Womanuim Quantum+AI Project

Quantum Machine Learning for Conspicuity Detection in Production

Jose Antonio Contreras Ortiz

Ricardo Rioda Santiago Sanchez

Womanuim Quantum+AI Project

Contents of this template

You can delete this slide when you're done editing the presentation

Team presentation

Jose Antonio Contreras Ortiz

https://www.linkedin.com/in/antonio-contreras-ortiz-0661561a6/

Biomedical Engineer who has a strong mathematical background, programming skills in different languages and has worked as a freelancer web developer for Mexican companies.In school, I have earned experience in data science and machine learning. I'm passionate about sciences and I enjoy learning about new topics and I can quickly understand new ideas.

Ricardo Rioda Santiago Sanchez

https://www.linkedin.com/in/antonio-contreras-ortiz-0661561a6/

Computer Systems Engineering student at the Tijuana Institute of Technology, and has participated in the 2019 National Physics Olympiad (ONF). He currently works in the data science area at ITJ Labs. He is passionate about mathematics, physics, and machine learning.

Womanuim Quantum+AI Project

Task 1

Task 1 conclusion

After working through the various codebooks, I realized that the materials provided by Pennylane are quite good. We enjoyed most of the content and are continuing with the different sections of the Pennylane codebooks. Most of the tasks are quite interesting. However, we would like to see future notebooks that focus on solving machine learning tasks. While the demos are helpful, constructing and solving these modules would enhance our understanding of the technologies and make the material more accessible to a broader audience. Here is our progress with the codebooks:

Jose Antonio Contreras Ortiz

- Introduction to Quantum Computing
- Single-Qubit Gates
- Circuits with Many Qubits
- Basic Quantum Algorithms
- Grover's Algorithm
- Noisy Quantum Theory
- Quantum Error Correction
- Quantum Fourier Transform
- Quantum Phase Estimation
- Shor's Algorithm

Ricardo Santiago Rioda

Theory Navigation Resources Hide

- Introduction to Quantum Computing
- Single-Qubit Gates
- Circuits with Many Qubits
- Basic Quantum Algorithms
- Grover's Algorithm
- Noisy Quantum Theory
- Quantum Error Correction
- Quantum Fourier Transform
- Quantum Phase Estimation
- Shor's Algorithm

Task 2

For Task 2, we focused on developing a variational classifier using PennyLane, where optimization techniques were applied to tackle the problem. We worked on the Iris classification task and tackled the parity function problem.

Just as in machine learning, we want to assess the generalization ability of our model. This is done by evaluating the model on examples it has never seen before. For this purpose, we use the test data.

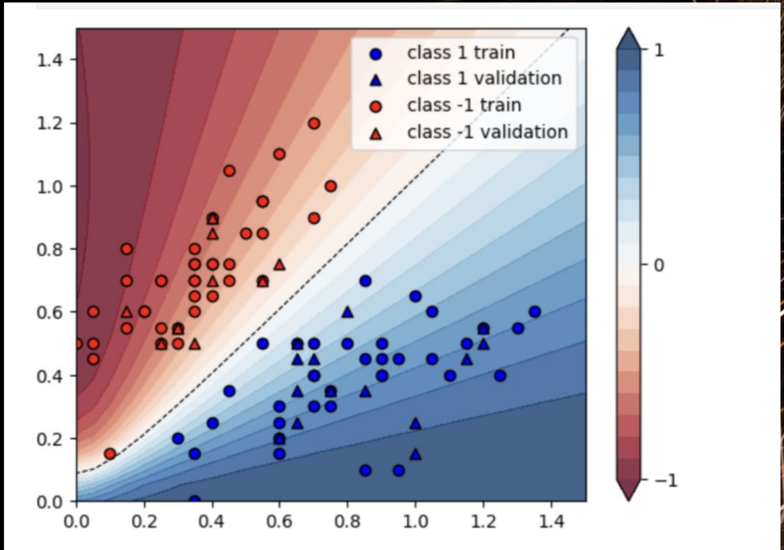
```
data = np.loadtxt("https://raw.githubusercontent.com/XanaduAI/qml/master/_static/demonstration_assets/variational
X_test = np.array(data[:, :-1])
Y_test = np.array(data[:, -1])
Y_test = Y_test * 2 - 1  # shift label from {0, 1} to {-1, 1}

predictions_test = [np.sign(variational_classifier(weights, bias, x)) for x in X_test]

for x,y,p in zip(X_test, Y_test, predictions_test):
    print(f"x = {x}, y = {y}, pred={p}")

acc_test = accuracy(Y_test, predictions_test)
print("Accuracy on unseen data:", acc_test)
```

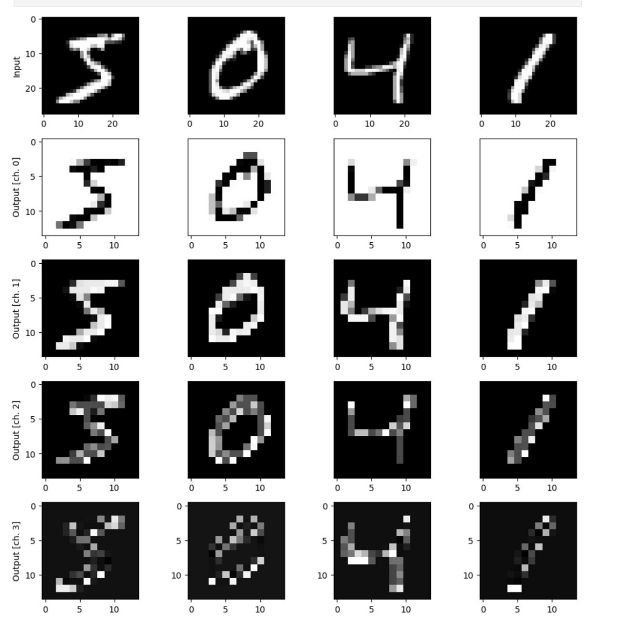
```
x = [0 0 0 0], y = -1, pred=-1.0
x = [0 0 1 1], y = -1, pred=-1.0
x = [1 0 1 0], y = -1, pred=-1.0
x = [1 1 1 0], y = 1, pred=1.0
x = [1 1 0 0], y = -1, pred=-1.0
x = [1 1 0 1], y = 1, pred=1.0
Accuracy on unseen data: 1.0
```

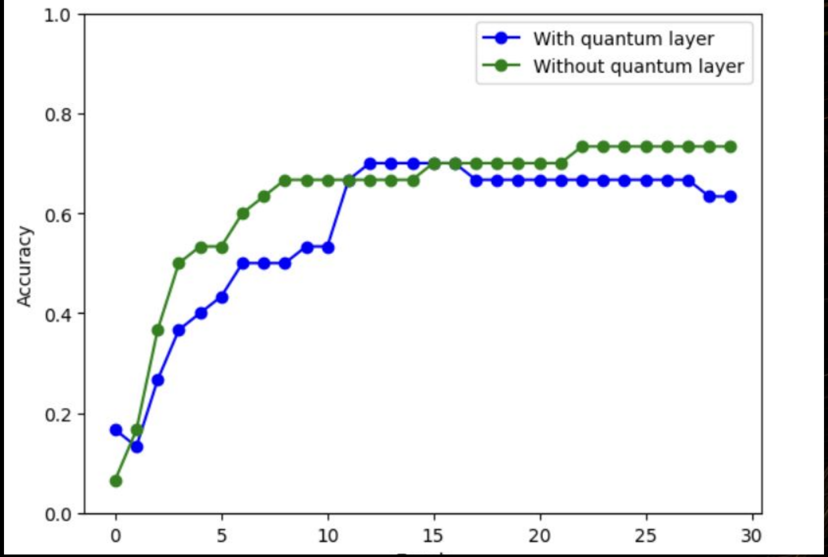


Womanuim Quantum+AI Project

Task 3

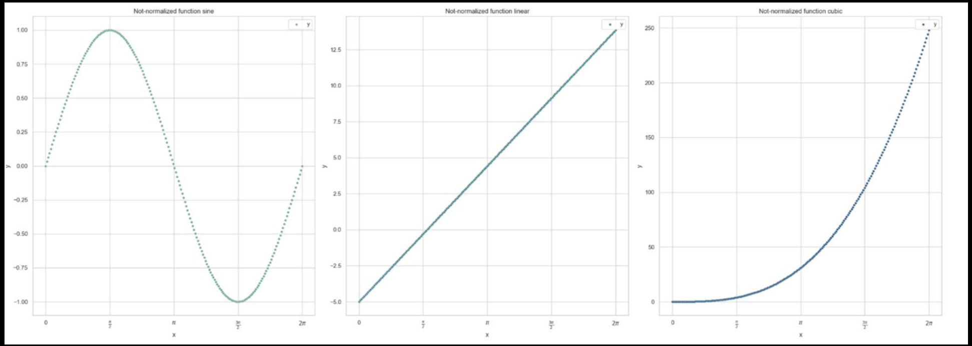
For Task 3, we worked on MNIST dataset using Hybrid quantum-classical model.

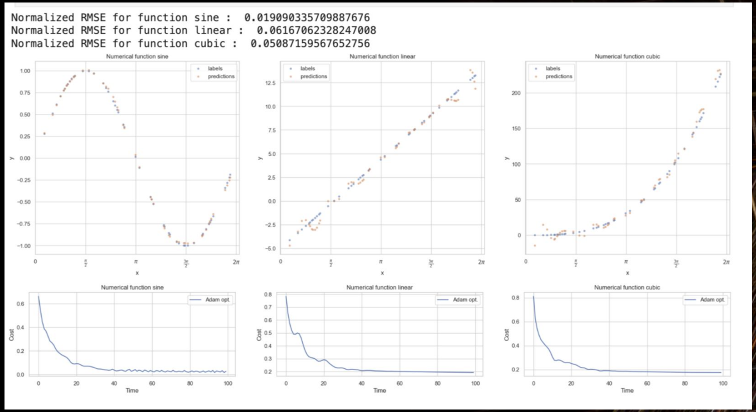




Task 4

We worked on the prediction of a regular sine function, a positive linear function and a cubic function.





The more layers you add to the variational classifier, the better it can approximate any function, thanks to the principles behind the Fourier transform.

Womanuim Quantum+AI Project

Task 5

TIG Aluminium 5083

Phase 1: Dataset Reorganization

Final output

Here's what we got

```
dataset
  test
    burn through
    contamination
    good weld
    lack of fusion
    lack of penetration
    misalignment
  train
    burn through
    contamination
    good weld
    lack of fusion
    lack of penetration
    misalignment
```

~/Desktop/QuantumML/W

This notebook focuses on converting the provided dataset into a usable format for machine learning models.

The original structure of the dataset was unsuitable, so we needed to reorganize the folder structure to make it compatible with machine learning requirements.

Womanuim Quantum+AI Project

Phase 2: Binary dataset

Resampling data and creating binary dataset

To balance the dataset, we used random resampling to obtain a fixed number of images per class. We chose to select 10,000 images for the "Good weld" cla5s and 2,000 images for each of the other classes. The decision to use 2,000 images was based on the fact that the "Burn through" class had over 2,000 images. We opted not to use data augmentation, as we believe future images will have consistent camera settings, orientation, and lighting conditions.

```
from datasets import Dataset
# Function to sample the desired number of images per class
def sample_dataset(dataset, class_counts):
    dataset_final = Dataset.from_list([])
    for label, count in class_counts.items():
        print(f"Class {label} Progess")
        # Get all indices for the current label
        dataset_only_one_label = dataset.filter(lambda example: example["label"] == label)
        dataset_only_one_label = dataset_only_one_label.shuffle()
        dataset_only_one_label = dataset_only_one_label.select(range(count))
        # Randomly sample the desired number of indices for the current label
        dataset_final = concatenate_datasets([dataset_final, dataset_only_one_label])
    return dataset_final

# Specify the desired number of images per class
class_counts = {
    0: 2000,
    1: 2000,
    2: 10000,
    3: 2000,
    4: 2000,
    5: 2000
}

# Sample the dataset to create the balanced dataset
balanced_dataset = sample_dataset(dataset, class_counts)
```

```
Class 0 Progess
Class 1 Progess
Filter:   0%|          | 0/33254 [00:00<?, ? examples/s]
Class 2 Progess
Filter:   0%|          | 0/33254 [00:00<?, ? examples/s]
Class 3 Progess
Filter:   0%|          | 0/33254 [00:00<?, ? examples/s]
Class 4 Progess
Filter:   0%|          | 0/33254 [00:00<?, ? examples/s]
Class 5 Progess
Filter:   0%|          | 0/33254 [00:00<?, ? examples/s]
```

Final result a dataset of 20000 images

In this notebook, we aimed to preprocess the data and create a binary dataset using the Hugging Face library.

This library offers significant advantages in image dataset creation, primarily due to its memory mapping method for loading datasets.

This approach allows us to efficiently handle and modify large datasets, such as our 6GB collection of images. All of this processing was done on Kaggle, as it facilitated easier online data loading.

Womanuim Quantum+AI Project

Phase 3: Classical Model

```
warnings.warn('was asked to gather along dimens
```

[545/545 46:04, Epoch 4/5]

Epoch	Training Loss	Validation Loss	Accuracy
0	0.060800	0.087856	0.970000
2	0.043600	0.035060	0.987333
4	0.023600	0.023507	0.990667

Vision Transformer Processing The model first processes the input images using the Vision Transformer. This step extracts high-level features from the images.

Phase 4: Quantum model

Epoch	Training Loss	Validation Loss	Accuracy
0	0.440400	0.394224	0.956167
2	0.432300	0.372869	0.967833
4	0.399300	0.357715	0.981333

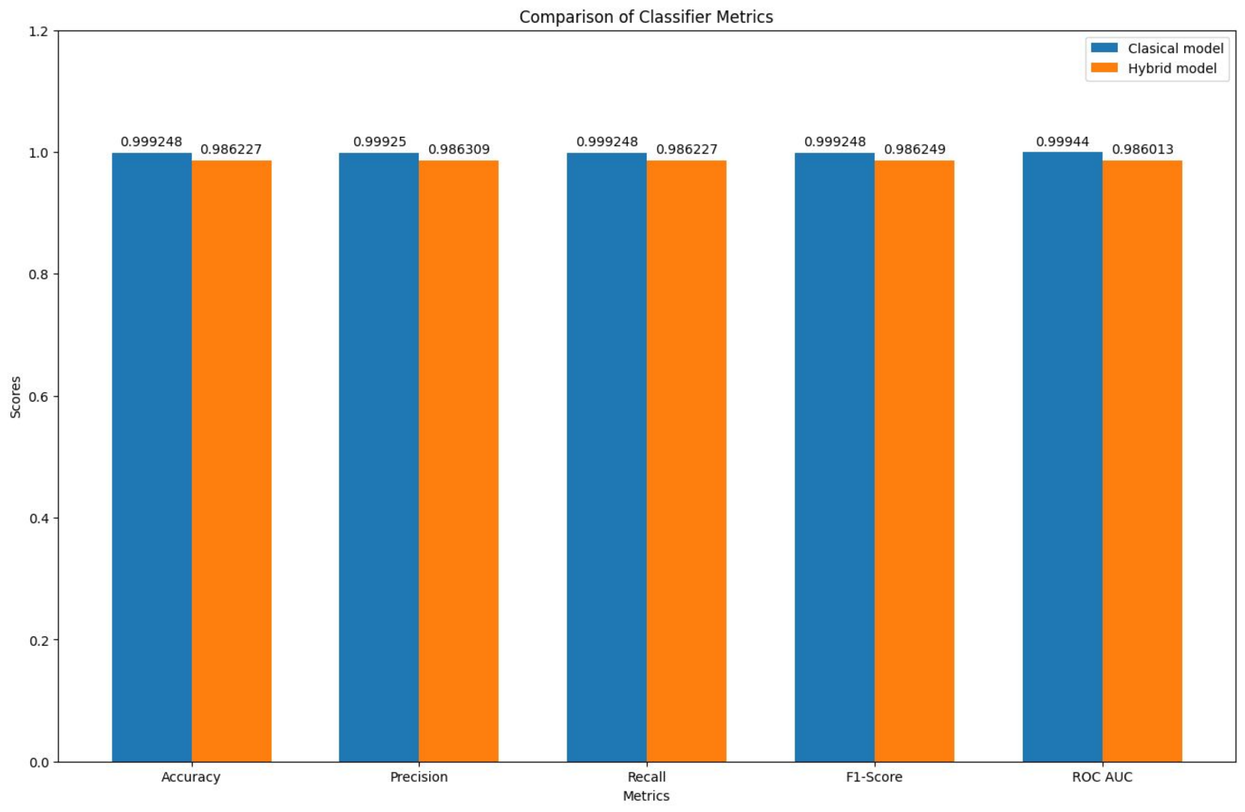
- Quantum-Enhanced Vision Transformer -

We created a hybrid model that combines a Vision Transformer architecture with a quantum circuit. The Vision Transformer processes the image input, and its output is passed through a dense layer. The resulting features are then fed into a quantum layer, implemented as a quantum circuit with 2 qubits.

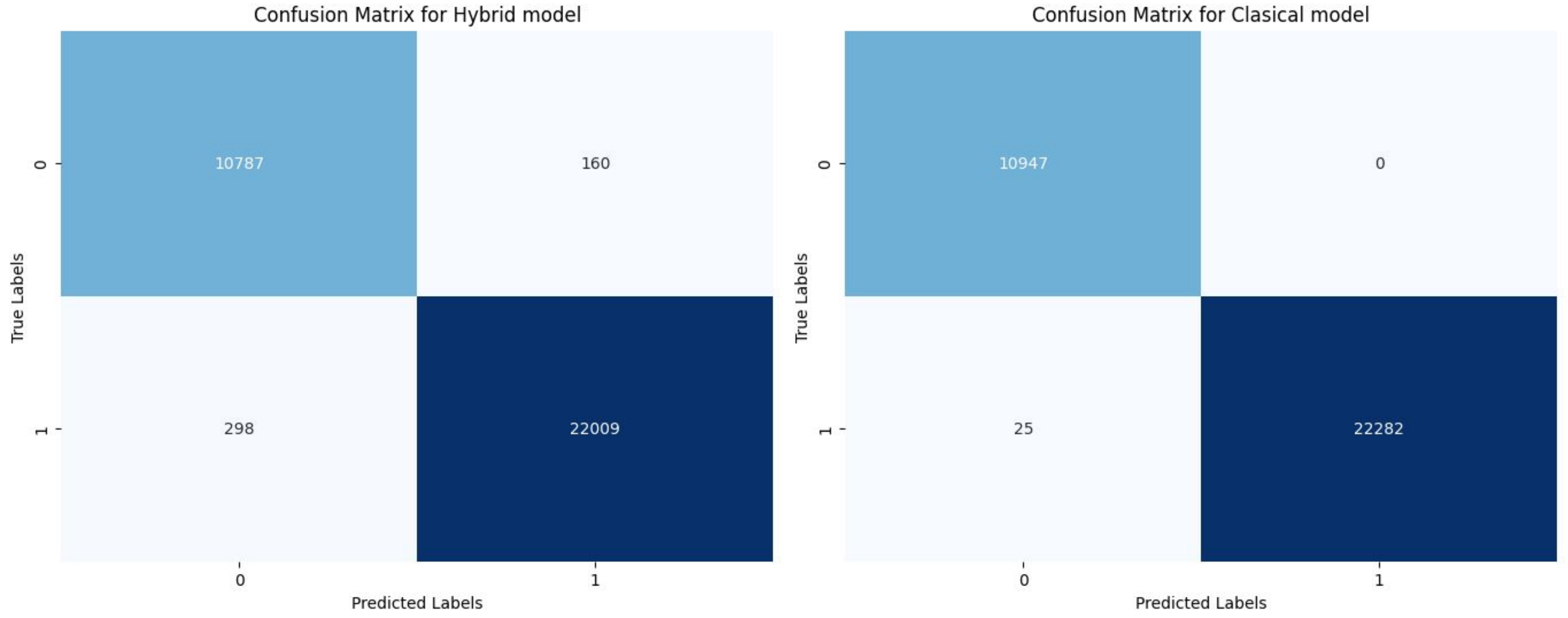
This quantum layer acts as a final transformation before producing the output logits, which can be used for classification. The quantum circuit is integrated into the model as a custom layer, enabling quantum computations on the features extracted by the Vision Transformer.

Womanuim Quantum+AI Project

Phase 5: Comparison between models



Phase 5: Comparison between models



References

Nielsen, M. A., & Chuang, I. L. (2010). *Quantum computation and quantum information: 10th anniversary edition*. Cambridge University Press.

Géron, A. (2019). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems*. O'Reilly Media.

Tunstall, L., von Werra, L., & Wolf, T. (2023). *Natural language processing with transformers: Building language applications with Hugging Face*. O'Reilly Media.

Hidary, J. D. (2021). *Quantum computing: An applied approach*. Springer.

Schuld, M., & Petruccione, F. (2021). *Machine learning with quantum computers*. Springer.