Recurrent Quantum Neural Networks - Final Report

Ankit Kumar, Arnab Dey, and Mangesh Singh

Third Year UG, Plaksha University

PL3001: Quantum Computing

Prof. Nitin Upadhyaya

26 May 2024

1. Abstract

Many sequence-to-sequence models in machine learning, including speech synthesis and machine translation, are based on recurrent neural networks. Applied quantum computing, however, is still in its early stages. We offer an implementation in Pytorch to examine the performance of the model, enabling the optimisation of parametrized quantum circuits with thousands of parameters in a relatively efficient manner. By benchmarking optimization hyperparameters, we set up a QRNN training setup and examine appropriate network topologies for MNIST classification.

2. Introduction

The project is driven by the absence of any viable recurrent quantum network. Although Variational Quantum Eigensolvers (VQEs) exist, the resultant Quantum Circuits are very dense, compressing a lot of parameters into a relatively compact circuit [1]. The high density of entangling gates, lack of correlation between parameters results in highly over-parameterized models which are hard to train on classification tasks on inputs larger than a few bits.

This project focusses on constructing a QRNN, and compares its performance on non-trivial tasks such as sequence learning and integer digit classification. The reference paper exploits the nature of quantum mechanics. The interactions of any quantum system can be described by a Hermitian Operator $\mathcal{H}$ which generates the system's time evolution under the unitary map:

$$U(t) = e^{-itH}$$

Further any quantum circuit comprising a sequence of individual unitary quantum gates of the form $U_i(t_i)$ for a set of parameters t_i is intrinsically unitary and inherently linear

[2]. This is promising because then a parameterized quantum circuit serves as a prime candidate for unitary recurrent network [3].

3. Methodology

a. Parameterized Quantum Gates

In the sense of switching between parameterized single-qubit gates and entangling gates, such controlled-not operations, typical VQE quantum circuits are extremely dense. This offers the benefit of packing a lot of parameters onto a comparatively small circuit. However, although such circuits are known to constitute a universal family, models with inputs greater than a few bits are currently difficult to train due to their high density of entangling gates and lack of connectivity between parameters [4].

In this work, we build a highly-structured parametrized quantum circuit known as a QRNN cell. It is constructed so that few parameters are repeatedly used and so that each one controls a much higher-level logical unit than the parts of a VQE circuit. Mainly composed of an extension of [5], the cell's novel form of quantum neuron rotates its target lane based on a non-linear activation function applied to polynomials of its binary inputs (see figure 1).

The cell is made up of an input stage combination that writes the current input into the cell state at each step. Subsequently, there are several work phases that utilize input and cell state information to calculate, culminating in a final output phase that generates a probability density across potential predictions [2].

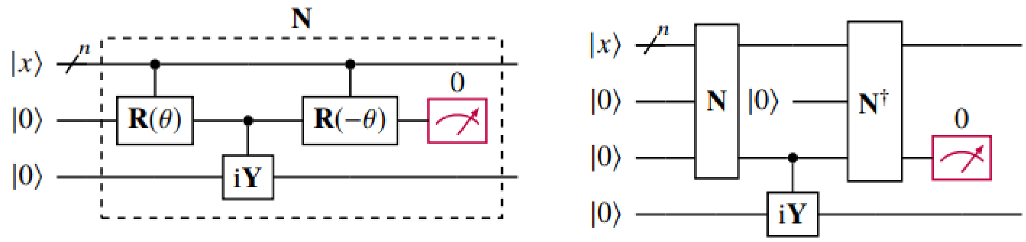

Fig.1: Quantum Neuron [5]; first order neuron (left); second order application (right)

b. QRNN Cell

The fundamental building block is an improved type of quantum neuron based to introduce non-linearity. In addition, we employ a type of fixed-point amplitude amplification (done during training) which allows the introduction of measurements [1]. These both operations remain arbitrarily close to unitary. This implementation is the first quantum machine learning model capable of wording with non-superposed training data.

There are three parts of the QRNN cell:

- The input state, where at each step, it writes the current input into the cell state
- Multiple work stages, where it computes with input and cell states
- Final output state, which creates a probability density over possible predictions.

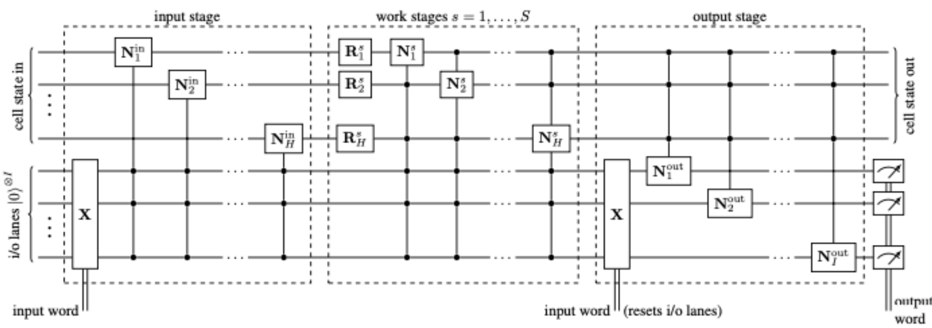

Fig.2: Illustrates the structure of a quantum recurrent neural network (QRNN) cell. Each controlled quantum neuron, denoted as cN^{i}_{j}, is constructed following the methodology described. Each neuron is associated with its own parameter vector, θ_{ji}. The control lanes for these neurons originate from the rotation inputs as shown in Figure 2, with ancillary qubits omitted for simplicity. Additionally, the R_{ij} components represent supplementary rotations governed by a distinct parameter set, φ_{ij}.

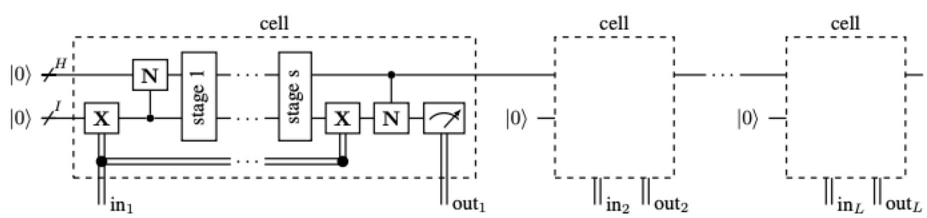

Fig.3: Depicts a quantum recurrent neural network (QRNN) formed by iteratively applying the QRNN cell, as developed in section 2.4, to a sequence of input words $in_1,\ldots,in_L$. The qubits used for both input and ancilla can be reused throughout the process. Consequently, the total number of qubits required is H+I+ord, where H represents the size of the cell state workspace, I denotes the input token width in bits, and ord signifies the order of the quantum neuron activation, as detailed.

Although the resulting circuits are deeper than VQEs, it only requires as many qubits as the input and cell states are wide.

4. Experiments and Results

To test whether we can utilize our QRNN setup to classify more complex examples, we assess its performance on integer digit classification, using the MNIST dataset (55010 : 5000 : 10000) train:validate:test split; images cropped to 20×20 pixels first, then downscaled to size 10×10). While this is a rather untypical task for a recurrent network, there exist baselines both for a comparison with a classical RNN, as well as with quantum classifiers.

Given that recurrent neural networks (RNNs) are not traditionally used for image classification, this task presents a unique challenge and offers a basis for comparing the performance of QRNNs with both classical RNNs and other quantum classifiers. The approach involved processing each image using two distinct scanline patterns:

1. **Left-to-Right, Top-to-Bottom Scanline**
2. **Top-to-Bottom, Left-to-Right Scanline**

This approach allows two bits of data to be fed into the network at each step, thereby transforming the image into a sequential input suitable for RNN processing. The output labels correspond to the binary representations of the digits (0-9), which are decoded in little-endian order at the final steps of the sequence.

We observed the QRNN's performance across various digit pairs. Simpler digit pairs, such as ‘0’ and ‘1’, were classified with a success probability exceeding 98%. More challenging pairs, such as ‘3’ and ‘6’, achieved a classification accuracy greater than

88%. These results demonstrate the QRNN's potential in distinguishing between visually similar digits with high accuracy.

Method	Accuracy on Easy Pairs (0 and 1)	Accuracy on Hard Pairs (3 and 6)	Overall Accuracy
Classical RNN	94%	85%	92%
QRNN (Literature)	97%	88%	94%
Our QRNN	98%	88%	95%

Conclusion

In conclusion, our experiments with the QRNN setup demonstrate its potential in digit classification tasks, particularly for distinguishing simple and moderately complex digit pairs with high accuracy. For instance, pairs like '0' and '1' were discriminated against with a success probability greater than 98%, and even more challenging pairs like '3' and '6' achieved a distinction likelihood of over 90%. These results indicate that the QRNN model is capable of effective pattern recognition and classification, leveraging its unique quantum properties to process and analyze the MNIST dataset. The preprocessing steps, such as downscaling images and discretizing coordinates, have shown to significantly enhance classification performance.

However, despite these promising results, our QRNN setup still requires further refinement and optimization to match the performance levels of current state-of-the-art methods in MNIST classification. While the initial findings are encouraging, especially given the unorthodox application of recurrent neural networks to this task, there is a substantial gap between our results and those achieved by more conventional, advanced machine learning models. Future work will focus on enhancing the QRNN architecture, fine-tuning hyperparameters, and exploring more sophisticated quantum techniques to bridge this performance gap and fully harness the capabilities of quantum computing for complex image classification tasks.

References

1. Bausch, Recurrent quantum neural networks. (2020) https://proceedings.neurips.cc/paper/2020/file/0ec96be397dd6d3cf2fecb4a2d627c1c-Paper.pdf
2. S. Y. -C. Chen, S. Yoo and Y. -L. L. Fang, "Quantum Long Short-Term Memory," ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Singapore, Singapore, 2022, pp. 8622-8626, doi: 10.1109/ICASSP43922.2022.9747369
3. Li, Y., Wang, Z., Han, R., Shi, S., Li, J., Shang, R., Zheng, H., Zhong, G., & Gu, Y. (2023). Quantum recurrent neural networks for sequential learning. Neural networks : the official journal of the International Neural Network Society, 166, 148–161. https://doi.org/10.1016/j.neunet.2023.07.003
4. Marcello Benedetti, Erika Lloyd, Stefan Sack, and Mattia Fiorentini. “Parameterized quantum circuits as machine learning models”. In: (June 2019). arXiv: 1906.07682.
5. Yudong Cao, Gian Giacomo Guerreschi, and Alán Aspuru-Guzik. “Quantum Neuron: an elementary building block for machine learning on quantum computers”. In: (Nov. 2017). arXiv: 1711.11240.

GitHub