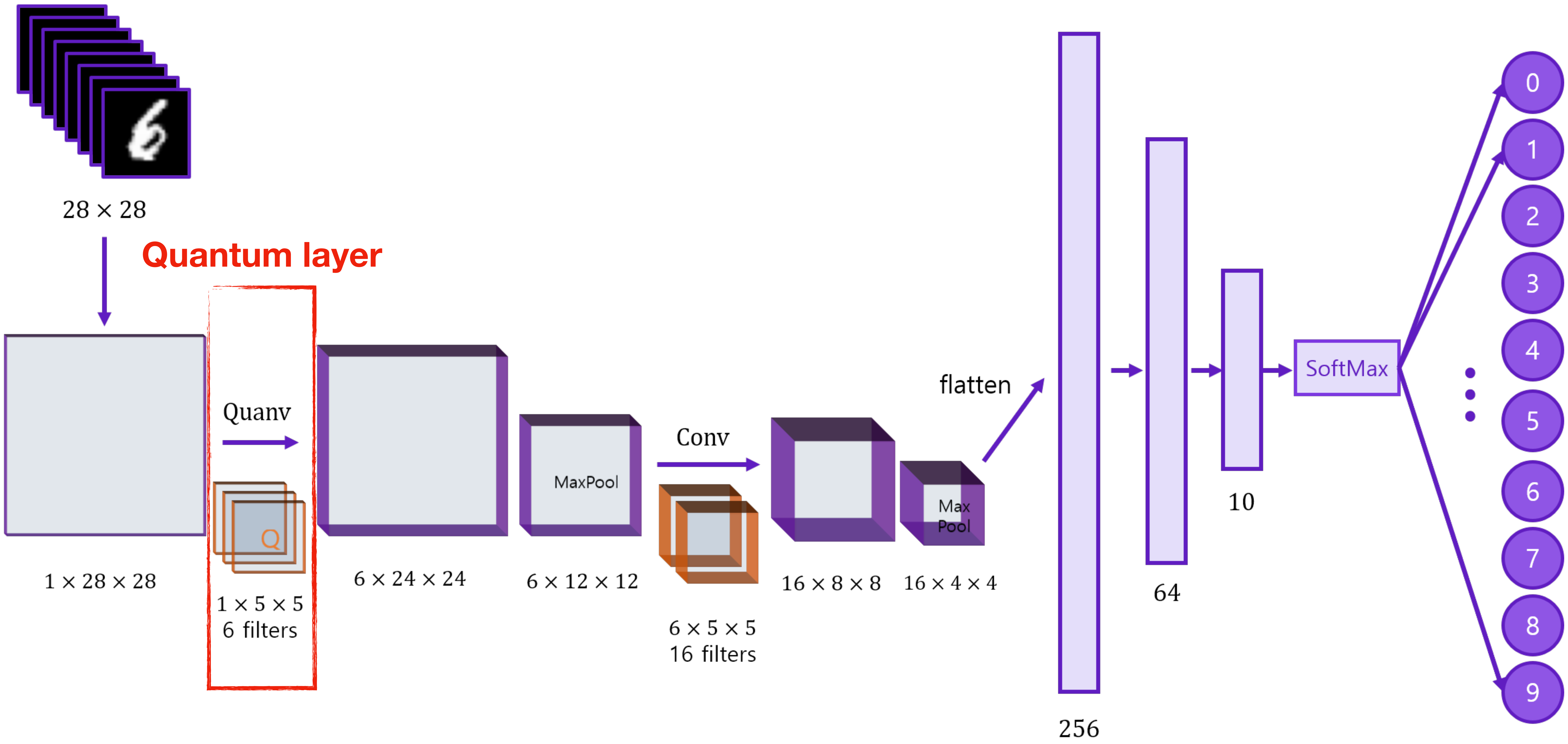
Quantum circuit is parametrized with certain parameters.

According to the parameter shift rule, we update the parameters and optimize the loss function.

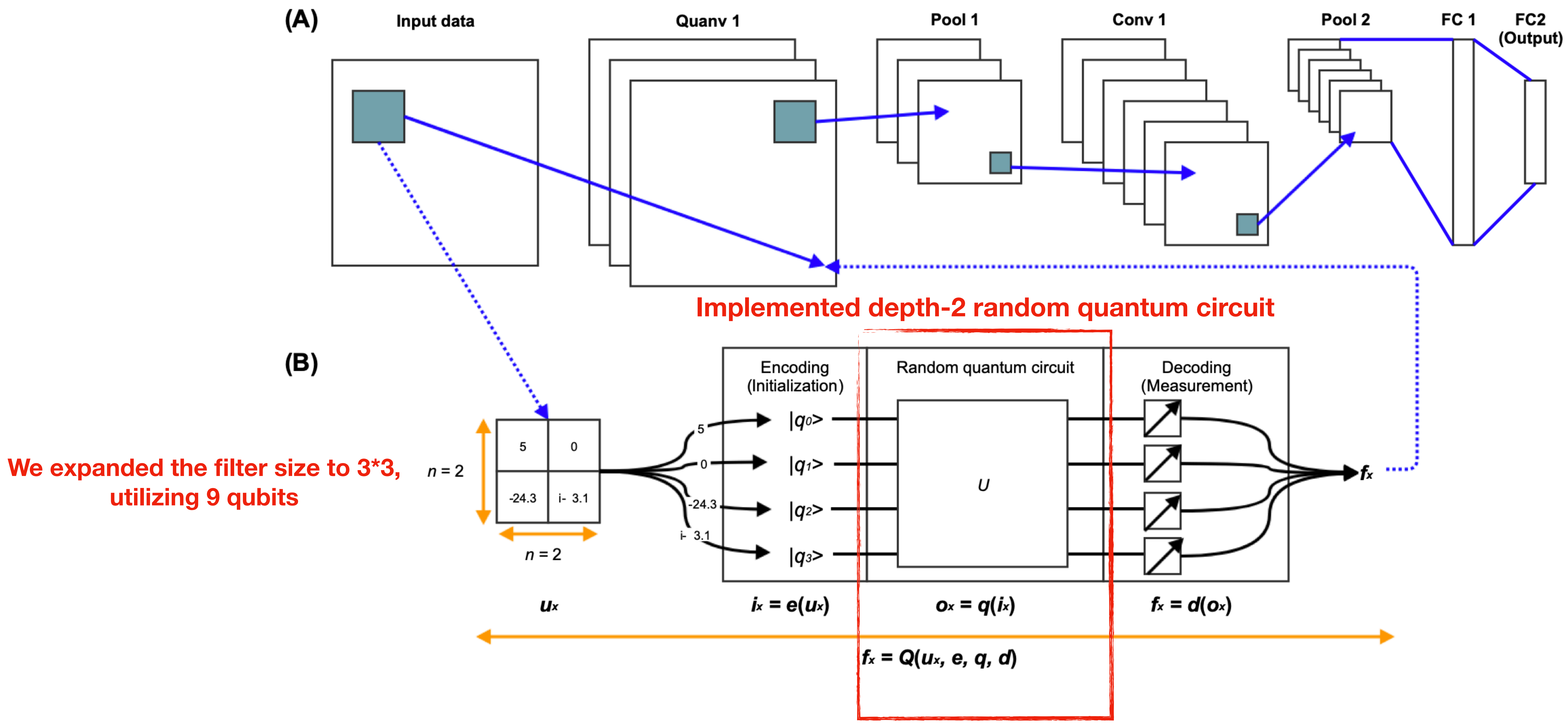
Qiskit tutorial (Fully-connected NN)



Tensorflow tutorial (Convolution)



Quantum-Convolutional Neural Network

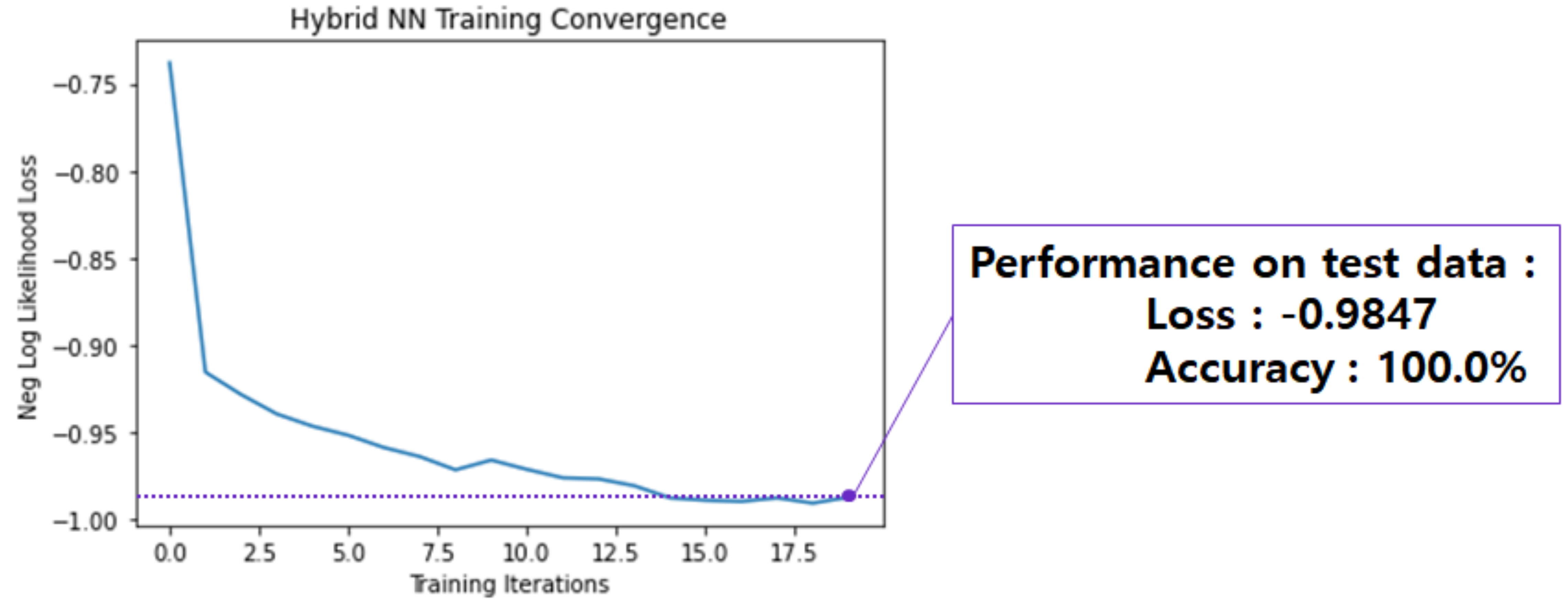


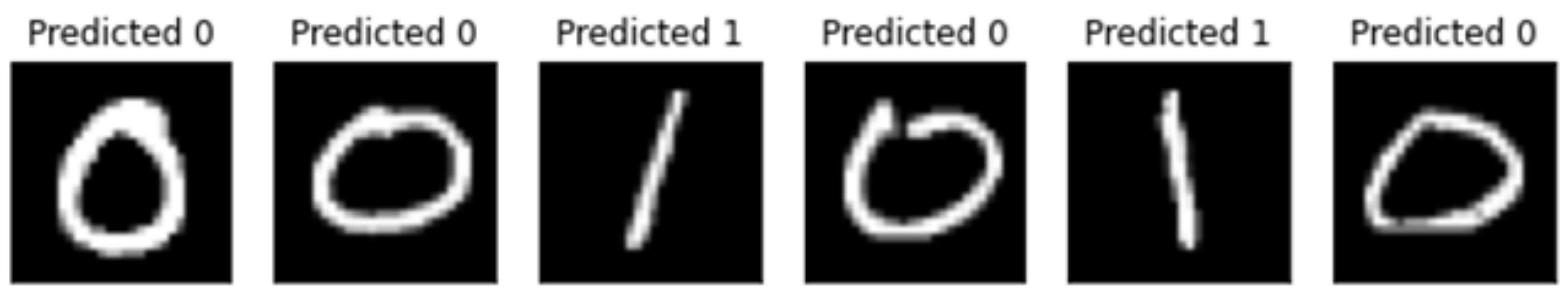
Reference: Henderson et al., “Quanvolutional Neural Networks: Powering Image Recognition with Quantum Circuits”, 2019

Result/ Performance

Qiskit Tutorial does only binary classification

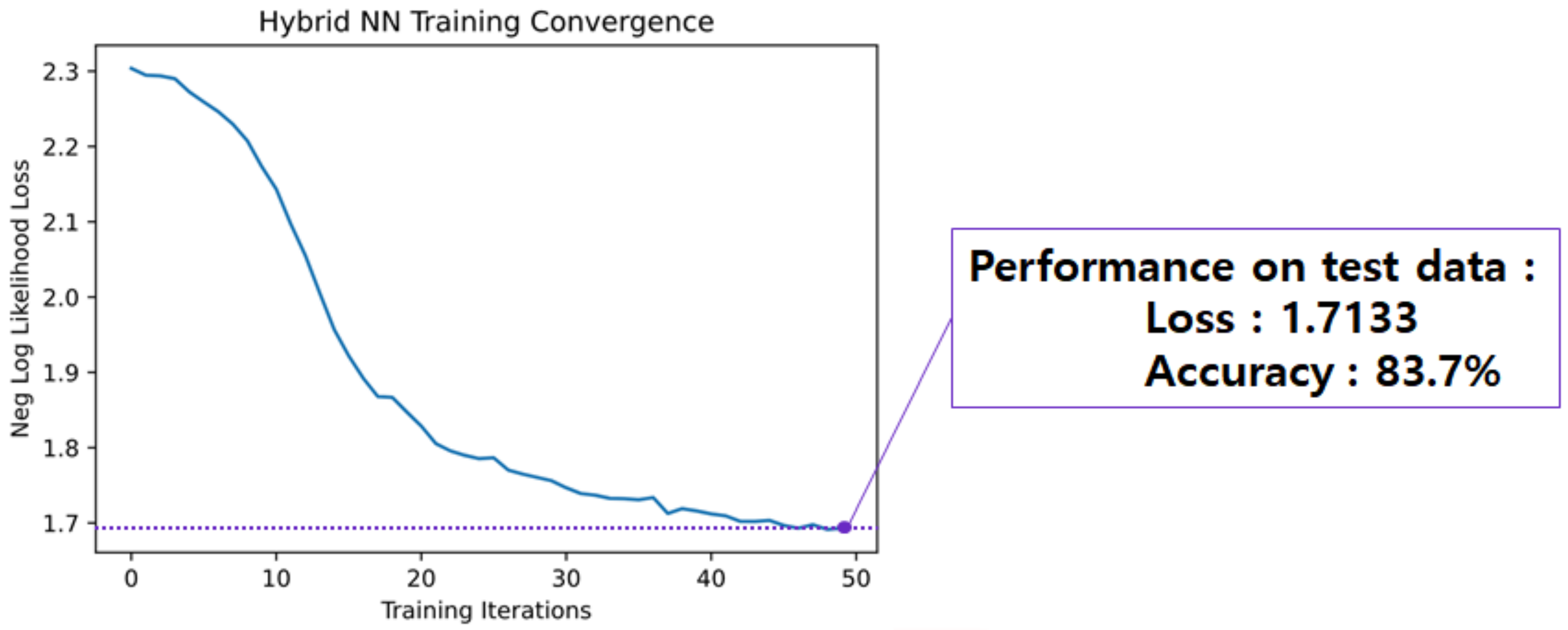
The graph of loss to the number of iterations.

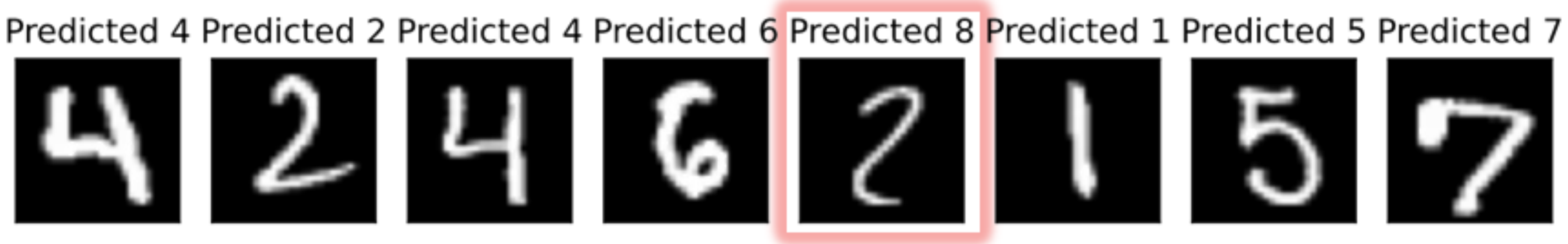




We expanded the network to classify full MNIST handwritten digit dataset !

The graph of loss to the number of iterations.





Summary/ Conclusion

1. We followed up quantum-classical hybrid neural network implementation in Qiskit Tutorial.
2. We found out the tutorial demonstrates only binary classification, so we expanded the model to classify full MNIST dataset. We obtained 83.7% accuracy.
3. In reference of Hendersen et al. (2019), we also implemented quantum-convolutional layer, but ran out of time while developing back propagation procedure.

Future Directions

1. Following Research can try implementing quantum-convolutional layer with efficient back propagation.
2. One can also try different models and tune hyper parameters for higher accuracy.
3. Training on different datasets or utilizing different quantum computer backend could be exciting.
4. Make all the neural network's layer in quantum circuits, and train on quantum datasets! (finally back to the first motivation)