A Novel Fully-Quantum LSTM: Design and Empirical Evaluation for Air Quality Index Prediction

by

DAVINDER SINGH

under the supervision of

Dr.Debajyoti Bera and Dr.Anubha Gupta

submitted

in partial fulfilment of the requirements for the degree of

MASTER OF TECHNOLOGY

to

COMPUTER SCIENCE AND ENGINEERING

INDRAPRASTHA INSTITUTE OF INFORMATION TECHNOLOGY DELHI

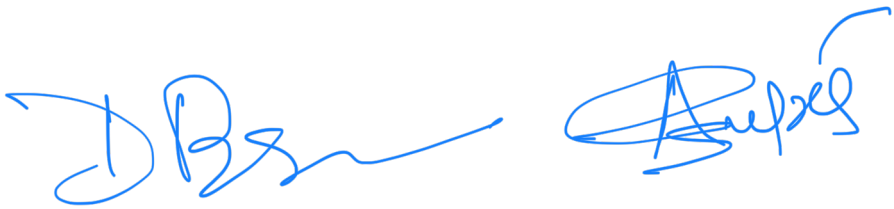
NEW DELHI- 110020

July, 2025

THESIS CERTIFICATE

This is to certify that the thesis titled "**A Novel Fully-Quantum LSTM: Design and Empirical Evaluation for Air Quality Index Prediction**", submitted by **Davinder Singh**, to the Indraprastha Institute of Information Technology Delhi, for the award of the degree of Master of Technology, is an original research work carried out by him under my supervision. In my opinion, the thesis has reached the standards fulfilling the requirements of the regulations relating to the degree.

The results contained in this thesis have not been submitted in part or full to any other university or institute for the award of any degree/diploma.



July, 2025

Dr. Anubha Gupta
Dr. Debajyoti Bera
Department of Computer Science and Engineering
Indraprastha Institute of Information Technology Delhi
New Delhi, 110020

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my M.Tech. thesis advisor, Dr. Debajyoti Bera, for his invaluable guidance and support throughout the course of this work. His insights and feedback played a crucial role in shaping this research.

I also wish to thank Dr. Anubha Gupta for her guidance on the classical machine learning aspects of this work, particularly in identifying appropriate evaluation metrics and methods to rigorously assess the model's performance.

I would like to acknowledge the contribution of Usnik Nath, who assisted in the evaluation of classical and hybrid-quantum Long Short-Term Memory models across various hyperparameter configurations, helping to arrive at the most optimal ones.

I extend my thanks to the faculty and staff of the Computer Science and Engineering Department at IIIT Delhi for providing a supportive academic environment and the necessary resources to carry out this research.

I am deeply grateful to my parents and younger brother for their unwavering support and encouragement throughout my academic journey. Their belief in me has been a continuous source of strength.

Lastly, I thank myself for staying committed through the challenges and for the consistent effort that went into bringing this work to completion.

Davinder Singh (MT23031)

Davinder Singh

(MT23031)

ABSTRACT

Air Quality Index (AQI) forecasting is essential for public health and environmental policy. While accurate, **traditional numerical weather prediction (NWP)** models demand vast computational resources. Recent trends have favoured deep learning data-driven approaches like **Long Short-Term Memory (LSTM)** networks, which model temporal dependencies in pollutant data. However, these methods typically require large datasets and a significant number of trainable parameters, making them inefficient and potentially overparameterised.

In contrast, **hybrid quantum-classical (HQC)** models have emerged, incorporating **parameterised quantum circuits (PQCs)** as core learning components while deferring operations such as memory retention and updates to classical sub-networks. Although these approaches reduce parameter counts and introduce some quantum mechanical properties, the presence of classical components means that quantum features like entanglement and superposition are not maintained throughout the network.

In this work, we explore a novel direction by investigating a **fully quantum Long Short-Term Memory (QLSTM)** architecture, a recurrent neural network constructed entirely from PQC layers. Unlike HQC models, our architecture performs all operations, including memory retention and updates, purely in the quantum domain. The model is implemented and simulated using PennyLane, and its learning and generalisation performance is benchmarked against classical and hybrid LSTM baselines.

Despite hardware limitations and noiseless simulation constraints, QLSTM demonstrates convergence and learning capability. Although it does not outperform its classical or hybrid counterparts in short-term AQI forecasting, it establishes the feasibility of full quantum recurrence in temporal modelling. We also explore optimisation strategies such as **RMSProp**, based on its successful use in previous hybrid models.

Our findings suggest that fully quantum recurrent networks are viable on near-term quantum simulators with limited qubit budgets. This work opens avenues for future research in circuit compression, noise-aware training, and theoretical analysis of quantum memory mechanisms. The proposed QLSTM thus lays the foundational groundwork for deeper integration of quantum computing into time series learning tasks.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

NOTATION

Symbol	Description
Quantum Computing Notation	
$\|0\rangle, \|1\rangle$	Standard basis states in Dirac notation
$\|\psi\rangle$	A general quantum state
$\langle\psi\|$	Dual (conjugate transpose) of $\|\psi\rangle$
$\langle\phi\|\psi\rangle$	Inner product between two quantum states
$\langle\mathcal{O}\rangle$	Expectation value of operator $\mathcal{O}$ with respect to a quantum state
X	Pauli-X (NOT) gate
H	Hadamard gate
$R_a(\theta)$	Rotation around the arbitrary axis a, where $a \in \{x, y, z\}$
CX	Controlled-X (CNOT) gate
$\sqrt{\text{SWAP}}$	Square root of SWAP gate
Classical ML / Neural Network Notation	
$\sigma(x)$	Sigmoid function: $\frac{1}{1+e^{-x}}$
$\tanh(x)$	Hyperbolic tangent function
$\odot$	Element-wise (Hadamard) product
RMSE	Root Mean Squared Error
R^2	Coefficient of Determination

CHAPTER 1

Introduction

Air Quality Index is a composite metric representing the concentration of various pollutants in the atmosphere. It is typically calculated as the maximum of sub-indices corresponding to individual pollutants, such as particulate matter (PM) of sizes 2.5μm and 10μm ($PM_{2.5}$ and PM_{10}), sulfur dioxide (SO_2), and carbon monoxide (CO).

The AQI plays a critical role in shaping environmental regulations and public health advisories. Exposure to elevated AQI levels has been strongly associated with an increase in hospital visits, prolonged respiratory illnesses, and reduced overall health quality. A study by Surit *et al.* (2023) observed a significant correlation between poor AQI and increased rates of respiratory ailments. Consequently, reliable AQI forecasting is vital for informed policymaking, medical preparedness, and regulatory enforcement—particularly in areas related to construction planning and vehicular pollution control.

1.1 Motivation

AQI levels fluctuate based on the concentration and movement of atmospheric pollutants, which are influenced by meteorological factors such as wind speed, direction, temperature, and pressure. Traditional NWP models use complex simulations of atmospheric physics to estimate pollutant transport and concentration. While these models offer high accuracy, they are **computationally intensive**, often requiring **supercomputers** and significant runtime—rendering them less practical for real-time or resource-constrained scenarios.

Researchers have increasingly turned to **data-driven methods** to overcome these limitations, especially **machine learning** and **deep learning models**. Among these, **Recurrent Neural Networks (RNNs)**, particularly LSTM models, have shown strong performance in capturing temporal dependencies in pollutant data. However, these models are typically **parameter-heavy** and demand large amounts of training data, which can hinder generalisation and make them less efficient in low-data environments.

This growing need for expressive and efficient models has led to an interest in **QML**, HQC models in particular, that integrate quantum circuits as trainable components within classical architectures. These models have shown promising results in reducing parameter counts while capturing richer feature correlations through **quantum entanglement** and **superposition**.

In this thesis, we take this one step further by investigating the feasibility of a QLSTM network, where every gate, memory cell, and transformation operates in the **quantum domain**. This approach explores whether a **purely quantum architecture** can learn and generalise in time-series tasks like AQI prediction, even when deployed on current **quantum simulators**.

1.2 Problem Statement

Hybrid quantum-classical (HQC) models have recently emerged as a promising direction in machine learning, particularly for reducing the number of trainable parameters while leveraging quantum properties such as superposition and entanglement. These models typically embed PQCs within classical architectures—using quantum subroutines for feature encoding or transformation—while deferring memory retention and optimisation to classical components.

However, this hybrid separation imposes a **fundamental limitation**: **quantum coherence is not preserved** throughout the entire network. Core memory operations, essential to recurrent networks, remain strictly classical. The absence of quantum analogues for memory transformations limits the exploration of **fully quantum temporal dynamics** and prevents **end-to-end entangled evolution** across time steps.

The central research problem addressed in this thesis is therefore twofold. First, we aim to investigate whether it is feasible to construct a QLSTM architecture, wherein all operations—including memory updates, gating mechanisms, and output computations—are implemented exclusively through **PQCs**. Second, we assess how such an architecture performs on a **real-world sequence modelling task**: short-term **Air Quality Index (AQI)** forecasting.

This thesis approaches the problem not only as a modelling challenge but also as an **engineering problem**—how to propagate memory across time steps, utilise it meaningfully in quantum computation for current outputs, and preserve **coherence across**

the full recurrent pipeline. To this end, we propose a **novel QLSTM design** that utilizes **ancilla qubits**, **entanglement-aware gate configurations** via $\sqrt{SWAP}$ gates, and **layered quantum logic** to implement memory propagation and state transitions.

While outperforming classical baselines is not the primary objective, this work rigorously evaluates the **feasibility and learning capability** of a **fully quantum recurrent model**. It lays foundational groundwork for future studies in **quantum memory representation**, **coherence-aware training**, and the architectural evolution of **quantum sequence models**.

1.3 Summary of Contributions

This thesis makes the following key contributions:

- **Proposal of a Fully Quantum LSTM (QLSTM) Architecture:** We design a novel recurrent neural network architecture that performs all core operations—input gating, memory retention, state update, and output transformation—purely in the quantum domain using PQCs.
- **Memory Propagation via Quantum Ancillas and $\sqrt{SWAP}$ Gates:** A core innovation of this work is the engineering solution for memory transfer across time steps using ancilla qubits and square-root SWAP (SQSWAP) gates, enabling temporal information flow without classical intervention.
- **End-to-End Quantum Pipeline Implementation:** We implement the QLSTM architecture entirely using the PennyLane framework and benchmark its feasibility, convergence, and learning performance on a short-term real-world time series forecasting task AQI prediction.
- **Comparative Evaluation with Classical and Hybrid Models:** We conduct an empirical study comparing the proposed QLSTM against classical LSTM and Hybrid Quantum-Classical (HQC) LSTM baselines. Our findings demonstrate that while QLSTM underperforms in raw metrics, it successfully converges and learns temporal dependencies, establishing its viability.
- **Exploration of Optimization Strategies in Quantum Training:** We study the performance impact of using RMSProp as the optimizer, motivated by its application in prior HQC literature, and show improved performance over Adam in the quantum training context.
- **Foundational Step Toward Fully Quantum Temporal Modelling :** We demonstrated the feasibility of maintaining a coherent quantum state through recurrent architecture for temporal modelling empirically, this work lays the groundwork for future research in quantum memory mechanisms, circuit compression, and noise-aware learning.

1.4 Organisation of the Thesis

This thesis is structured as follows. Chapter 2 provides a literature review of both statistical and deep learning-based approaches for AQI prediction, along with detailed discussions on the architectures of LSTM and Hybrid Quantum-Classical (HQC) LSTM models. Chapter 3 introduces the proposed fully quantum LSTM (QLSTM) model, explaining its design components, layered structure, and how the circuit scales in terms of qubit usage and depth. Chapter 4 describes the methodology, including data collection, preprocessing, and the training pipeline. Chapter 5 presents the results of the experiments, comparing the performance of the QLSTM with classical and hybrid models, and interprets the findings. Finally, Chapter 6 concludes the thesis and discusses future directions, such as circuit compression, noise-aware training, and quantum memory analysis.

CHAPTER 2

Background and Literature Review

This chapter provides the necessary background to understand the technical and domain-specific foundations of the thesis. It begins with an overview of quantum computing and its foundational concepts, followed by a discussion on quantum machine learning and parameterised quantum circuits. Subsequently, we introduce recurrent neural networks, with a focus on Long Short-Term Memory (LSTM) architectures, which serve as the classical baseline in this work. We then review recent developments in hybrid and fully quantum LSTM models. Finally, we survey existing approaches to Air Quality Index (AQI) forecasting and highlight the limitations that motivate our proposed quantum architecture.

2.1 Fundamentals of Quantum Computing

Richard Feynman noted in his seminal address Feynman (1982), simulating quantum physics on classical computers is fundamentally limited. Since quantum systems grow exponentially in complexity with the number of interacting particles and exhibit inherent randomness and nonlocality, classical computers cannot efficiently simulate such behaviours, as they cannot faithfully reproduce the whole quantum state space or the intricate entanglement between qubits. Feynman proposed that to simulate quantum phenomena effectively, one must build a quantum system, thus laying the conceptual groundwork for quantum computing.

Quantum computers operate on the principles of quantum mechanics, which govern the behaviour of systems at the subatomic scale. Unlike classical computers, where the fundamental unit of information is the **bit** (which can take a value of either 0 or 1), quantum computers use **quantum bits** or **qubits**.

A qubit exists in a complex vector space of dimension two, and its state is represented as a linear combination (superposition) of the computational basis states $|0\rangle$ and $|1\rangle$:

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle$$

where $\alpha, \beta \in \mathbb{C}$ and $|\alpha|^2 + |\beta|^2 = 1$. The quantities $|\alpha|^2$ and $|\beta|^2$ represent the probabilities of measuring the qubit in states $|0\rangle$ and $|1\rangle$, respectively.

For example, the state

$$|\psi\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$$

can be written in vector form as

$$\begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix},$$

which represents a qubit with equal probability (50%) of collapsing to $|0\rangle$ or $|1\rangle$ upon measurement.

This ability to exist in a superposition of states is one of the fundamental features that gives quantum computing a computational edge over classical computing. It allows quantum computers to simultaneously represent and manipulate multiple possible states, enabling parallelism at the level of quantum state space. Another critical phenomenon is entanglement, which allows qubits to exhibit strong correlations such that the state of one qubit instantaneously affects the state of another, regardless of spatial separation. When used together, superposition and entanglement enable quantum computers to process and encode information in ways that scale exponentially with the number of qubits, far beyond what classical systems can achieve.

Quantum computations are performed by applying **quantum gates**, which are reversible transformations represented by unitary matrices. These gates manipulate the state of qubits, allowing for controlled evolution of the quantum system. In the circuit model of quantum computing, qubits are depicted as wires, and gates act upon them sequentially to form quantum circuits. Each gate transforms the input quantum state into another valid state while preserving the overall probability amplitude norm.

Common single-qubit gates include the **Hadamard gate** (which creates superposition), **Pauli-X gate** (analogous to a classical NOT), and parameterized **rotation gates** that rotate qubit states around Bloch sphere axes. Multi-qubit gates, such as the **CNOT gate**, **SWAP**, and **square-root SWAP**, enable entanglement and conditional logic. The QLSTM architecture developed in this thesis relies on a subset of these gates, detailed in Table 2.1.

One final part of quantum computing is the use of **measurement operators**, which connect the quantum world to classical outcomes. In quantum computers, measurement is typically performed in the computational basis $\{|0\rangle, |1\rangle\}$. When a qubit in state

Table 2.1: Matrix representations of selected quantum gates used in this thesis.

Gate	**Matrix Representation**
Hadamard (H)	$\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$
Pauli-X (X)	$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$
Rotation-X ($R_x(\theta)$)	$\begin{bmatrix} \cos(\theta/2) & -i\sin(\theta/2) \\ -i\sin(\theta/2) & \cos(\theta/2) \end{bmatrix}$
Rotation-Y ($R_y(\theta)$)	$\begin{bmatrix} \cos(\theta/2) & -\sin(\theta/2) \\ \sin(\theta/2) & \cos(\theta/2) \end{bmatrix}$
Rotation-Z ($R_z(\theta)$)	$\begin{bmatrix} e^{-i\theta/2} & 0 \\ 0 & e^{i\theta/2} \end{bmatrix}$
Controlled-NOT (CNOT)	$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$
SWAP	$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$\sqrt{\text{SWAP}}$	$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1}{2}(1+i) & \frac{1}{2}(1-i) & 0 \\ 0 & \frac{1}{2}(1-i) & \frac{1}{2}(1+i) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle$ is measured, it collapses probabilistically to $|0\rangle$ with probability $|\alpha|^2$ and to $|1\rangle$ with probability $|\beta|^2$. This irreversible collapse extracts classical information from the quantum system which then can be postprocessed on a classical computer for final output.

2.2 Introduction to Quantum Machine Learning

Quantum Machine Learning (QML) is a niche and rapidly developing paradigm that aims to introduce quantum advantage to the field of machine learning by leveraging fundamental quantum phenomena such as superposition and entanglement. These principles allow quantum systems to encode and process large amounts of information in high-dimensional Hilbert spaces, enabling novel computational techniques for learning tasks.

There are two primary strategies for integrating quantum computing with machine learning:

1. **Quantum Kernel Methods**: In this approach, classical data is embedded into a high-dimensional quantum Hilbert space using a parameterised quantum circuit known as a *feature map*. This method is analogous to classical kernel techniques such as the Support Vector Machine (SVM). Specifically, the **Quantum Support Vector Machine (QSVM)** leverages quantum kernels to perform classification.

   In classical SVM, if data is not linearly separable, a kernel function (such as the radial basis function, or RBF) is used to map the data into a higher-dimensional space where a linear hyperplane can effectively separate the classes. Quantum kernel methods generalise this idea by using quantum circuits to implicitly compute kernel values of the form:

   $$K(x_i, x_j) = |\langle \phi(x_i) | \phi(x_j) \rangle|^2,$$

   where $\phi(x)$ is the quantum feature map applied to input x. The inner product is estimated using quantum measurements, allowing the kernel to capture complex relationships that may be infeasible classically.

2. **Variational Quantum Circuits (VQCs)**: These are hybrid quantum-classical models, sometimes referred to as *Quantum Neural Networks*, and were systematically studied by Cerezo *et al.* (2021). VQCs are based on the universality of parameterised quantum gates: as shown by Plesch and Brukner (2011), combinations of rotation and entangling gates can approximate any unitary operation to arbitrary precision also it was proved by Benedetti *et al.* (2019) that PQCs are capable of learning similar to classical neural network models and have remarkable expressive power, .

   Typically, a VQC consists of three main stages:
   - **Embedding Layer**: Classical input data is embedded into quantum states using parameterised rotations or amplitude encoding.

- **Parameterized Quantum Circuit (PQC) Layer(s)**: One or more layers of parameterised gates and entangling operations form the trainable core of the circuit.
- **Measurement Layer**: Quantum expectation values (typically of Pauli operators) are measured and returned to the classical computer.

As illustrated in Figure 2.1, the PQC layer used in this thesis is structured as follows: a layer of Hadamard gates is first applied to all qubits to place them in superposition. This is followed by one or more *strongly entangled layers*, each consisting of:

(a) A layer of universal single-qubit rotation gates parameterised by angles α, β, γ,

(b) A layer of `CNOT` gates connecting each qubit to its immediate neighbour,

(c) An additional `CNOT` gate connecting the topmost qubit to the bottommost qubit, forming a *circular entanglement pattern*.

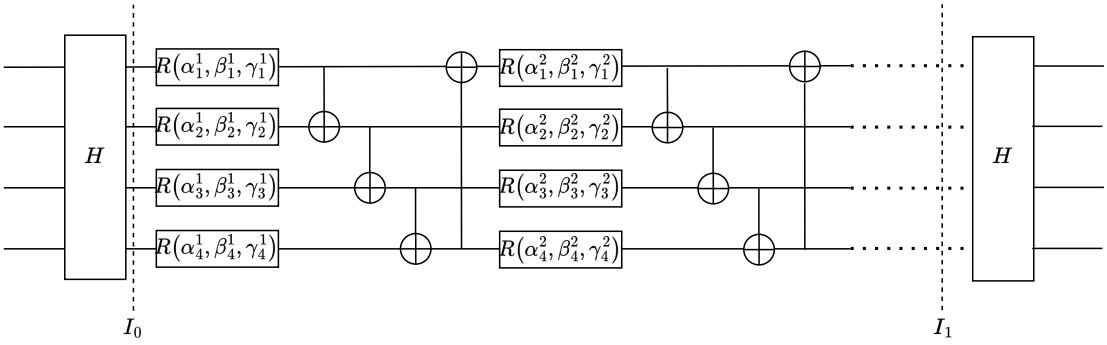


Figure 2.1: A basic Parameterised Quantum Circuit (PQC)

This sequence may be repeated multiple times to enhance circuit expressivity. Optionally, a final Hadamard layer is added to bring the qubits back from their entangled state into a computational basis suitable for measurement, as shown in Figure 2.2.

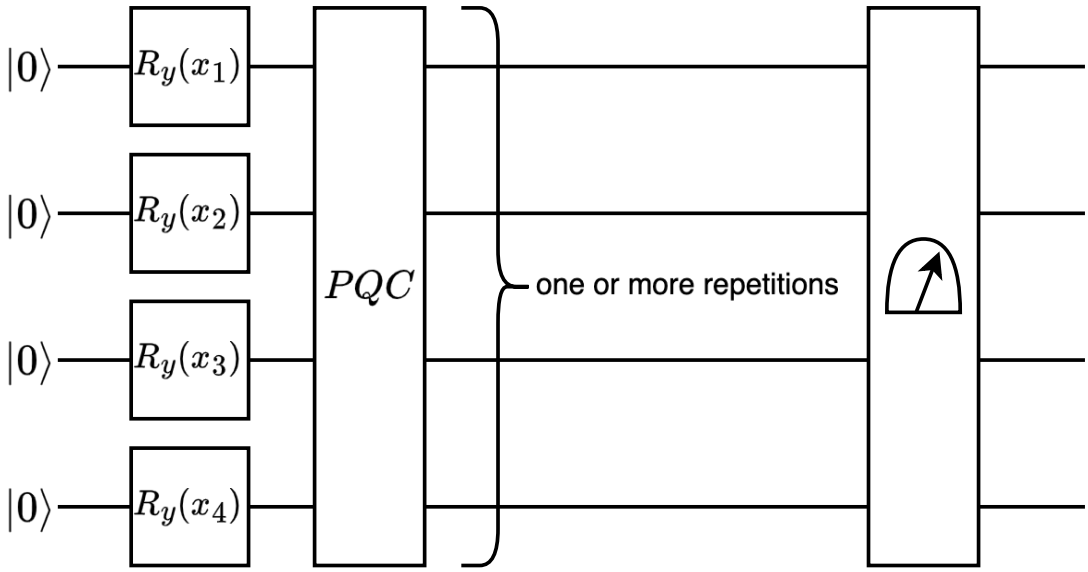


Figure 2.2: A variational quantum circuit with angle embedding using $R_y(x_i)$ rotations, followed by a parameterised quantum circuit (PQC) and measurement.

The entire circuit is trained using classical optimisation techniques, as illustrated in Figure 2.3. The VQC acts as a function approximator, with parameters adjusted to minimise a classical loss function computed from quantum measurement outcomes.

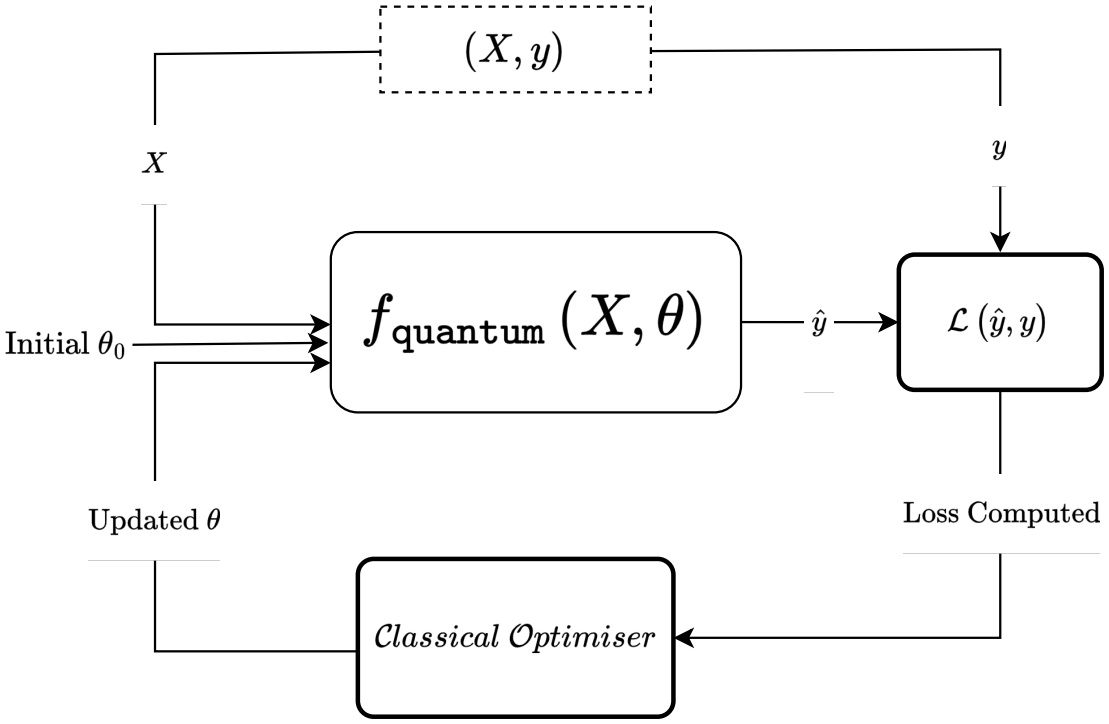


Figure 2.3: Hybrid quantum-classical workflow for training a Variational Quantum Circuit (VQC). The quantum model $f_{\text{quantum}}(X, \theta)$ produces predictions $\hat{y}$, the loss $\mathcal{L}(\hat{y}, y)$ is computed classically, and a classical optimizer updates the parameters θ.

VQCs form the foundation of many modern quantum machine learning architectures and are commonly used in quantum versions of supervised models, including quantum classifiers, quantum convolutional networks, and quantum recurrent networks.

Before moving to the next section, it is important to briefly discuss how quantum networks are trained.In classical neural networks, training is typically performed using **backpropagation**, where gradients of the loss function are computed with respect to the weights of the network through the application of the chain rule across layers.

In the quantum setting, however, computing gradients is less straightforward. While classical components like non-linear activations and matrix multiplications have well-defined derivatives, quantum circuits consist of unitary gate operations, and their gradients must be defined in a way that is compatible with quantum mechanics.

A widely used method for computing gradients in quantum circuits is the **parameter-shift rule**, introduced by Crooks (2019). This method allows for exact gradient com-

putation of expectation values with respect to gate parameters by evaluating the circuit at two shifted parameter values. For a parameterized gate $U(\theta)$, the gradient of the expectation value $\langle \hat{O} \rangle$ with respect to θ is given by:

$$\frac{\partial \langle \hat{O} \rangle}{\partial \theta} = \frac{1}{2} \left[\langle \hat{O} \rangle_{\theta + \frac{\pi}{2}} - \langle \hat{O} \rangle_{\theta - \frac{\pi}{2}} \right]$$

This rule enables gradient-based optimisation of quantum circuits using classical optimisers. Intuitively, for each parameterised gate in the circuit, the parameter-shift rule evaluates the circuit twice—once with the gate parameter shifted to $\theta + \frac{\pi}{2}$, and once with $\theta - \frac{\pi}{2}$. The difference between the two resulting expectation values is then used to estimate the effect of the parameter change, effectively yielding the gradient of the circuit output with respect to that parameter. The overall complexity of this procedure is $\mathcal{O}(2M)$, where M denotes the number of trainable parameters in the quantum circuit.

However, this technique does not apply to all types of quantum gates. For gates that do not satisfy the conditions required for parameter-shift (e.g., those not generated by simple Pauli operators), gradients are instead approximated using finite-difference methods—that is, by evaluating the circuit with increasingly small parameter increments.

By combining both techniques—exact gradients via the parameter-shift rule and numerical approximations via finite differences—we can perform end-to-end training of variational quantum models, including the QLSTM architecture proposed in this thesis.

2.3 Classical Long Short-Term Memory (LSTM) Network

Now that the fundamentals of quantum computing and quantum machine learning have been introduced, we now turn to the classical architecture used as a baseline in this thesis.

The Long Short-Term Memory (LSTM) network, introduced by Hochreiter and Schmidhuber (1997), is a type of Recurrent Neural Network (RNN) designed to model sequential and temporal dependencies in data. The term *recurrent* refers to the networks structure, where a hidden state is maintained and passed from one timestep to the next, allowing the model to capture both short-term and long-term dependencies across time.

An LSTM network is typically unrolled for n timesteps, where n corresponds to the length of the input time series.

Figure 2.4 shows a schematic diagram of a vanilla LSTM cell, which is repeated at each timestep during unrolling.

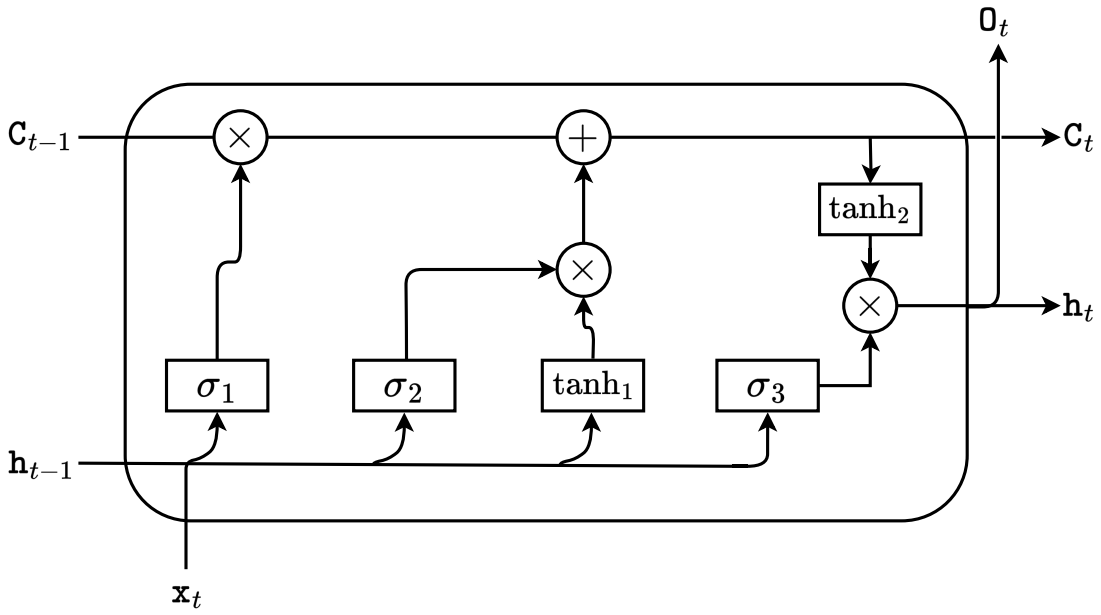


Figure 2.4: Schematic diagram of a vanilla LSTM cell with gated operations controlling memory and output flow.

The vanilla LSTM cell contains five trainable components, represented as $\sigma_{1:3}$ and $\tanh_{1:2}$, which correspond to fully connected layers followed by non-linear activation functions. The `cell state` (`C`) serves as the long-term memory, responsible for carrying context across multiple timesteps. The `hidden state` (`h`), often referred to as the short-term memory, propagates the immediate output from one timestep to the next.

These components collectively control how much information is retained, forgotten, or passed forward at each step. The following set of equations describes the internal functioning of the LSTM cell:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{2.1}$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{2.2}$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{2.3}$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \tag{2.4}$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{2.5}$$

$$h_t = o_t \odot \tanh(C_t) \tag{2.6}$$

These equations represent the forget gate (f_t), input gate (i_t), candidate memory content ($\tilde{C}_t$), updated cell state (C_t), output gate (o_t), and hidden state (h_t). Together, they control how information flows through the network at each timestep.

- f_t: Forget gate — decides how much of the past cell state C_{t-1} to retain.
- i_t: Input gate — controls how much of the new candidate memory $\tilde{C}_t$ to add.
- $\tilde{C}_t$: Candidate memory — new content generated from current input and hidden state.
- C_t: Updated cell state — combines past memory (via f_t) and new input (via i_t).
- o_t: Output gate — regulates what part of the cell state affects the output.
- h_t: Hidden state — short-term output passed to the next timestep.

2.4 Related Work: Hybrid Quantum LSTM Networks

Classical LSTM networks perform remarkably well in the domain of sequence modelling. Unlike simple recurrent neural networks, which suffer from issues such as the **vanishing gradient** problem — where, as the network depth or number of timesteps increases, the gradient values diminish progressively during backpropagation, eventually becoming negligible — LSTMs mitigate this through their gating mechanisms. However, as more recurrent units are added to capture longer-term dependencies, the model becomes increasingly parameterised, which can make training more challenging and computationally expensive.

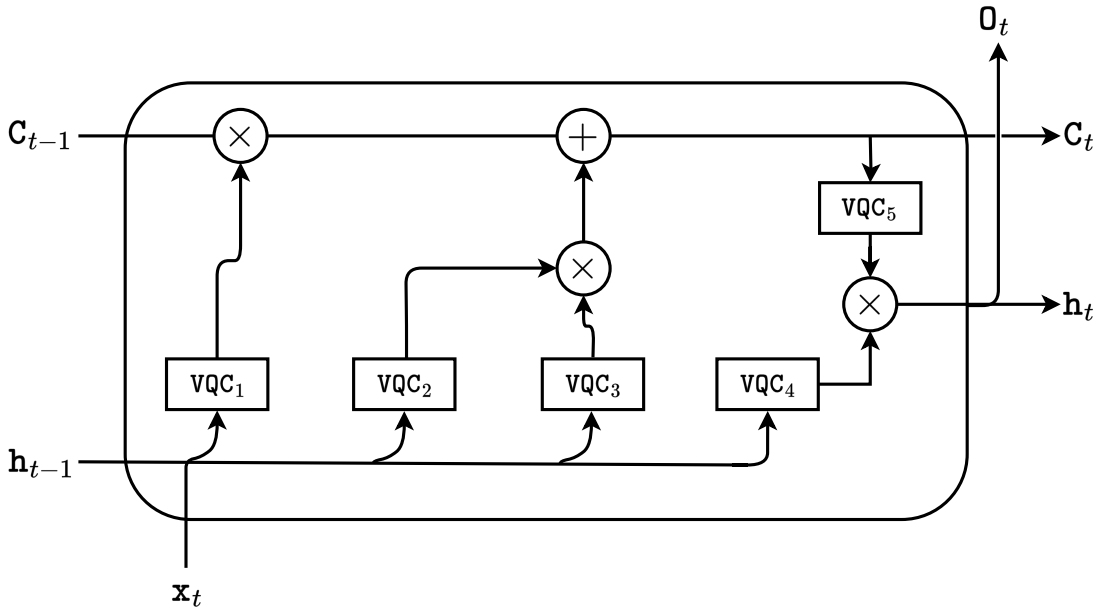


Figure 2.5: Schematic diagram of a hybrid quantum-classical LSTM cell, in which the classical gating mechanisms are replaced by variational quantum circuits (VQC_1 to VQC_5), responsible for regulating memory and output flow.

The Hybrid Quantum LSTM (HLSTM), introduced by Chen *et al.* (2022), is a hybrid quantum-classical architecture in which the trainable components of the classical LSTM are replaced with variational quantum circuits (VQCs); see Figure 2.2 for an illustration of the VQC structure. Figure 2.5 depicts the modified structure of the LSTM network following the inclusion of VQCs.

Each `VQC` unit comprises a linear layer that maps the input features to a dimension suitable for the number of qubits used. This mapped input is then processed by the VQC, which is executed either on a simulator or a quantum backend. The outputs from each qubit are measured, producing values in the range $[-1, 1]$, which are subsequently passed through another linear layer to project them back into the appropriate domain. The remainder of the LSTM cell functions identically to its classical counterpart.

Chen *et al.* (2022) benchmarked their HLSTM model against a classical LSTM with an equivalent number of trainable parameters and observed improved performance on certain tasks, such as modelling periodic functions like $\sin(x)$ and damped harmonic oscillators. However, one limitation of this approach is that the quantum circuits are confined to individual gates and do not maintain quantum entanglement across timesteps, which potentially restricts the expressive power of the network. This limitation motivates the development of a fully quantum LSTM architecture, as proposed in this thesis.

2.5 Application Domain: AQI Forecasting and Challenges

Air Quality Index (AQI) forecasting, unlike the task of modelling simple periodic functions, is a considerably more complex challenge. It requires integrating information from multiple pollutant components such as $PM_{2.5}$, PM_{10}, NO_2, SO_2, CO, and O_3. Accurate AQI prediction depends significantly on the length of the context window, the transformation of input features, and how effectively inter-variable relationships are captured.

In practice, AQI time series are non-linear, multi-dimensional, and often non-stationary. They exhibit seasonal and diurnal patterns, as well as abrupt fluctuations caused by weather anomalies, local emission events, and socio-political factors. Traditionally, AQI prediction has relied on numerical models that simulate atmospheric dynamics using supercomputers. While such models can provide accurate forecasts, they are computationally intensive and often fail to capture the true chaotic behaviour of real-world air quality in a timely manner. To address this, many researchers have shifted their focus to short-term AQI forecasting—ranging from a few hours to a couple of days—based on historical data patterns.

Ravindiran *et al.* (2023) employed various classical machine learning methods grounded in statistical algorithms, including **Random Forest**, **Light Gradient Boosting Machine (LGBM)**, **AdaBoost**, and **XGBoost**. Their experiments were conducted on a dataset comprising 1920 instances, derived from open-source AQI records collected between January 2017 and December 2022 in Visakhapatnam, a city in Andhra Pradesh, India. Their key findings include:

- **Key Pollutants Influencing AQI:** $PM_{2.5}$ and carbon monoxide (CO) showed a strong temporal correlation with AQI throughout the year, particularly during the winter. Their trends closely mirrored AQI fluctuations, highlighting their critical role in AQI computation (Fig. 5; Table S4).
- **Photochemical Ozone Formation and VOC Contribution:** Ground-level ozone levels peaked during winter due to temperature inversion effects that trap volatile organic compounds (VOCs) and NO_x near the surface. These conditions promote photochemical ozone formation, with major sources including vehicular emissions, thermal power plants, and heavy industries.
- **Industrial Resumption and Pollution Spike Post-Lockdown:** A noticeable increase in air pollutants in 2021 compared to 2020 was observed, attributed to the resurgence of industrial activity following COVID-19 lockdowns (Fig. S5). This

underscores the substantial contribution of industrial sources to ambient pollution.

Their study demonstrated that classical statistical models achieved high accuracy, with R^2 scores approaching 99%. However, they did not test generalisation across other Indian cities, nor did they explore deep learning approaches in detail. This left open the opportunity to apply neural architectures for AQI forecasting.

Building upon this foundation, Jiao *et al.* (2019) investigated AQI prediction using a multivariate LSTM architecture applied to air quality data from Shanghai. Their work provides valuable insight into both the strengths and limitations of classical deep learning methods for environmental forecasting:

- **Correlation of Meteorological and Pollutant Parameters with AQI:** The study reported strong positive correlations between AQI and pollutants such as $PM_{2.5}$ (0.95), PM_{10} (0.89), CO (0.81), SO_2 (0.70), and NO_2 (0.64). In contrast, meteorological variables such as wind direction and temperature exhibited weaker correlations (below 0.4). This finding underscores the dominant influence of pollutant concentrations in AQI determination, validating their importance in predictive modelling.
- **Multivariate Time Series Forecasting using LSTM:** A nine-dimensional feature set comprising pollutants and meteorological variables was used to train the LSTM. The model achieved a low mean squared error (MSE) of 10.95, demonstrating its capability to learn complex temporal dependencies and deliver accurate short-term AQI forecasts.

In this thesis, AQI forecasting serves as a real-world benchmark to assess the performance of the proposed quantum models. The selection of AQI as the application domain is motivated by its societal relevance, the inherent difficulty of the task, and the presence of cyclic as well as chaotic patterns that make it a suitable candidate for exploring the strengths of quantum-enhanced temporal architectures.

2.6 Summary

This chapter provided a comprehensive overview of the foundational concepts and related work relevant to this thesis. It began with the fundamentals of quantum computing, highlighting the principles of superposition, entanglement, and quantum measurement. Building upon this, the chapter introduced Quantum Machine Learning, focusing particularly on Variational Quantum Circuits (VQCs) and it's learning component Param-

eterised Quantum Circuit (PQC) as a key architectural component for quantum neural networks.

The chapter then reviewed the classical Long Short-Term Memory (LSTM) network, detailing its structure and advantages in modelling temporal sequences. This was followed by a discussion of hybrid quantum-classical LSTM architectures, where classical gating mechanisms are replaced by VQCs. While these hybrid models demonstrate promise, they fall short in maintaining quantum coherence and entanglement across timesteps, which motivates the fully quantum approach explored in this thesis.

Finally, the chapter outlined the challenges and complexities associated with AQI forecasting as a real-world application domain. Prior works in both statistical and deep learning-based approaches were discussed, demonstrating the need for models capable of capturing long-term dependencies and multivariate non-linear relationships.

The next chapter introduces the proposed fully quantum LSTM architecture, detailing its design, theoretical underpinnings, and training pipeline.

CHAPTER 3

Pure Quantum Model Architecture

As discussed in the motivation section, the hybrid LSTM architecture proposed by Chen *et al.* (2022) confines quantum effects—such as superposition and entanglement—within isolated VQC blocks (see Figure 2.5). This architectural constraint raises an intriguing question: can we design a fully quantum variant of LSTM where quantum coherence is preserved across time steps?

This chapter presents the proposed architecture addressing that question. We first describe the guiding intuition behind the design, followed by a detailed explanation of the circuit structure, qubit usage, and information flow within the model.

3.1 Limitations of Existing QRNNs

The Quantum Recurrent Neural Network (QRNN) introduced by Li *et al.* (2023) served as the initial inspiration for our model. As illustrated in Figure 3.1, the architecture follows a simple many-to-one structure, taking N inputs and producing a single output. Parameterised Quantum Circuits (PQCs) are connected sequentially to form a recurrent structure, where each PQC$_i$ receives an input X_i. A final measurement layer is applied only at the end of the sequence to obtain the prediction, denoted by $\hat{y}_5$ in the figure. The authors benchmarked this model on standard periodic functions and demonstrated that it converges effectively.

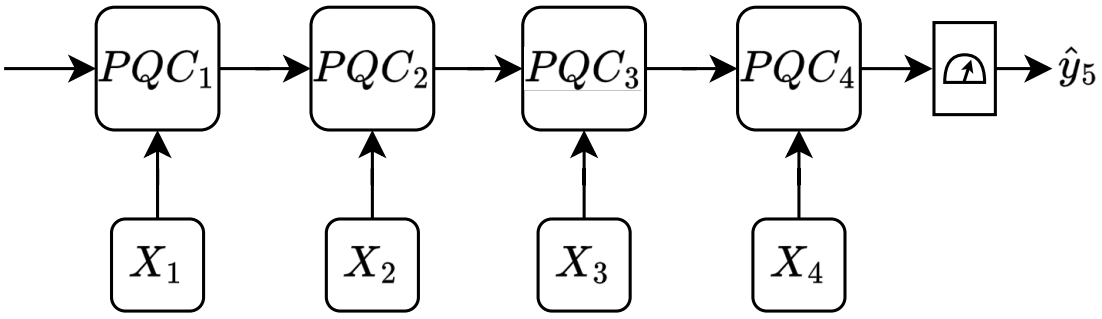


Figure 3.1: A simple quantum recurrent network consisting of four PQCs chained sequentially, ending in a measurement layer that produces the final prediction $\hat{y}_5$ after processing inputs X_i, where the subscript i denotes the input at the i^{th} time step.

The limitation with the QRNN approach becomes evident when compared to classical LSTMs. In an LSTM, each time step produces an output, which is used during **Back-propagation Through Time (BPTT)**–a modified version of standard backpropagation designed for training sequence and recurrent models. As illustrated in Figure 3.2, the output at each time step contributes to both the global and local optimisation of the network through gradient updates.

In contrast, the QRNN architecture does not allow such training flexibility. Modifying the network to produce outputs at each time step would require introducing measurement layers after every PQC, which would collapse the quantum state and disrupt entanglement and superposition. As a result, the network would lose its fully quantum character and effectively become hybrid.

Nonetheless, without this capability, the QRNN remains restricted to a many-to-one structure and is unsuitable for a broad class of sequence modelling tasks—such as next-word prediction in language models—where intermediate outputs are essential.

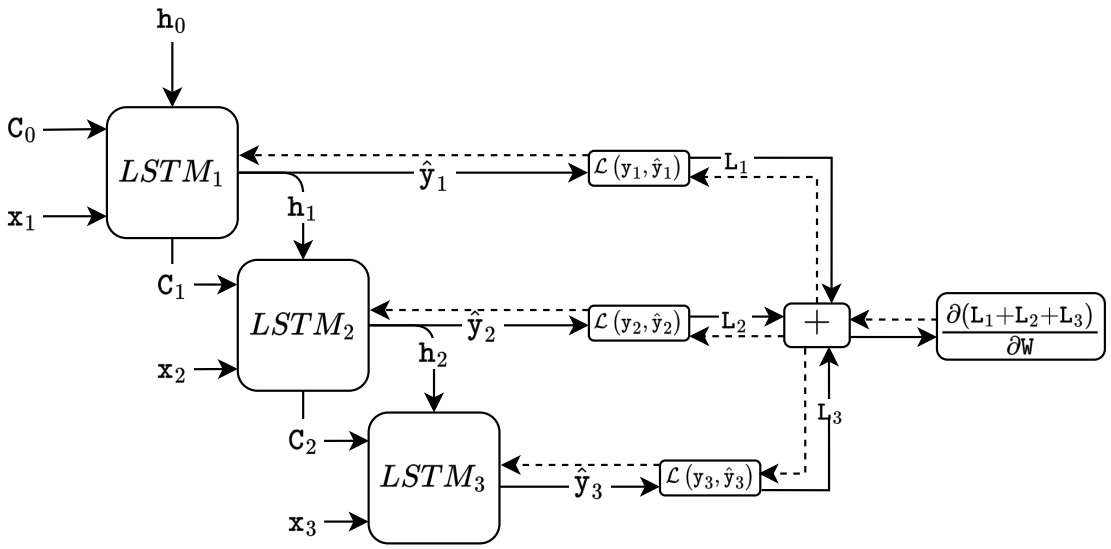


Figure 3.2: Illustration of Backpropagation Through Time (BPTT) in an LSTM network unrolled across three time steps. At each step, the LSTM produces an output $\hat{y}_t$, which is compared to the ground truth y_t to compute the loss $\mathcal{L}(y_t, \hat{y}_t)$. The total loss is obtained by summing the individual time-step losses, and the gradient of this total loss is computed with respect to the shared parameters W (shown by dashed arrows). All LSTM units share the same parameters across time.

Furthermore, even if **partial measurement** were considered at each step to extract intermediate outputs, the utility would be minimal. Partial measurements in quantum circuits yield binary outcomes (0 or 1) according to the probability distribution encoded in the qubit state. These binary values are inadequate for many prediction tasks, par-

ticularly regression problems or language modelling, which require richer continuous representations.

Additionally, introducing measurements mid-sequence collapses the quantum state, thereby destroying entanglement and coherence that the network relies on the qubits that were measured. As a result, incorporating intermediate measurements would not only degrade the quantum nature of the model but also render the extracted outputs insufficient for meaningful learning.

Hence, QRNNs in their current form remain unsuitable for general sequence modelling tasks without compromising quantum coherence.

3.2 Motivation from Classical LSTM

To address the limitations discussed above, we reinterpreted the LSTM architecture by drawing it in a form analogous to a quantum circuit—that is, as a sequence of operations (gates) applied over time along persistent data lines. In Figure 3.3, the input, forget, update, and output gates are depicted as sequential operations. For example, the parameters $\mathtt{W}_c$ and $\mathtt{b}_c$ denote the weight and bias matrices of the candidate (update) gate, which produces the candidate cell state used in updating the LSTM's long-term memory (cell state).

Each gate acts on the current input $\mathtt{x}_t$ and the previous hidden state $\mathtt{h}_{t-1}$, and after each operation, these variables are propagated forward to the next gate—similar to how qubit lines persist across multiple gates in a quantum circuit.

This topological perspective offered an intuitive foundation for constructing an equivalent architecture in the quantum domain—one that could preserve coherence and enable sequential processing of quantum data.

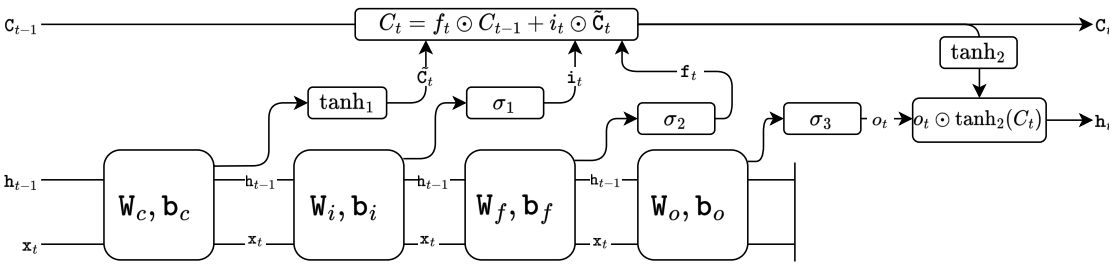


Figure 3.3: Classical LSTM cell redrawn in a topologically sorted, circuit-inspired format.

While this reinterpretation informs the structure of a quantum recurrent network, it does not resolve a key challenge: how can quantum memory be propagated from one recurrent unit to another without inducing decoherence, while still preserving the quantum advantage? Addressing this question was central to the design of our proposed architecture.

3.3 Proposed Quantum LSTM Architecture

With the necessary background established, including a review of existing approaches and their limitations, we are now prepared to introduce the proposed Quantum Long Short-Term Memory (QLSTM) architecture. Building upon the motivation drawn from the classical LSTMs circuit-inspired topological form, the architecture leverages quantum principles such as superposition and entanglement throughout the recurrent structure. Furthermore, the proposed design adopts a many-to-many configuration by default, but it can be readily adapted for many-to-one scenarios as well.

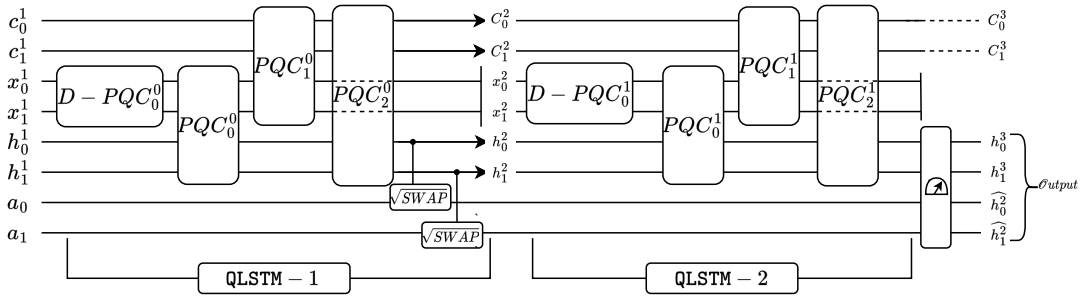


Figure 3.4: Proposed Quantum LSTM (QLSTM) architecture, unrolled across two time steps. Each QLSTM block consists of a data re-uploading PQC (D-PQC_0^t) followed by multiple sequential PQCs that operate on input (x^t), hidden state (h^t), and cell state (c^t) qubits. Ancilla qubits (a_0, a_1) and $\sqrt{\text{SWAP}}$ gates are employed to preserve coherence and propagate memory without measurement. Final outputs are extracted using a non-destructive measurement strategy, enabling a many-to-many output scheme.

The architecture illustrated in Figure 3.4 consists of several key components, each of which is discussed in the sections that follow.Among the most important are: (1) the use of data re-uploading Parameterised Quantum Circuits (PQCs), which are specialised quantum circuits designed to repeatedly encode input data into the quantum system at multiple layers; and (2) the memory propagation mechanism, which employs square root of SWAP ($\sqrt{\text{SWAP}}$) gates along with ancilla qubits (i.e., additional helper qubits)

to transfer quantum information across time steps without measurement. Together, these components enable the proposed architecture to preserve coherence, support recurrent behaviour, and maintain entanglement across layers.

3.4 Data Reuploading Parameterised Quantum Circuits

The data reuploading strategy enables quantum circuits, even those with a small number of qubits, to represent complex functions by reintroducing classical input features at multiple points within the circuit. Originally proposed as a method to build universal quantum classifiers using repeated single-qubit rotations and classical subroutines Pérez-Salinas *et al.* (2020), this approach significantly enhances the expressive power of Parameterised Quantum Circuits (PQCs). Each reuploading step involves embedding the same input data via parameterised quantum gates (e.g., $R_Y(x_i)$, $R_Z(x_i)$ or $R_X(x_i)$), with generic rotation gates to introduce parameterization $R(\alpha, \beta, \gamma)$ followed by learnable unitary transformations.

In the context of our QLSTM model, we adopt multi-layered data reuploading PQCs (denoted as DPQC blocks) at each time step to encode both current inputs and preserve feature richness across the depth of the network. This repeated data encoding compensates for limited qubit depth and provides greater flexibility in learning temporal and nonlinear patterns in sequential data. A schematic illustration of the DPQC block used in our architecture is shown in Figure 3.5.

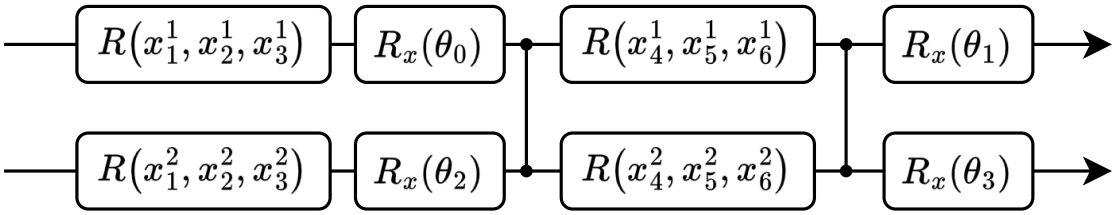


Figure 3.5: Quantum RNN

3.5 Memory Propagation and Layer Composition

A central challenge in designing a quantum analogue of the LSTM lies in propagating memory across time steps without performing intermediate measurements, which would collapse the quantum state and disrupt coherence. In classical LSTMs, mem-

ory is explicitly maintained through hidden and cell states passed from one unit to the next. To achieve a similar mechanism in our QLSTM, we leverage a combination of ancilla qubits and $\sqrt{\text{SWAP}}$ gates to facilitate coherent state transfer between successive QLSTM blocks.

A key architectural constraint arises from the need to use intermediate values—such as the cell state and output—from the current QLSTM block to compute the final output and pass information forward. However, accessing these values directly through measurement would destructively collapse the quantum state, thereby altering it irreversibly. Once altered, the quantum state can no longer be reliably forwarded to the next QLSTM unit without first re-encoding it classically, which would effectively revert the model to a hybrid form. To address this, we employ the $\sqrt{\text{SWAP}}$ operation.

The SWAP gate in quantum computing functions similarly to its classical counterpart: it exchanges the states of two qubits. The $\sqrt{\text{SWAP}}$ gate is defined by taking the matrix square root of the unitary representing the SWAP operation. Unlike a full SWAP, the $\sqrt{\text{SWAP}}$ gate only partially exchanges the quantum state, effectively creating a shallow entanglement between the two qubits. When applied twice in sequence, the effect is equivalent to a complete SWAP operation.

To store the entangled hidden state without disturbing the original qubit, we introduce additional qubits—commonly referred to as *ancilla qubits*—initialised in the $|0\rangle$ state. After the QLSTM operation at each time step, $\sqrt{\text{SWAP}}$ gates are applied between the hidden-state qubits (h_i) and their corresponding ancillae. This procedure creates a weakly entangled copy of the hidden state in the ancilla qubits without collapsing the original state. These ancillae are then measured only at the final step, allowing us to extract the outputs in a many-to-many manner while preserving quantum coherence during the intermediate computations.

Although partial measurements have been proposed in theory as a means to extract intermediate information from quantum circuits, in practice they are incompatible with the requirements of coherence and entanglement preservation. Our architecture avoids this issue entirely by introducing a measurement-free mechanism for memory propagation using ancilla qubits and $\sqrt{\text{SWAP}}$ gates.

An additional advantage of our memory propagation design is its scalability to deeper recurrent architectures. Let N denote the number of qubits representing the hidden state, M the number used for the cell state, and P the number used for input encoding

at each time step. For a QLSTM with L layers, an extra set of N ancilla qubits is required for each of the upper $L-1$ layers to enable hidden state transfer. This results in a total qubit requirement given by:

$$Q_{\text{total}} = M + P + N \times L$$

This formulation highlights that the **model scales linearly with the number of layers**, while keeping the total qubit overhead manageable—making it viable for near-term quantum hardware with modest qubit capacities.

3.6 Measurement and Post-processing Layers

To preserve quantum coherence and entanglement throughout the sequence, all measurements in the proposed QLSTM architecture are deferred until the final time step. This design decision avoids mid-sequence collapse of the quantum state, allowing the network to maintain superposition and quantum correlations across the entire temporal window.

At the final time step, the ancilla qubits—which have accumulated the weakly entangled hidden states over time—are measured in the computational basis. These measurement outcomes serve as the output of the quantum model, effectively encoding the temporal evolution of the system in a quantum-coherent manner.

The measured binary outputs are then passed to a classical post-processing layer. In our implementation, we utilise a fully connected (dense) linear layer that maps the observed bitstrings to the target output domain. This hybrid interface enables tasks such as classification or regression while preserving the fully quantum nature of the recurrent computation itself.

Delaying the measurement until the final step is essential not only for retaining coherence but also for enabling the model to function as a many-to-many architecture, producing a prediction at each time step without disrupting future computations.

3.7 Comparison with Classical and Hybrid LSTMs

To contextualise the architectural advancements of our QLSTM, we compare it with both classical Long Short-Term Memory (LSTM) networks and existing hybrid Quantum-Classical LSTM (HQLSTM) models.

- **Classical LSTM:** Composed entirely of classical operations, these networks use matrix multiplications and nonlinearities (e.g., sigmoid, tanh) to process sequences. Memory is explicitly retained via hidden and cell states, allowing seamless many-to-many recurrence.
- **Hybrid LSTM (e.g., HQLSTM Chen *et al.* (2022)):** These models replace classical gate operations with Parameterised Quantum Circuits (PQCs) but still rely on intermediate measurement at each time step to extract outputs. This breaks quantum coherence across the sequence, making the network effectively block-wise quantum and limiting its ability to leverage entanglement over time.
- **Recent Fully Quantum LSTM Li *et al.* (2025):** A more recent architecture also aims to preserve quantum characteristics by avoiding full classical conversion at each time step. However, it partitions the quantum state space across time by reducing the number of propagating qubits (e.g., $n \rightarrow n/4$) as layers progress. Additionally, partial measurements are employed to extract intermediate outputs, which inherently destroy entanglement and limit coherence. This results in diminishing quantum resources with depth, posing challenges for scalability and full sequence-wide coherence. The model is implemented using the QPanda framework Dou *et al.* (2022).
- **Proposed QLSTM (Ours):** Our architecture avoids both intermediate measurement and qubit partitioning. Instead, it uses ancilla qubits and $\sqrt{\text{SWAP}}$ operations to transfer quantum information forward across time steps without collapsing the wavefunction. This preserves coherence across the entire sequence and allows for genuine many-to-many prediction in a fully quantum manner, with scalability in depth and time.

The following table summarises key architectural differences:

Model	**Intermediate Measurement**	**Quantum Coherence**	**Many-to-Many**
Classical LSTM	N/A	✗	✓
Hybrid LSTM	✓	✗(block-wise only)	Limited
QPanda QLSTM	✓(partial)	✗(degrades with depth)	Limited
Proposed QLSTM	✗	✓(sequence-wide)	✓

Table 3.1: Architectural comparison across classical, hybrid, and fully quantum LSTM variants.

As the table illustrates, our design is the only one to retain quantum coherence across time steps without requiring any mid-sequence measurements or qubit reduction. This

makes it a strong candidate for scalable, fully quantum sequential modelling—without reverting to hybrid compromises.

3.8 Summary

In this chapter, we presented a fully quantum variant of the Long Short-Term Memory (LSTM) network—termed the Quantum LSTM (QLSTM)—designed to preserve quantum coherence while supporting many-to-many sequential tasks. Motivated by a topological reinterpretation of classical LSTMs as gate-like sequences, we proposed a circuit-compatible quantum formulation that naturally maps to the quantum computing paradigm.

We began by analysing limitations in prior QRNN and hybrid QLSTM architectures, including their reliance on intermediate measurements and destructive readouts, which limit coherence and scalability. We then introduced a memory propagation mechanism that utilises ancilla qubits and $\sqrt{\text{SWAP}}$ gates to transfer hidden quantum states forward in time without measurement-induced collapse.

A key advantage of our QLSTM is its ability to maintain coherence throughout the entire sequence. Unlike earlier architectures—such as the recent QGRU/QLSTM model implemented using QPanda, which relies on qubit partitioning and partial measurement—our design enables consistent quantum information flow without reducing the effective qubit space over time.

The use of data reuploading PQCs further enhances the expressivity of each recurrent unit, compensating for shallow circuit depth and limited qubit counts typical of near-term quantum devices. The resulting model is scalable, fully quantum, and capable of performing many-to-many predictions without reverting to hybrid forms.

The next chapter describes the data preparation pipeline and training methodology used to evaluate this architecture on real-world time-series forecasting tasks.

CHAPTER 4

Data Gathering and Training Pipeline

This chapter outlines the data preparation and training pipeline used for evaluating the proposed QLSTM model. As illustrated in Figure 4.1, the pipeline encompasses dataset acquisition, standardisation, windowing, batching, and a modular training dispatch system compatible with fully quantum, hybrid, and classical backends.

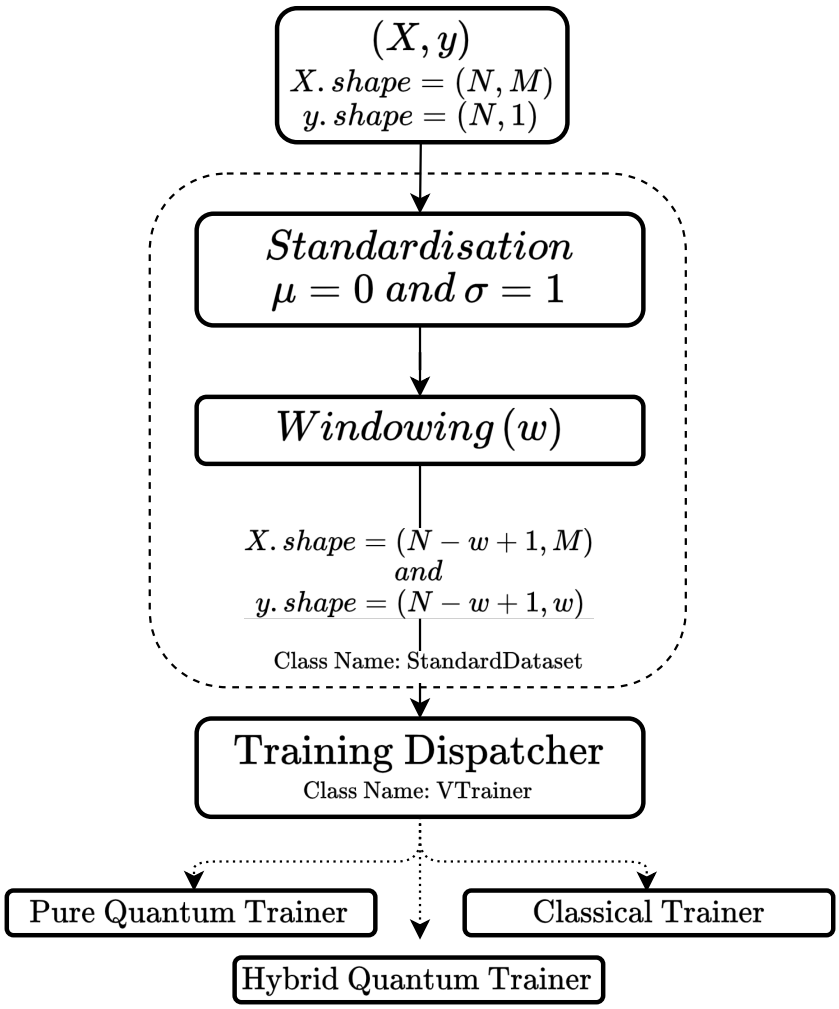


Figure 4.1: End-to-end training pipeline, including preprocessing (standardisation and windowing), modular dispatch logic, and trainer specialisation for quantum, hybrid, or classical architectures.

The sections that follow describe each stage of the pipeline in detail, beginning with data acquisition and feature selection, followed by preprocessing and formatting steps required to adapt the dataset for use with LSTM-based models.

4.1 Dataset Description

The AQI data used in this study were obtained from publicly available records provided by the Central Pollution Control Board (CPCB) of India[1]. The dataset includes the following features: `Datetime`, `PM2.5`, `PM10`, `NO`, `NO2`, `NOx`, `NH3`, `CO`, `SO2`, `O3`, `Benzene`, `Toluene`, `Xylene`, and `AQI`.

Given the high dimensionality of the dataset, feature selection was necessary to reduce complexity and improve model performance. A correlation heatmap, shown in Figure 4.2, was computed to visualise linear dependencies between the variables. Based on their strong correlation with `AQI`, we selected a subset of pollutants—`PM2.5`, `PM10`, `NO`, `NO2`, `NOx`, `NH3`, and `CO`—for use as model inputs.

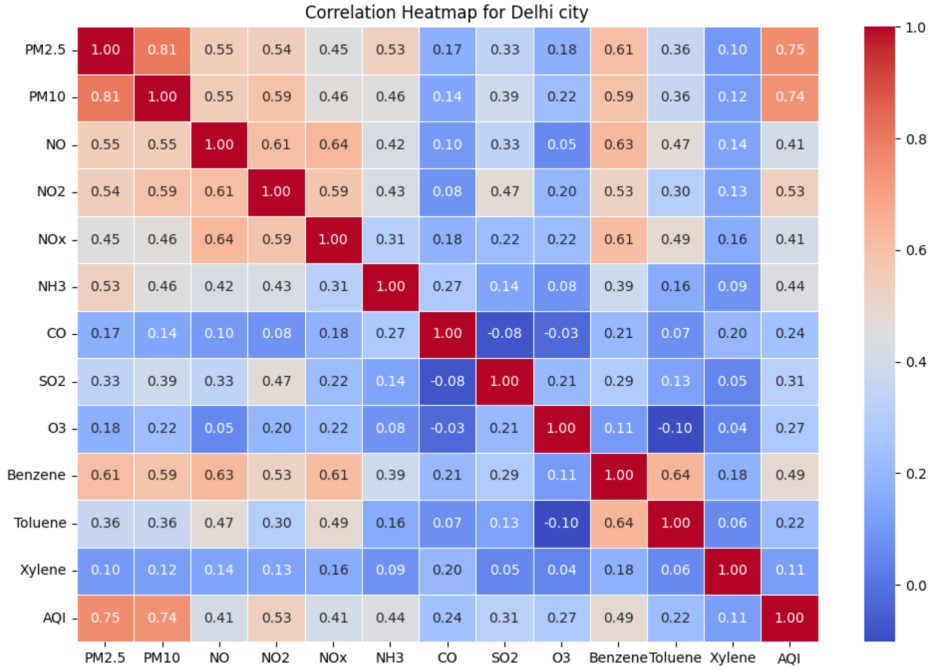


Figure 4.2: Correlation heatmap of air pollutants with AQI for Delhi. Features with the strongest correlations (e.g., PM2.5, PM10, NOx, NO2, NH3, CO) were selected for model input.

This study focused exclusively on data from Delhi.

4.2 Platform and Tools

All quantum simulations and experiments were conducted using **PennyLane** Bergholm *et al.* (2022), a cross-platform library for hybrid quantum-classical machine learning.

[1] `https://airquality.cpcb.gov.in/AQI_India/`

Classical deep learning baselines and hybrid model components were implemented using **PyTorch** and **Keras**, while data preprocessing and transformation were handled using **NumPy** and **Pandas**.

4.3 Preprocessing and Windowing

As illustrated in Figure 4.1, the dataset was standardised before being fed into the model. Standardisation ensures that all features are scaled to have zero mean and unit variance, which improves numerical stability and accelerates convergence. The standardisation transformation is applied using:

$$F(x) = \frac{x - \mu}{\sigma}$$

where x is the original value, μ is the feature-wise mean, and σ is the feature-wise standard deviation computed over the training data.

For all experiments, we fixed the context window size to 3 time steps and configured the models to predict AQI values for the subsequent 2 steps in the future. This many-to-many setup was selected to evaluate the models' ability to perform short-horizon temporal forecasting, while maintaining consistent conditions across classical, hybrid, and fully quantum variants. These parameters were held constant throughout all model runs to ensure a fair and reproducible comparison.

The dataset was split into 70% training, 20% validation, and 10% testing sets using `train_test_split` from Scikit-learn Pedregosa *et al.* (2018), with shuffling disabled to preserve temporal order, as required for time-series regression.

Following standardisation, the data was transformed using a sliding window approach. A fixed-length window of T consecutive time steps was used as input to the model. Depending on the task configuration, the output could be a single target value (many-to-one) or a sequence (many-to-many). This transformation reformulates the time series into a supervised learning problem suitable for recurrent neural architectures.

Finally, the windowed data was grouped into batches for efficient training. These batched tensors were passed through a modular dispatch system, which routed them to the appropriate classical, hybrid, or quantum training module.

4.4 Training Strategy

Different training strategies were employed for the three types of models evaluated:

- **Classical LSTM:** Implemented using Keras, the model was trained with the Adam optimiser using standard LSTM layers provided by the framework.
- **Hybrid Model:** Built using PyTorch and PennyLane, the hybrid LSTM replaces some classical transformations with parameterised quantum circuits (PQCs). Training leveraged PennyLanes autograd interface for PyTorch and used the Adam optimiser.
- **Proposed QLSTM:** Constructed entirely within PennyLane and wrapped as a PyTorch-compatible module. While initial experiments used the Adam optimiser, switching to **RMSprop** (as suggested in Ruder (2017) and adopted by Chen *et al.* (2022)) led to notable improvements in convergence and stability.

For evaluation, standard regression metrics were used. Mean Squared Error (MSE) and Root Mean Squared Error (RMSE) served as training losses, while R^2 (coefficient of determination) was used to measure the explainability and generalisation of the trained models.

4.5 Summary

This chapter presented the end-to-end data processing and training methodology adopted for evaluating the QLSTM model. Beginning with data acquisition from CPCB, we performed correlation-based feature selection, applied standardisation, and employed a sliding window approach for temporal structuring. We then described our modular pipeline for dispatching preprocessed batches to quantum, hybrid, or classical backends.

The next chapter details the results obtained from training each model, along with quantitative and visual performance evaluations on real-world AQI forecasting tasks.

CHAPTER 5

Results and Their Interpretation

This chapter presents the empirical findings from our experiments across three model variants: a classical LSTM baseline, a hybrid quantum-classical model, and a fully quantum recurrent model. The goal is not only to benchmark predictive accuracy but also to investigate the feasibility and learning capability of quantum architectures in real-world forecasting tasks.

5.1 Performance Comparison Overview

Table 5.1 provides a consolidated comparison of the three models — classical LSTM, hybrid quantum-classical, and fully quantum — in terms of parameter count, training duration, and performance metrics. The results correspond to a prediction horizon of 2 future steps using a context window of size 2. The table summarises training, validation, and test scores based on Root Mean Squared Error (RMSE) and R^2 coefficient.

Model	**Params**	**Epochs**	**Train RMSE**	**Train** R^2	**Val RMSE**	**Val** R^2	**Test** R^2
Classical	22,664	15	0.1042	0.9891	0.0479	0.9978	0.9978
Hybrid	1,506	15	0.1122	0.9874	0.0608	0.9965	0.9965
Pure	446	12	0.4609	0.7876	0.4700	0.7895	0.7895

Table 5.1: Performance summary across classical, hybrid, and fully quantum models.

5.2 Model-wise Analysis and Interpretation

5.2.1 Classical LSTM Baseline

The classical LSTM model, serving as the upper performance benchmark, achieved the highest accuracy across all metrics. With a large parameter count (22,664), the model demonstrates excellent convergence and generalisation. The validation RMSE remains lower than the training RMSE, possibly due to standardisation and dropout regularisation — a behaviour often seen in well-tuned deep learning models.

5.2.2 Hybrid Quantum-Classical Model

The hybrid model manages to perform comparably to the classical LSTM, despite having just 1,506 trainable parameters. The R^2 scores across training, validation, and test datasets remain very close to the classical baseline. This indicates that the quantum subcomponents embedded within the LSTM architecture are capable of learning useful temporal patterns while drastically reducing parameter overhead.

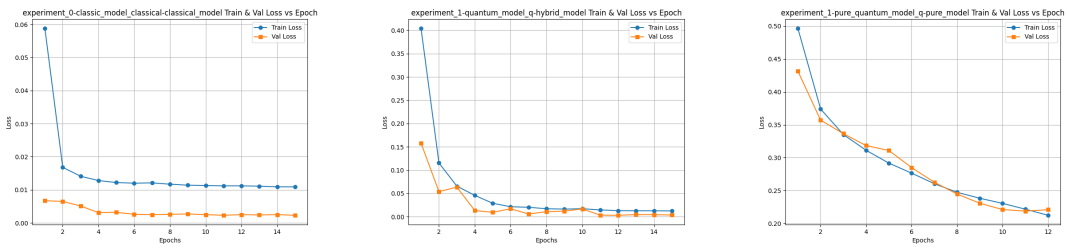
5.2.3 Pure Quantum Model

The pure quantum model was limited to 12 epochs due to **simulator runtime constraints** and used only 446 parameters — the fewest among all models. Despite these limitations, it still attained a test R^2 score of approximately 0.79. Although lower than the other two models, this result confirms that fully quantum models are capable of learning meaningful temporal representations. The performance suggests potential for further improvement through deeper quantum layers, hardware execution, or refined gradient strategies.

In summary, this comparative analysis demonstrates that while classical and hybrid models are currently more accurate, the fully quantum model exhibits non-trivial learning ability even in a constrained simulation environment. It thus represents a meaningful step toward practical quantum sequence learning systems.

5.3 Visual Analysis

5.3.1 Loss Curves

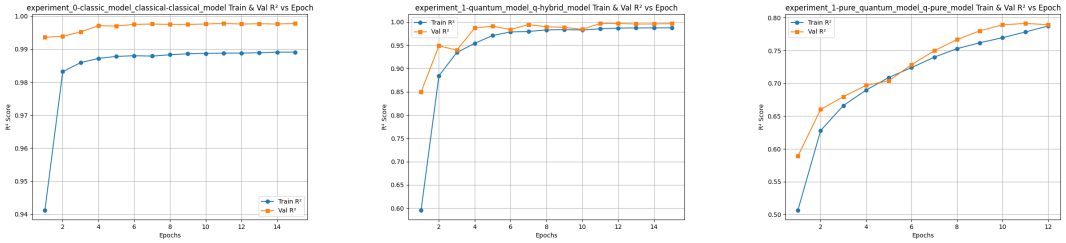


(a) Classical (b) Hybrid (c) Pure Quantum

Figure 5.1: Training and Validation Loss vs Epoch

As shown in Figure 5.1, the classical model consistently achieves low RMSE loss values across both training and validation sets. The slightly lower validation loss compared to training loss suggests mild underfitting, likely due to strong regularisation. This is expected given the use of standardised inputs and indicates stable convergence. The hybrid model, despite having significantly fewer parameters, closely mirrors the classical loss trajectory, demonstrating effective learning capability with reduced model complexity. In contrast, the pure quantum model exhibits a similarly steep and consistent decline in training loss. The validation loss closely tracks the training loss, suggesting that the model generalises well without signs of overfitting or underfitting. A small divergence at the final epochs is observed, which may be attributed to the noise characteristics and numerical artefacts introduced by quantum simulation backends.

5.3.2 R^2 Performance

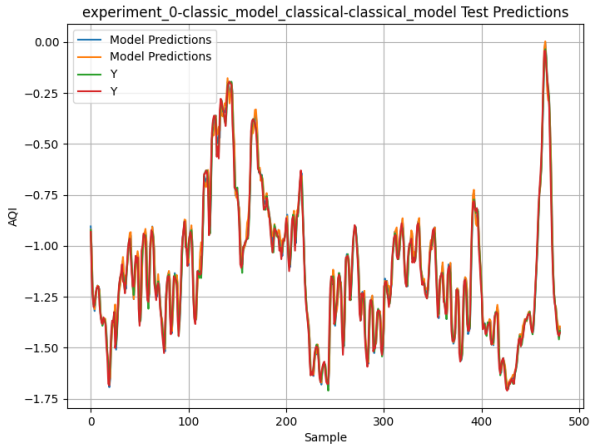


(a) Classical (b) Hybrid (c) Pure Quantum

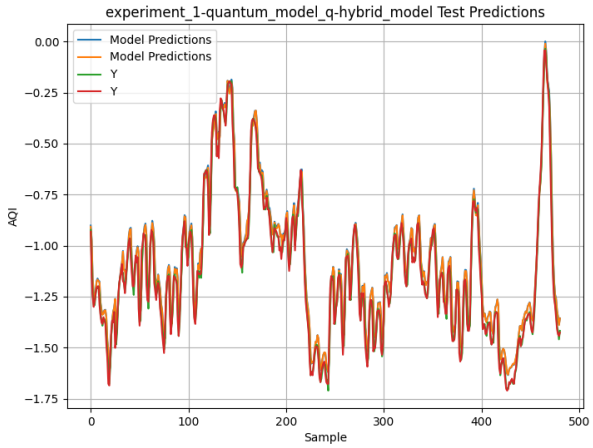
Figure 5.2: Training and validation R^2 scores across the classical, hybrid, and pure quantum models.

In Figure 5.2, the classical model consistently achieves high R^2 values throughout training and validation, indicating excellent generalisation. The hybrid model, while using significantly fewer parameters, closely follows the classical performance curve. In contrast, the pure quantum model exhibits more variance and a lower peak, plateauing around 0.79, which, despite being modest, suggests that the model is learning meaningful temporal dependencies given its constraints and the fact that it is running in a simulated environment.

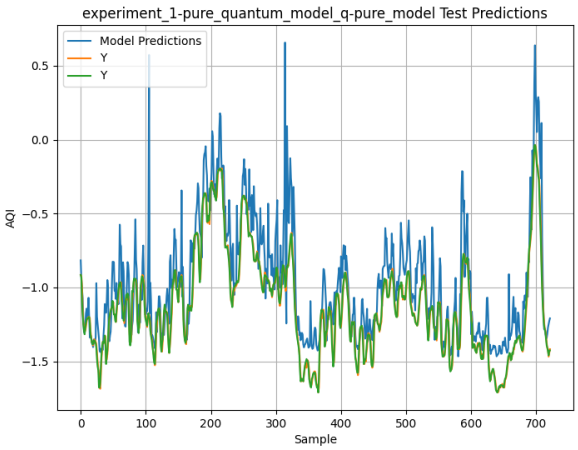
5.3.3 Prediction vs Actual Comparison



(a) Classical



(b) Hybrid



(c) Pure Quantum

Figure 5.3: Test Predictions vs Actual Values

The plots in Figure 5.3 compare the prediction performance of all three models. The classical model, with a high R^2 score, closely tracks the ground truth across the test

samples, reflecting strong predictive accuracy and generalisation. The hybrid model, despite having significantly fewer parameters, performs comparably to the classical baseline. This reinforces its capacity to leverage quantum submodules effectively for pattern extraction. The pure quantum model, while exhibiting occasional spikes and noise due to simulator limitations, still manages to capture the overall temporal trend. Despite having the smallest parameter count, it learns to approximate a complex, non-linear function such as AQI. This result highlights the potential of fully quantum models and underscores the need for continued research into their scalability and robustness.

5.4 Discussion

The classical and hybrid models achieved near-optimal regression scores, with the hybrid model demonstrating remarkable parameter efficiency. Despite having significantly fewer trainable parameters, the hybrid architecture retained predictive performance close to that of the classical baseline, underscoring the potential of quantum components in practical learning tasks.

Although the fully quantum model underperforms in absolute terms, its ability to learn meaningful temporal patterns validates the feasibility of quantum sequence modelling. The observed learning dynamics suggest that performance can be further enhanced through deeper quantum layers, coherence-aware training techniques, and improved circuit architectures tailored for quantum information flow.

5.5 Summary

This chapter provided a comparative evaluation of the classical, hybrid, and pure quantum models through both quantitative metrics and visual interpretations. While classical architectures continue to dominate in predictive performance, the hybrid model achieves a strong balance between accuracy and model complexity. Most notably, the fully quantum model—despite its limitations—demonstrates non-trivial learning capabilities, thereby establishing a foundational benchmark and motivating future advancements in quantum recurrent neural networks.

CHAPTER 6

Conclusion and Future Work

Every revolutionary idea - in science, politics, art, or whatever - seems to evoke three stages of reaction:

1. It's completely impossible.

2. It's possible, but it's not worth doing.

3. I said it was a good idea all along.

— Arthur C. Clarke

6.1 Summary of Work

This thesis explored the design, implementation, and evaluation of a fully quantum recurrent neural network — a pure Quantum Long Short-Term Memory (QLSTM) model applied to real-world time-series forecasting tasks. Building upon the structural analogy between classical RNNs and quantum circuits, we proposed a novel architecture that employs parameterised quantum circuits, ancilla-assisted memory propagation, and coherence-preserving mechanisms to model temporal dependencies without mid-sequence measurements.

The model was benchmarked alongside classical and hybrid counterparts on Air Quality Index (AQI) prediction using publicly available data. All models were evaluated in a consistent many-to-many setup, using a context window of 3 steps to predict the AQI for the subsequent 2 time steps. This uniform configuration ensured a fair and reproducible comparison.

6.2 Key Insights and Contributions

- Introduced a fully quantum many-to-many recurrent architecture using ancilla qubits and $\sqrt{\text{SWAP}}$ gates, enabling quantum coherence across time steps without intermediate measurement.
- Demonstrated the feasibility of training a fully quantum LSTM model using real-world data with minimal parameter count and under tight simulation constraints.

- Compared classical, hybrid, and quantum variants through empirical evaluations, showing that the proposed quantum model can learn non-trivial temporal dependencies.
- Developed a modular training and evaluation pipeline compatible with all three model variants, facilitating reproducible experimentation and future extensibility.
- Preliminary experiments showed that while classical LSTM maintained high performance with reduced parameter counts, hybrid models began to degrade under similar constraints — motivating the exploration of fully quantum architectures.

6.3 Limitations of the Study

While the results affirm the viability of pure quantum recurrent models, the following limitations remain:

- The quantum model was trained under strict runtime limits, which constrained both the training duration and the depth of the network.
- All quantum operations were executed on classical simulators, limiting scalability and failing to capture hardware-specific challenges such as noise and decoherence.
- Only basic quantum gradient descent (e.g., parameter shift rule) was used; advanced optimisation techniques such as momentum or adaptive learning (e.g. AdamW) could not be employed due to time and hardware constraints.
- The Lightning simulator — although faster — does not support backpropagation for the proposed architecture, requiring use of the default simulator and further increasing runtime.
- Limited support for backpropagation via the shift-rule across generic quantum gates restricted experimentation with alternate memory propagation schemes and more complex architectures.

6.4 Future Directions

This work lays the foundation for a new class of quantum recurrent models. Several promising directions for future research include:

- Deployment and benchmarking on real quantum hardware to evaluate performance under practical conditions, including gate noise, limited qubit coherence times, and measurement errors.
- Investigation of quantum memory representations, such as delay-line-inspired circuits or entangled memory banks, for more expressive and persistent temporal encoding.

- Development of coherence-aware training methods that explicitly account for circuit depth, decoherence thresholds, and entanglement-preserving operations.
- Exploration of model compression and quantum transfer learning to enable efficient generalisation while minimising qubit and parameter requirements.
- Theoretical analysis of the proposed architecture, including formal proofs regarding its memory retention properties and expressiveness over time-evolving sequences.

6.5 Final Remarks

This thesis explored the architectural and practical feasibility of a fully quantum recurrent model. While the results do not yet surpass classical counterparts, they demonstrate a clear signal: quantum models can learn, generalise, and extract temporal patterns from real-world data using only quantum computational elements.

As with any emerging technology, the field of quantum machine learning is progressing through the inevitable stages of resistance and gradual acceptance. This work aligns with what Arthur C. Clarke famously identified as the second stage of every revolutionary idea: *"It's possible, but it's not worth doing."* The model presented here makes that possibility concrete — showing that not only is it possible, but also that it holds value and potential as quantum hardware advances.

REFERENCES

1. **Benedetti, M.**, **E. Lloyd**, **S. Sack**, and **M. Fiorentini** (2019). Parameterized quantum circuits as machine learning models. **4**(4), 043001. ISSN 2058-9565. URL http://arxiv.org/abs/1906.07682.

2. **Bergholm, V.**, **J. Izaac**, **M. Schuld**, **C. Gogolin**, **S. Ahmed**, **V. Ajith**, **M. S. Alam**, **G. Alonso-Linaje**, **B. AkashNarayanan**, **A. Asadi**, **J. M. Arrazola**, **U. Azad**, **S. Banning**, **C. Blank**, **T. R. Bromley**, **B. A. Cordier**, **J. Ceroni**, **A. Delgado**, **O. D. Matteo**, **A. Dusko**, **T. Garg**, **D. Guala**, **A. Hayes**, **R. Hill**, **A. Ijaz**, **T. Isacsson**, **D. Ittah**, **S. Jahangiri**, **P. Jain**, **E. Jiang**, **A. Khandelwal**, **K. Kottmann**, **R. A. Lang**, **C. Lee**, **T. Loke**, **A. Lowe**, **K. McKiernan**, **J. J. Meyer**, **J. A. Montañez-Barrera**, **R. Moyard**, **Z. Niu**, **L. J. O'Riordan**, **S. Oud**, **A. Panigrahi**, **C.-Y. Park**, **D. Polatajko**, **N. Quesada**, **C. Roberts**, **N. Sá**, **I. Schoch**, **B. Shi**, **S. Shu**, **S. Sim**, **A. Singh**, **I. Strandberg**, **J. Soni**, **A. Száva**, **S. Thabet**, **R. A. Vargas-Hernández**, **T. Vincent**, **N. Vitucci**, **M. Weber**, **D. Wierichs**, **R. Wiersema**, **M. Willmann**, **V. Wong**, **S. Zhang**, and **N. Killoran** (2022). PennyLane: Automatic differentiation of hybrid quantum-classical computations. URL http://arxiv.org/abs/1811.04968.

3. **Cerezo, M.**, **A. Arrasmith**, **R. Babbush**, **S. C. Benjamin**, **S. Endo**, **K. Fujii**, **J. R. McClean**, **K. Mitarai**, **X. Yuan**, **L. Cincio**, and **P. J. Coles** (2021). Variational quantum algorithms. *Nature Reviews Physics*, **3**(9), 625644. ISSN 2522-5820. URL http://dx.doi.org/10.1038/s42254-021-00348-9.

4. **Chen, S. Y.-C.**, **S. Yoo**, and **Y.-L. L. Fang**, Quantum long short-term memory. *In ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2022.

5. **Crooks, G. E.** (2019). Gradients of parameterized quantum gates using the parameter-shift rule and gate decomposition. URL http://arxiv.org/abs/1905.13311.

6. **Dou, M.**, **T. Zou**, **Y. Fang**, **J. Wang**, **D. Zhao**, **L. Yu**, **B. Chen**, **W. Guo**, **Y. Li**, **Z. Chen**, and **G. Guo** (2022). Qpanda: high-performance quantum computing framework for multiple application scenarios. URL https://arxiv.org/abs/2212.14201.

7. **Feynman, R. P.** (1982). Simulating physics with computers | international journal of theoretical physics. URL https://link.springer.com/article/10.1007/BF02650179.

8. **Hochreiter, S.** and **J. Schmidhuber** (1997). Long short-term memory. **9**(8), 1735–1780. ISSN 0899-7667. URL https://doi.org/10.1162/neco.1997.9.8.1735.

9. **Jiao, Y.**, **Z. Wang**, and **Y. Zhang**, Prediction of air quality index based on LSTM. *In 2019 IEEE 8th Joint International Information Technology and Artificial Intelligence Conference (ITAIC)*. 2019. URL https://ieeexplore.ieee.org/document/8785602.

10. **Li, Y.**, **Z. Wang**, **R. Han**, **S. Shi**, **J. Li**, **R. Shang**, **H. Zheng**, **G. Zhong**, and **Y. Gu** (2023). Quantum recurrent neural networks for sequential learning. URL https://arxiv.org/abs/2302.03244.

11. **Li, Y.**, **Z. Wang**, **R. Xing**, **C. Shao**, **S. Shi**, **J. Li**, **G. Zhong**, and **Y. Gu** (2025). Quantum gated recurrent neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **47**(4), 2493–2504.

12. **Pedregosa, F.**, **G. Varoquaux**, **A. Gramfort**, **V. Michel**, **B. Thirion**, **O. Grisel**, **M. Blondel**, **A. Müller**, **J. Nothman**, **G. Louppe**, **P. Prettenhofer**, **R. Weiss**, **V. Dubourg**, **J. Vanderplas**, **A. Passos**, **D. Cournapeau**, **M. Brucher**, **M. Perrot**, and **Édouard Duchesnay** (2018). Scikit-learn: Machine learning in python. URL https://arxiv.org/abs/1201.0490.

13. **Plesch, M.** and **Brukner** (2011). Quantum-state preparation with universal gate decompositions. *Physical Review A*, **83**(3). ISSN 1094-1622. URL http://dx.doi.org/10.1103/PhysRevA.83.032302.

14. **Pérez-Salinas, A.**, **A. Cervera-Lierta**, **E. Gil-Fuster**, and **J. I. Latorre** (2020). Data re-uploading for a universal quantum classifier. **4**, 226. ISSN 2521-327X. URL http://arxiv.org/abs/1907.02085.

15. **Ravindiran, G.**, **G. Hayder**, **K. Kanagarathinam**, **A. Alagumalai**, and **C. Sonne** (2023). Air quality prediction by machine learning models: A predictive study on the indian coastal city of visakhapatnam. **338**, 139518. ISSN 0045-6535. URL https://www.sciencedirect.com/science/article/pii/S004565352301785X.

16. **Ruder, S.** (2017). An overview of gradient descent optimization algorithms. URL http://arxiv.org/abs/1609.04747.

17. **Surit, P.**, **W. Wongtanasarasin**, **C. Boonnag**, and **B. Wittayachamnankul** (2023). Association between air quality index and effects on emergency department visits for acute respiratory and cardiovascular diseases. **18**(11), e0294107. ISSN 1932-6203. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10653395/.