

QAMP Spring 2022

#21

TorchRuntime tutorial 2.0 (QML)



Mentor:

Elena Peña Tapia



Mentee:

Inho Choi

Qiskit Runtime & TorchRuntime

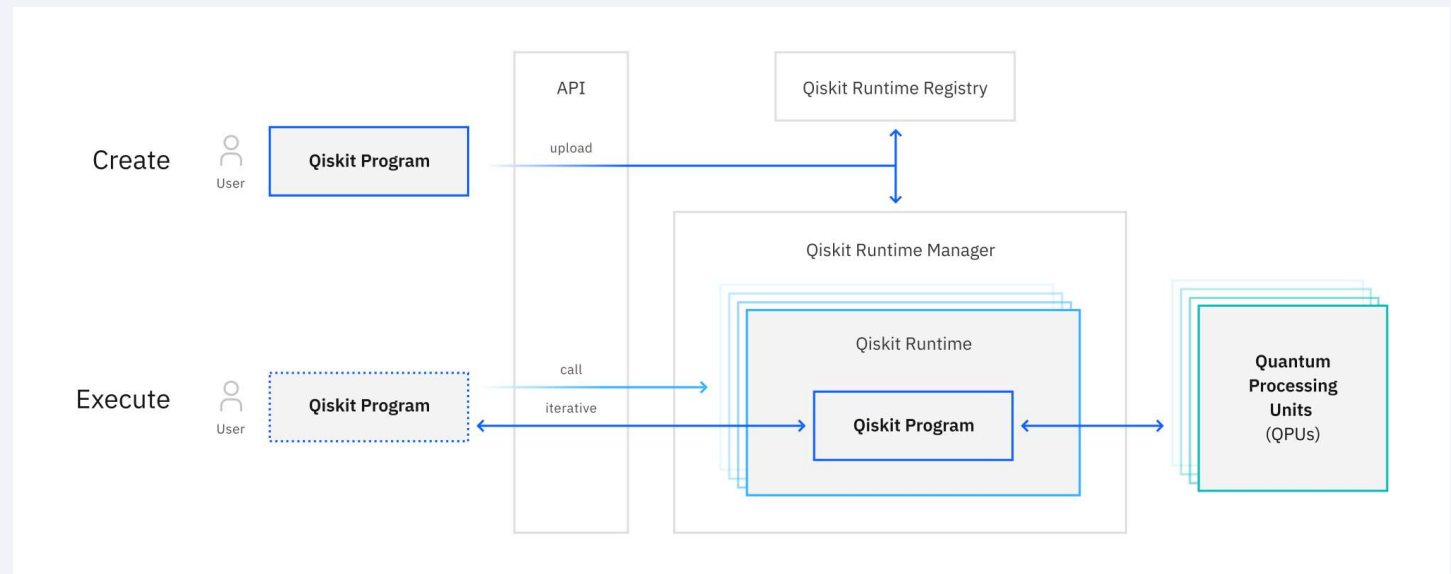


Qiskit Runtime

- The time spent queuing processes is **reduced** for iterative computation
- Achieved a **120x speedup** in a lithium hydride molecule simulation

Torch Runtime

- Allows training models or predicting the outputs with trained models significantly **faster**



<https://quantum-computing.ibm.com/lab/docs/iq/runtime/start>

Project Goal

- **New** torch runtime **tutorial** that shows a more realistic usage with complex examples

• NOTE

This page was generated from docs/tutorials/06_torch_runtime.ipynb.

Torch Runtime 📄

In this tutorial, we introduce Torch Runtime, and show how to use it via the `TorchRuntimeClient` class in Qiskit Machine Learning. Torch Runtime leverages Qiskit Runtime for hybrid quantum-classical machine learning based on a PyTorch Module. It allows training models or predicting the outputs with trained models significantly faster. We show how to use Torch Runtime with two simple examples for regression and classification tasks in the following.

1. Regression 📄

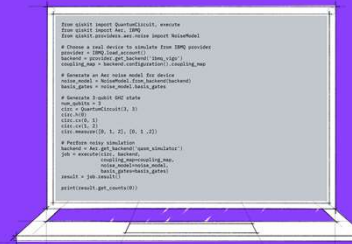
First, we show how to use Torch Runtime via `TorchRuntimeClient` using the simple regression example. In the example, we will perform a regression task on a randomly generated dataset following a sin wave.

```
[1]: import numpy as np
import matplotlib.pyplot as plt

from torch import Tensor, manual_seed, is_tensor
from torch.nn import Linear, CrossEntropyLoss, MSELoss
from torch.optim import LBFGS, Adam
from torch.utils.data import Dataset, DataLoader

from qiskit import Aer, QuantumCircuit
from qiskit.utils import QuantumInstance, algorithm_globals
from qiskit.opflow import AerPauliExpectation
from qiskit.circuit import Parameter
from qiskit.circuit.library import RealAmplitudes, ZZFeatureMap
from qiskit_machine_learning.neural_networks import CircuitQNN, TwoLayerQNN
from qiskit_machine_learning.connectors import TorchConnector
from qiskit_machine_learning.runtime import TorchRuntimeClient, TorchRuntimeResult
```

Update the code for the new runtime API



Add python/torch/qiskit-machine-learning version checks



Add key serialization information

- Add solution for current unpickling error.
- Show how to perform classification on the MNIST dataset using a hybrid quantum-classical neural network and TorchRuntime
- Research on serialization to justify these fixes

