QRNN (QCNN-Adaptation)

1. Objective

The goal of this experiment is to extend the quantum RNN architecture explored in previous iterations by integrating improved design choices for encoding, loss computation, and overall model structure. This builds upon the observations from our earlier experimentation (Quantum_Report.pdf), where a shallow and stacked (2-layered) QRNN failed to generalize effectively. The primary focus is to make foundational improvements before exploring other architectural changes.

2. Key Enhancements

Component	Previous Iteration	Current Iteration
Dimensionality Reduction	PCA (fixed projection)	Autoencoder (learned compression)
Input Dimension	5D, 15D, 32D	8D, 32D
Quantum Output	1 PauliZ expectation	2 PauliZ expectations
Loss Function	Binary Cross-Entropy with Sigmoid	Cross-Entropy with Softmax
Hidden State Propagation	Re-encoded hidden with logit	Re-encoded hidden with tanh(logits.sum())

3. Model Overview

- Input Embedding

  - Used pre-trained 100 dimensional GloVe vectors for words

  - Trained an autoencoder to compress these vectors into a 32D or 8D latent space

  - Achieved faster encoding and reduced circuit depth

- Quantum Layer Architecture (QRNN-Adaptation)

  - Input: 8D per word, fixed sequence length of 10 tokens

  - Encoding: Dense Angle Encoding using 3 rotations (RX, RY, RZ) per wire on 4 wires/qubits (4th wire/qubit used for hidden state)

  - Hidden State: RY encoded on wire 3 using tanh(logits.sum()) from the previous word

  - Entanglement: Basic CNOT Entanglement ($0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 0$)

  - Trainable Parameters: 6 for input (RY/RZ per wire) + 2 for hidden state

  - Output: PauliZ(wire-2), PauliZ(wire-3) → Softmax (2-class classification)

4. Experimental Results

- Training

  - Epochs: 50

  - Average Time per Epoch:

    - ~ 400 seconds (32D input)

    - ~ 150 seconds (8D input)

  - Training Accuracy:

    - Plateaued at 49.75% (32D input)

    - Plateaued at 50.12% (8D input)

  - Loss: Remained stagnant with no meaningful drop

- Testing

  - Accuracy:

    - 49% (32D input)

    - 50% (8D input)

It is observed that the training and inference speeds, as well as training and testing accuracies, improved with lower-dimensional input. However, the overall model performance remains poor. Further experiments will involve architectural improvements discussed in the next section.

5. Proposed Next Steps

To overcome the performance ceiling, the following structural improvements are planned:

- **Two Stacked QRNN Layers**
    - Each with separate QNodes and trainable weights
    - Hidden state passed from Layer 1 → Layer 2
- **Interaction Layers Between QRNNs**
    - Add entanglement layers between the QRNN layers
- **Final Classifier Layer**
    - Ring entanglement + rotation gates before PauliZ measurements
    - Output 2 or 4 expectation values for more expressive classification

These improvements aim to address the lack of depth and abstraction in the current QRNN version.

6. Conclusion

QRNN-Adaptation introduced several foundational improvements, including proper loss functions, a two-output readout mechanism, and autoencoder-based input dimensionality reduction. While these changes marginally improved accuracy and efficiency, the results underscore the need for more expressive architectures. The upcoming Quantum Recurrent Neural Network will incorporate stacked layers and inter-layer quantum interactions (entanglement) to better model sequential relationships.

The complete implementation of QRNN-Adaptation is available in **QRNN-Adaptation.ipynb**.

Name: Ashutosh Rai
Scholar No: 2422308
Guide: Dr. Partha Pakray