Nile University

School of Information Technology and Computer Science

Program of Computer Science.

Heatwave Forecasting Using Quantum Machine Learning

Senior Project II

Submitted in Partial Fulfilment of the Requirements

For the Bachelor's Degree in Information Technology and Computer Science

Submitted by:

SaifEldeen Khaled Emera Abdelmohsen	211001638
Muhammed Megahed Ali	211001594
Islam Nasr Atwan	211001735
Bishoy Ashraf Halim	211001882
Eyad Essam Elsanory	211002042
Abdullah Tarek Abdellatif	211002119

Supervised by:

Dr. Ahmed El-Mahdy

&

Dr. Marwa Sorour

GIZA – EGYPT

SPRING 2025

Abstract

Heatwaves pose serious risks to public health, infrastructure, and economic stability, making accurate and scalable forecasting methods essential in climate resilience planning. In this study, we explore the integration of quantum computing with classical Long Short-Term Memory (LSTM) architectures by evaluating the potential of Quantum Long Short-Term Memory (QLSTM) models for improving heatwave forecast accuracy. The QLSTM replaces classical neural network gates within the LSTM structure with Variational Quantum Circuits (VQCs), potentially leveraging quantum advantages such as parameter efficiency and richer feature mapping. Meteorological data from multiple global stations—including Cairo, Riyadh, Phoenix, Scottsdale, London, and Barcelona —is collected, processed, and used for heatwave forecasting under two configurations: (1) predicting the next day using three days of historical data, and (2) predicting the next three days using seven days of input data. Based on preliminary evaluations, the second configuration is further explored, as it demonstrated stable performance and practical relevance for operational heatwave forecasting. The performances of classical LSTM and QLTSM are investigated for both configurations for the data collected on the mentioned cities to evaluate the generalizability of our approach under varying temperature dynamics and climate volatility. IBM's quantum hardware simulator is used for the QLSTM, and the model is tested under realistic noise settings to assess its hardware readiness. Our results indicate that QLSTM models achieve comparable, and in some cases slightly stronger performance metrics relative to classical LSTM models, despite using significantly fewer trainable parameters. This demonstrates their potential efficiency and robustness in capturing nonlinear temperature patterns within constrained computational resources.

Keywords: Heatwave Forecasting, Quantum Machine Learning, QLSTM, PennyLane, LSTM, Climate Modeling, IBM

Contents

Contents **1**

List of Figures **3**

List of Tables **5**

1 Introduction **7**
1.1 Background 7
1.2 Motivation 7
1.3 Objectives 8
1.4 Scope 9
1.5 Significance of the Study 9

2 Related Work **10**
2.1 Introduction to Literature Review 10
2.2 Theoretical Framework 10
2.3 Previous Research and Studies 13
2.3.1 Simulating Climatic Variables by Using Stochastic Approach . . . 13
2.3.2 Ensemble Machine Learning Approaches for Weather Forecasting 13
2.3.3 Recurrent Neural Networks for Localized Weather Prediction . . 14
2.3.4 Quantum Kernel Learning for Semiconductor Fabrication Modelling 14
2.4 Current State of the Field 15

3 Materials and Methods **16**
3.1 System Description 16
3.2 System Requirements 16
3.3 Design Constraints 17
3.4 Research Design 17
3.5 Data Design 19

3.6 Algorithmic Design 22
3.6.1 Classical Long Short-Term Memory (LSTM) 22
3.6.2 Variational Quantum Circuits (VQC) 24
3.6.3 Variational Quantum Circuits (VQC) used in QLSTM 25
3.6.4 Noisy Variational Quantum Circuits (VQC) used in QLSTM . . . 26
3.6.5 Quantum Long Short-Term Memory (QLSTM) 27
3.7 Component Design 29
3.7.1 System Decomposition 30
3.7.2 Architectural Design 30
3.7.3 Mathematical Formulation 31
3.7.4 Component Interaction Diagram 32
3.7.5 Integration with External Systems 32
3.7.6 Experimental Setup 32
3.8 Implementation and Results 32
3.8.1 Model Configuration 33
3.8.2 Benchmarking: Classical LSTM vs. Ideal QLSTM vs. Noisy QLSTM (Scottsdale Airport) 56
3.9 Discussion and Conclusion 59
3.9.1 Discussion 59
3.10 Conclusion 61

Bibliography **63**

List of Figures

1 Geographical locations of selected meteorological stations used in this study. 19
2 Data pipeline from retrieval to forecasting in the proposed heatwave prediction system. 22
3 Classical LSTM Cell Architecture illustrating its gates, memory cell, and information flow. 23
4 Variational Quantum Circuit (VQC) structure illustrating the embedding, parameterised unitary blocks, and measurement process, where n means the number of qubits. 24
5 Variational Quantum Circuit (VQC) structure illustrating the embedding, parameterised unitary blocks, and measurement process, where n means the number of qubits. 26
6 Noisy Variational Quantum Circuit (VQC) for 3 qubits, illustrating data embedding via ZZFeatureMap, parameterized entangling layers, and final measurements. Noise is applied to simulate realistic hardware effects such as gate errors and decoherence. 27
7 QLSTM Architecture illustrating the integration of Variational Quantum Circuits within the LSTM gating mechanisms. 28
8 High-level system architecture for heatwave forecasting. 30
9 Cairo: Training Loss Convergence 36
10 London: Training Loss Convergence 37
11 Barcelona: Training Loss Convergence 37
12 Phoenix: Training Loss Convergence 37
13 Riyadh: Training Loss Convergence 38
14 Confusion matrices for heatwave prediction across all stations. 38
15 Training Loss Curve for Cairo Ideal QLSTM model. 42
16 Training Loss Curve for London Ideal QLSTM model. 42
17 Training Loss Curve for Barcelona Ideal QLSTM model. 43
18 Training Loss Curve for Pheonix Ideal QLSTM model. 43

19 Training Loss Curve for Riyadh Ideal QLSTM model. 44
20 Training Loss Curve for Scottsdale Airport Ideal QLSTM model. 44
21 Confusion matrices for heatwave prediction across all stations. 45
22 Prediction Metrics Over 3 Days for Cairo Multi-Step LSTM Model. . . . 49
23 Prediction Metrics Over 3 Days for London Multi-Step LSTM Model. . . 49
24 Prediction Metrics Over 3 Days for Barcelona Multi-Step LSTM Model. . 50
25 Prediction Metrics Over 3 Days for Pheonix Multi-Step LSTM Model. . . 50
26 Prediction Metrics Over 3 Days for Riyadh Multi-Step LSTM Model. . . 50
27 Prediction Metrics Over 3 Days for Scottsdale Airport Multi-Step LSTM Model. 51
28 Training Loss Curve For Cairo Multi-Step QLSTM Model. 53
29 Training Loss Curve for London Multi-Step QLSTM Model. 53
30 Training Loss Curve for Barcelona Multi-Step QLSTM Model. 54
31 Training Loss Curve for Riyadh Multi-Step QLSTM Model. 54
32 Output of the 1 epoch noisy Qiskit 56
33 Classical LSTM Confusion Matrix for Scottsdale multi-step heatwave prediction. 58
34 Ideal QLSTM Confusion Matrix for Scottsdale multi-step heatwave prediction. 58
35 Noisy QLSTM Confusion Matrix for Scottsdale multi-step heatwave prediction. 59

List of Tables

1 Description of Meteorological Dataset Features 20
2 Cairo International Airport Model Summary and Performance Metrics . . 34
3 London Heathrow Airport Model Summary and Performance Metrics . . 34
4 Barcelona El-Prat Airport Model Summary and Performance Metrics . . . 35
5 Phoenix Sky Harbor Airport Model Summary and Performance Metrics . 35
6 Riyadh King Khalid Airport Model Summary and Performance Metrics . 36
7 Precision, Recall, and F1 Scores for Heatwave Prediction Models 39
8 Cairo Quantum Single-Step QLSTM Performance Metrics 40
9 London Quantum Single-Step QLSTM Performance Metrics 40
10 Barcelona Quantum Single-Step QLSTM Performance Metrics 40
11 Phoenix Quantum Single-Step QLSTM Performance Metrics 41
12 Riyadh Quantum Single-Step QLSTM Performance Metrics 41
13 Scottsdale Quantum Single-Step QLSTM Performance Metrics 41
14 Precision, Recall, and F1 Scores for Heatwave Prediction Models 45
15 Cairo International Airport Model Summary and Performance Metrics . . 47
16 London Heathrow Airport Model Summary and Performance Metrics . . 47
17 Barcelona El-Prat Airport Model Summary and Performance Metrics . . . 48
18 Phoenix Sky Harbor Airport Model Summary and Performance Metrics . 48
19 Riyadh King Khalid Airport Model Summary and Performance Metrics . 49
20 Cairo Multi-Step QLSTM Performance Metrics 52
21 London Multi-Step QLSTM Performance Metrics 52
22 Barcelona Multi-Step QLSTM Performance Metrics 52
23 Riyadh Multi-Step QLSTM Performance Metrics 52
24 Prediction Metrics for Each Forecast Day Noisy Qiskit Quantum Model) . 56
25 Day 1 Prediction Performance Comparison 57
26 Day 2 Prediction Performance Comparison 57
27 Day 3 Prediction Performance Comparison 57
28 Model Complexity and Training Configuration 57

List of Abbreviations

QLSTM: Quantum Long Short Term Memory.
VQC: Variational Quantum Circuit.
RNNs: Recurrent Neural Networks.
QML: Quantum Machine Learning.
MCC: Matthews Correlation Coefficient.

Chapter 1

Introduction

1.1 Background

Heatwaves have long posed significant threats across multiple societal sectors, including agriculture, transportation, disaster management, and public health [17]. Their frequency, intensity, and duration have all risen due to the worsening effects of global warming, which has a negative impact on human well-being, infrastructural resilience, and economic stability. For instance, studies conducted on Spain's 17 Autonomous Communities between 2000 and 2013 demonstrate the cross-sectoral effects of global warming on electricity consumption in the fields of agriculture, construction, industry, services, and public administration [15].

The capability of traditional weather forecasting techniques to capture the sequential, non-linear, and frequently chaotic patterns found in meteorological data is naturally limited since they primarily rely on statistical models that presume independence among data points [23]. Recent advances in machine learning have led to powerful alternatives capable of effectively modeling these temporal correlations.

1.2 Motivation

Global warming has led to an increase in the frequency and intensity of heatwaves, endangering infrastructural stability, agricultural output, and public health [15]. Conventional forecasting methods often struggle to capture the complex temporal and non-linear patterns in meteorological data, limiting their ability to provide early warnings and valuable insights [23]. Despite their advanced capabilities, deep learning models such as RNNs and traditional LSTMs face challenges such as vanishing gradients in long sequences requiring significant computational resources due to their large number of parameters. [1, 8, 9].

By incorporating Variational Quantum Circuits (VQCs) into LSTM topologies, recent research has presented Quantum Long Short-Term Memory (QLSTM) networks as an alternative that can overcome these drawbacks [17]. In addition to drastically reducing the number of trainable parameters, this approach makes better use of quantum features like entanglement and superposition to describe complicated data distributions. Examining QLSTM models is a possible avenue for improving climate resilience forecasting tools, especially in light of the pressing need to create scalable and computationally efficient models that preserve high forecasting accuracy.

1.3 Objectives

Assessing the potential of Quantum Long Short-Term Memory (QLSTM) models to improve heatwave forecasting accuracy in comparison to standard LSTM models is the main goal of this study. This study specifically seeks to:

1. Execute and contrast traditional LSTM and QLSTM systems in the two forecasting scenarios—predicting the next day using three days of history data and predicting the following three days using seven days of input data.

2. Assess the generalizability of the model in predicting heatwaves with varying temperature dynamics and volatility, by exploring its performance across a variety of climatic regions, such as Barcelona, Cairo, London, Phoenix, and Riyadh.

3. Investigate parameter efficiency and robustness by analyzing the number of trainable parameters and the models' ability to capture non-linear temporal dependencies effectively.

4. Prepare for future deployment on actual quantum devices and assess hardware readiness by running the QLSTM model under realistic noise settings on IBM's quantum hardware simulator.

Through these objectives, this study seeks to determine whether QLSTM models can serve as computationally efficient and accurate alternatives to classical deep learning models in operational heatwave forecasting systems.

1.4 Scope

Using temperature time series data, this study investigates the use of Quantum Long Short-Term Memory (QLSTM) models for heatwave forecasting. In order to assess the potential benefits of Variational Quantum Circuits (VQCs) over traditional LSTM designs, it specifically looks into how VQCs may be integrated into LSTM gates. Two forecasting setups are considered: predicting the next day using three days of input data, and predicting the next three days using seven days of input data. However, the detailed analysis and deployment are focused on the latter configuration because of its practical relevance for extended forecasting. To evaluate the models' generalisability across various heatwave profiles, experiments are carried out using meteorological datasets from a variety of climatic locales, such as Barcelona, Cairo, London, Phoenix, and Riyadh. In order to assess the practicality of deploying the QLSTM models on actual quantum hardware in further research, the scope also includes training and assessing the models on simulated quantum backends, such as noisy quantum simulations utilising IBM's Qiskit Aer simulator and Pennylane's mixed backend. This work evaluates the modelling feasibility of quantum techniques for this crucial environmental application, although it does not deploy fully functional heatwave early warning systems and is limited to univariate temperature forecasts.

1.5 Significance of the Study

The significance of this study lies in its exploration of quantum-enhanced forecasting methods within the critical domain of heatwave prediction. By evaluating the Quantum Long Short-Term Memory (QLSTM) model against its classical counterpart, this research provides valuable insights into the practical applicability of quantum machine learning for complex, non-linear climate time series data. Given that heatwaves pose escalating threats to public health, infrastructure, and economic stability, developing more efficient and potentially faster forecasting models is of substantial importance for proactive disaster management and resilience planning. Additionally, this work highlights a practical approach to computational efficiency by showing that it is possible to include Variational Quantum Circuits into traditional structures to lower model parameter counts while preserving predictive accuracy.

Chapter 2

Related Work

2.1 Introduction to Literature Review

Many conventional and quantum-enhanced machine learning algorithms have been used to forecast extreme weather events, especially heatwaves. With Matthews Correlation Coefficient (MCC) scores of 0.45 at shorter lead periods, classical deep learning models, such the CNN-based architecture created for France's prolonged heatwaves [12], have reached notable predicting horizons of up to 15 days. Although these models used dataset balancing strategies to address class imbalance and convolutional kernels to extract spatial information, they were not interpretable enough to uncover the underlying physical processes causing heatwave events. Recent developments have concentrated on using Quantum Machine Learning (QML) techniques to increase accuracy and processing efficiency. For example, for short-term temperature forecasting in Ottawa, QGAPHEnsemble proposed a hybrid QLSTM ensemble with adaptive weighting, producing a MAPE of 0.91 as opposed to 1.99 by conventional LSTM [17]. These models demonstrated the potential of QML in enhancing predictive performance while lowering parameter counts by optimising hyperparameters using Quantum Genetic Algorithms and Particle Swarm Optimisation.

2.2 Theoretical Framework

Quantum Information Fundamentals

All computational systems manipulate information, where the fundamental unit of quantum information is the qubit. Unlike classical bits, which exist solely as 0 or 1, qubits exist in a linear superposition:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle, \tag{2.1}$$

where α and β are complex amplitudes constrained by $|\alpha|^2+|\beta|^2= 1$. Measurement collapses the qubit into one of the basis states probabilistically, irreversibly losing prior amplitude information [3]. Furthermore, multiple qubits form entangled states that cannot be factored into individual qubit products, such as the Bell state:

$$|\Phi^-\rangle = \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle), \tag{2.2}$$

Superposition

Superposition is one of the fundamental concepts underlying quantum computing. In essence, a superposition is a state where a qubit exists in multiple states simultaneously. These states are generally linear combinations of the basis states of the system, which, in quantum computing, are most often binary.

The binary basis states 0 and 1 are represented as $|0\rangle$ and $|1\rangle$ respectively, and mathematically defined as:

$$|0\rangle = \begin{bmatrix} 1 & 0 \end{bmatrix}, \quad |1\rangle = \begin{bmatrix} 0 & 1 \end{bmatrix}. \tag{2.3}$$

A general superposition state can be written as:

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle, \tag{2.4}$$

where α and β are complex coefficients representing the amplitudes of the respective basis states. The squared magnitudes of these coefficients represent probabilities, and their sum must equal one:

$$|\alpha|^2+|\beta|^2= 1. \tag{2.5}$$

Here, $|\alpha|^2$ indicates the probability of measuring the qubit in state $|0\rangle$, while $|\beta|^2$ represents the probability of finding it in state $|1\rangle$ upon measurement.

The Hadamard gate is commonly used to place a qubit into a superposition state. However, it is important to note that upon measurement, the qubit's superposition collapses to one of its basis states, which presents a significant challenge when extracting information from quantum systems [21].

Feature Mapping in Quantum Machine Learning

Feature mapping encodes classical inputs into quantum states for further processing. In this study, two embedding methods were employed:

AngleEmbedding with Learnable Rotations Because of its ease of use and compatibility with multi-feature inputs, angle embedding was chosen for the single-step forecasting model, which makes use of the entire set of meteorological data features. Angle embedding uses single-qubit rotation gates to encode classical inputs x, with each input element determining the angle of its matching rotation gate:

$$|\psi_x\rangle = \bigotimes_{i=1}^{n} R(x_i)|\psi_0\rangle, \tag{2.6}$$

where R is a rotation matrix applied to each qubit, and $|\psi_0\rangle$ is the initial quantum state.

To reduce the number of required qubits, sequential rotations can be applied to each single qubit i as:

$$V_i = \bigotimes_{j=1}^{S} R_{j,i}(x_j), \tag{2.7}$$

where the angles are fixed. Introducing a learnable scaling factor reformulates this as:

$$V_i(\theta_i) = \bigotimes_{j=1}^{S} R_{j,i}(\theta_{j,i} \cdot x_j), \tag{2.8}$$

making the encoding parameterised by θ.

The overall unitary transformation becomes:

$$U(\theta) = \prod_{i=1}^{n} V_i(\theta_i) W_i, \tag{2.9}$$

where W_i are non-parameterised gates such as CNOTs. To promote sparsity, dropout is introduced as:

$$U(\theta) = \prod_{i=1}^{n} V_i(\theta_i)(\delta_i \cdot W_i), \tag{2.10}$$

where $\delta_i \sim \text{Bernoulli}(p)$ with $0 < p < 1$.

Typically, the encoding begins with a Hadamard gate to generate a uniform superposition, leading to:

$$|\psi_y\rangle = U(\theta)(H|\psi_0\rangle), \tag{2.11}$$

where H denotes the Hadamard gate.

ZZFeatureMap For the multi-step forecasting model, which uses only the temperature feature as input, the ZZFeatureMap [6] was employed. The ZZFeatureMap extends the standard ZFeatureMap by introducing entanglement between qubits, thus capturing pairwise interactions between features. While the ZFeatureMap focuses solely on individual feature rotations, the ZZFeatureMap incorporates entangling gates, enabling it to represent relationships between input dimensions more effectively. However, this increased expressivity comes at the cost of higher sensitivity to quantum noise, particularly phase-flip errors affecting entangled qubits, which can degrade learning performance [18].

2.3 Previous Research and Studies

2.3.1 Simulating Climatic Variables by Using Stochastic Approach

For agricultural planning, productivity, and disease management, Yürekli et al. (2007) concentrated on forecasting important climate factors including temperature, relative humidity, and sun radiation. They stressed that the majority of conventional statistical techniques use the premise that observations are independent throughout time, which is frequently untrue for meteorological data because values have a tendency to cluster. They used the Autoregressive Integrated Moving Average (ARIMA) model because it was better than neural networks, exponential smoothing, and moving averages in accounting for serial correlations in data. Its ability to incorporate both autoregressive (AR) and moving average (MA) components through a methodical three-stage process—model selection, parameter estimation, and diagnostic checking—is what gives ARIMA its versatility. Nevertheless, the model's drawbacks include the need for a sizable dataset (at least 50 observations) and the demand for practitioner knowledge to guarantee precise implementation.

2.3.2 Ensemble Machine Learning Approaches for Weather Forecasting

An ensemble machine learning architecture was presented by Sravanthi et al. (2020) with the goal of enhancing weather forecast accuracy and lessening the effects of abrupt climate change and natural disasters. Their strategy used many ensemble techniques, including as stacking algorithms, random forest, boosting, and bagging. According to the study, boosting repeatedly improves models by concentrating on previously misclassified data points, whereas bagging increases model stability by averaging findings from several bootstrap samples. By choosing feature subsets during tree building, Random Forest, like bagging,

adds more randomisation while lowering prediction variance. In contrast, stacking leverages diverse model strengths to improve overall performance by combining many base learners through a meta-learner. They trained base learners on a subset of data and used their predictions as inputs for a higher-level model as part of their two-level stacking technique. In addition, they described specific methods such as Support Vector Machines (SVM) for optimal hyperplane classification, Generalised Boosting Models (GBM) for iterative decision tree refinement, and Linear Discriminant Analysis (LDA) for dimensionality reduction. According to their findings, the ensemble architecture proved to be a reliable approach for weather forecasting applications, regularly outperforming individual algorithms [19].

2.3.3 Recurrent Neural Networks for Localized Weather Prediction

The shortcomings of typical weather files, which often fall short of accurately capturing local climatic conditions in building performance simulations (BPS), were discussed by Han et al. (2021). They suggested employing Recurrent Neural Networks (RNNs) to provide more accurate localised meteorological data. Their study demonstrated that, although being synthesised over extended periods of time and frequently derived from airports, typical weather files like TMY or WYEC2 do not account for microclimate differences such urban heat islands, which causes errors in energy and daylighting models. RNNs were selected because of their capacity to represent intricate temporal relationships; the vanishing gradient issue that normal RNNs have is successfully resolved by designs such as Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU). They also contrasted RNNs with other techniques, pointing out that although statistical methods such as ARIMA are used for time-series data, they are not scalable for numerous variables and are ineffective at handling missing data. Their findings showed that RNN models may be used to efficiently forecast weather parameters by merging restricted local data with larger airport datasets, improving simulation accuracy in localised contexts [5].

2.3.4 Quantum Kernel Learning for Semiconductor Fabrication Modelling

The difficult challenge of simulating intricate processes in semiconductor fabrication, particularly Ohmic contact creation, which is recognised for its high-dimensional parameter space and sparse availability of experimental data; this was addressed by Wang et al. in 2025. They stressed how the enormous expense and time needed for semiconductor production frequently limits the collection of such data. Despite much research into using classical machine learning (CML) techniques to improve process modelling and device

characterisation, these models often perform poorly in nonlinear, small-sample datasets. Because CML mostly depends on big datasets to adequately reflect the complex nonlinear interactions in semiconductor processes, this results in overfitting and poor generalisation. Wang and their group investigated quantum computing (QC) methods, particularly quantum machine learning (QML), to overcome these constraints. The ability of QML to efficiently convert classical data into high-dimensional quantum Hilbert spaces makes it unique and enables it to identify subtle patterns in even the smallest datasets. QML has superior generalisation capabilities with small datasets than CML because it makes use of quantum kernels, which capture intricate feature relationships. For semiconductor modelling, where data is limited and process parameters are highly interconnected, kernel-based QML is therefore an acceptable technique. Despite its theoretical benefits, their research also underlined that QML has not yet demonstrated appreciable gains over CML in semiconductor manufacturing models, especially for crucial processes like Ohmic contact creation in GaN high-electron-mobility transistors (HEMTs). Given the sensitive and nonlinear nature of GaN HEMT data, this serves as an ideal test case to explore QML's potential in such high-cost, data-limited fabrication environments [22].

2.4 Current State of the Field

Accurately identifying and predicting heatwaves is still difficult, despite advancements in heatwave forecasting brought about by statistical and machine learning algorithms. Long-term predictions and infrequent severe occurrences like heatwaves are frequently difficult for current models, such as ARIMA, RNNs, and ensemble techniques, to handle, despite their effectiveness in capturing trends and short-term temperature fluctuations. Because of this, it is challenging to deliver trustworthy early warnings, which are essential for infrastructure development and public safety. Concurrently, the topic of quantum computing is gaining momentum and has great promise for improving computational tasks, such as weather forecasting. Its ability to be applied practically to real-world issues is still restricted, though. The measurement of qubits collapses their superposition states, which impacts the stability of the results. Research explicitly using quantum algorithms for heatwave forecasting is currently lacking, despite the fact that many recent articles focus on enhancing these algorithms to lower noise. Therefore, using quantum computing for reliable and accurate heatwave forecasting is still a problem in the field.

Chapter 3

Materials and Methods

3.1 System Description

The system developed in this research focuses on forecasting heatwave events by predicting daily maximum temperatures using both classical Long Short-Term Memory (LSTM) models and Quantum Long Short-Term Memory (QLSTM) models. The aim is to assess the performance benefits of integrating quantum computing techniques within traditional deep learning architectures for climatic prediction tasks. The forecasting system operates as a time series regression framework. It inputs historical weather data to output temperature predictions across different forecasting horizons. To improve model expressivity and learning capability, variational quantum circuits were included into LSTM gate operations to create the QLSTM models. The purpose of this method is to assist urban planners and climate researchers in comprehending the viability of quantum-enhanced models for crucial environmental applications like heatwave forecasting.

3.2 System Requirements

To implement and evaluate the system effectively, several requirements were established:

- Hardware Requirements:
  - A local machine with CPU to efficiently train classical LSTM models and handle large matrix computations.
  - CPU/GPU Acceleration with high RAM capacity for quantum simulation tasks, as quantum circuit simulators are resource-intensive, especially with increasing qubit counts.

- Software Requirements
  - Python (3.8) as the main programming environment.
  - TensorFlow and Keras for classical LSTM model development, training, and evaluation.
  - PennyLane to construct and integrate quantum circuits within the QLSTM architecture.
  - Qiskit As the quantum simulator backend for executing parameterised quantum circuits.

3.3 Design Constraints

Key constraints addressed in this study include:

- Data Availability and Quality: Weather datasets contained missing values and irregular timestamp recordings, requiring thorough data cleaning, imputation, and aggregation to ensure consistency in input sequences.
- Quantum Simulation Limitations: Due to current quantum hardware noise levels and qubit scalability restrictions, QLSTM models were simulated rather than executed on physical quantum computers. This limited experimentation to smaller qubit configurations to maintain feasible simulation times.
- Computational Cost: Quantum circuit simulations are computationally intensive, often requiring significantly longer runtimes compared to classical models. Resource management and runtime optimisation were essential throughout experiments.
- Model Generalisation: Ensuring that models did not overfit to specific city datasets but learned generalisable temporal patterns was a primary concern during training and evaluation phases.

3.4 Research Design

The research followed a systematic design comprising the following stages:

1. **Data Collection:**

- Historical meteorological data for five cities—Barcelona, Cairo, London, Phoenix, and Riyadh—were retrieved using the Iowa Environmental Mesonet ASOS API.
- Collected parameters included air temperature, dew point, relative humidity, wind direction and speed, altimeter setting, sea-level pressure, visibility, and observation timestamps.

2. **Data Preprocessing:**
    - Missing data points were handled through forward filling or mean imputation.
    - The dataset was aggregated to daily maximum temperatures, ensuring uniformity across all input sequences for model training.
3. **Heatwave Definition:**
    - Heatwaves were defined as periods where the daily maximum temperature exceeded the 90th percentile threshold for three or more consecutive days in each city, aligning with standard climatological practices.
4. **Model Implementation:**
    - **Classical Long Short-Term Memory (LSTM):** Developed using Keras, comprising input, hidden, and output dense layers.
    - **Quantum Long Short-Term Memory (QLSTM):** Implemented using PennyLane, integrating variational quantum circuits within LSTM gate structures to enhance learning capacity.
5. **Model Configurations:**
    - **Single-step forecasting:** Used three days of historical temperature data to predict the next day's maximum temperature.
    - **Multi-step forecasting:** Used seven days of input data to predict maximum temperatures for the subsequent three days.
6. **Evaluation:** Models were evaluated using the following metrics to assess performance across cities and forecast horizons:
    - **R^2 (Coefficient of Determination)**:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{3.1}$$

where y_i is the true value, $\hat{y}_i$ is the predicted value, $\bar{y}$ is the mean of true values, and n is the number of samples.

- **MAE (Mean Absolute Error)**:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{3.2}$$

- **RMSE (Root Mean Squared Error)**:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{3.3}$$

3.5 Data Design

For this study, raw meteorological data was obtained for multiple global stations including Phoenix, Riyadh, London, Barcelona, and Scottsdale

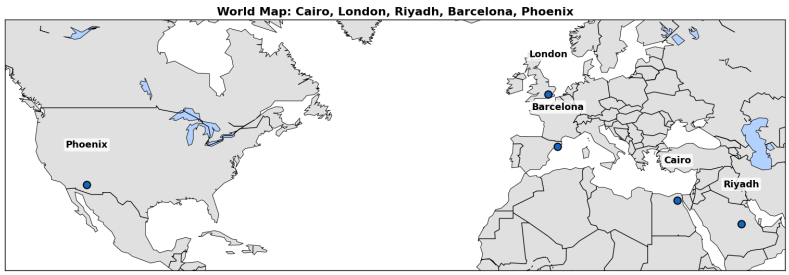

Figure 1: Geographical locations of selected meteorological stations used in this study.

Station Geographical Coverage The map in Figure 1 shows the distribution of the meteorological stations included in this research: Cairo (Egypt), Riyadh (Saudi Arabia), Phoenix and Scottsdale (USA), London (UK), and Barcelona (Spain). These stations were purposefully chosen to represent diverse climatic regions and heatwave patterns, including arid desert climates (Riyadh, Phoenix), Mediterranean coastal climates (Barcelona), temperate maritime climates (London), and hot semi-arid climates (Cairo).

This geographical diversity ensures the models are trained and evaluated on data covering a wide range of heatwave characteristics, enabling assessment of their generalisability across regions with different temperature dynamics, humidity levels, and weather variability. Such coverage is essential for developing robust heatwave forecasting models applicable to global climate resilience efforts.

Table 1: Description of Meteorological Dataset Features

Feature	**Description**
`tmpf`	Air temperature measured in degrees Fahrenheit (°F), representing the ambient temperature near the surface at the time of observation.
`dwpf`	Dew point temperature in degrees Fahrenheit (°F), indicating the temperature at which air becomes saturated, reflecting atmospheric moisture content.
`relh`	Relative humidity (%), the ratio of current vapor pressure to saturation vapor pressure, indicating how close the air is to saturation.
`drct`	Wind direction in degrees, measured clockwise from true north, indicating the direction from which the wind is blowing.
`sknt`	Wind speed in knots, representing the magnitude of horizontal air movement near the surface.
`alti`	Altimeter setting in inches of mercury (inHg), indicating station pressure adjusted to sea level for aviation purposes.
`mslp`	Mean sea-level pressure in millibars (mb), representing atmospheric pressure reduced to sea level, essential for synoptic weather analysis.
`vsby`	Visibility in miles, representing the maximum horizontal distance at which objects can be clearly seen and identified.
`valid`	Timestamp of observation in UTC, indicating the exact date and time of the recorded measurement.

The meteorological dataset 1 used in this study comprises a diverse set of features, each playing a distinct role in heatwave prediction. Air temperature (tmpf) offers direct measurements of near-surface thermal conditions, forming the primary indicator for heatwave detection and threshold definition. Dew point temperature (dwpf) represents the temperature at which air reaches saturation, thus serving as a proxy for atmospheric moisture

content, which in turn influences perceived temperature and human heat stress. Relative humidity (relh), calculated as the ratio of actual to maximum possible vapor pressure, further characterises atmospheric moisture and affects thermal comfort indices.

Wind-related parameters, namely wind direction (drct) and wind speed (sknt), provide insights into air mass movement, which can alter surface temperature distributions and contribute to convective heat exchange and cooling. Altimeter setting (alti), representing station pressure corrected to sea level in inches of mercury, and mean sea-level pressure (mslp) in millibars, are critical for understanding the larger-scale synoptic systems driving regional weather patterns and temperature dynamics. Visibility (vsby), recorded in miles, can indirectly indicate atmospheric phenomena such as haze, fog, or dust, reflecting humidity levels or temperature inversion layers that influence local heating. Lastly, the valid feature captures the exact timestamp of each observation in UTC, enabling rigorous time series ordering and sequence-based model training.

Prior to modelling, the datasets were systematically assessed for missing values across all parameters to ensure data integrity. For instance, the Phoenix dataset exhibited substantial missing entries, with 20,315 absent wind direction records, 3,045 missing temperature data points, and 3,122 missing dew point temperature entries. To prepare for time series analysis, the timestamp column ('valid') was converted to datetime format. Subsequently, daily maximum temperatures were extracted by grouping the data by date and selecting the highest recorded temperature for each day within the study period spanning from January 1, 2010, to October 21, 2024.

Heatwave thresholds for each city were then determined using the 90th percentile of daily maximum temperatures. This yielded specific thresholds such as 42.22°C for Phoenix, 44.0°C for Riyadh, 24.0°C for London, 36.0°C for Cairo, and 28.0°C for Barcelona, reflecting their respective climatic regimes.

Missing data within the processed daily datasets were addressed using a combination of imputation techniques. Forward filling was employed to maintain temporal continuity, particularly for sequential model requirements, while mean imputation was used for numerical features such as wind direction ('drct') and altimeter setting ('alti') where forward filling was not feasible due to non-sequential missing patterns. These preprocessing steps ensured that the final datasets were free of missing values, enabling robust and uninterrupted input preparation for the subsequent LSTM and QLSTM model training and evaluation phases.

The overall data pipeline implemented in this research is illustrated in Figure 2. It outlines each step starting from data retrieval to model training and prediction output, providing a clear overview of the systematic approach used for heatwave forecasting in this study.

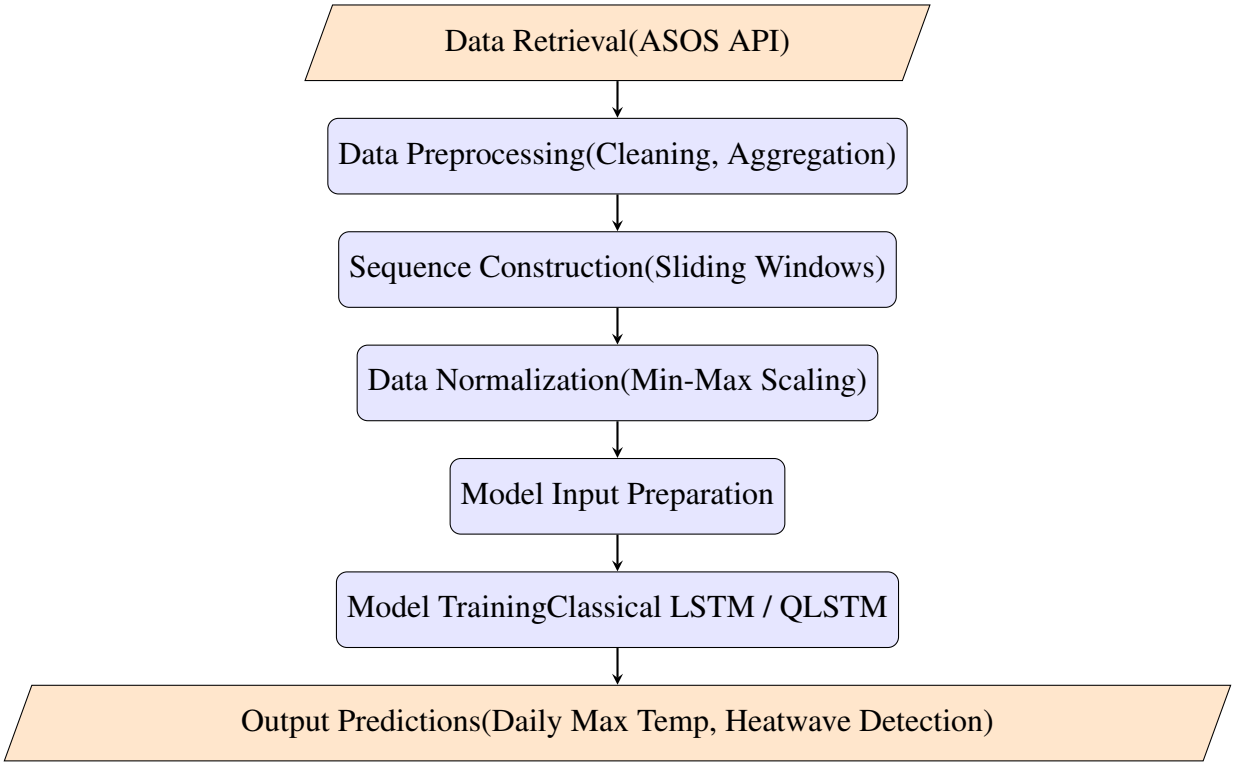

Figure 2: Data pipeline from retrieval to forecasting in the proposed heatwave prediction system.

3.6 Algorithmic Design

3.6.1 Classical Long Short-Term Memory (LSTM)

The Long Short-Term Memory (LSTM) model is a specialised recurrent neural network (RNN) architecture designed to overcome the limitations faced by traditional RNNs, particularly the vanishing and exploding gradient problems. These issues often prevent standard RNNs from learning long-term dependencies in sequential data, especially when relevant events are separated by long time lags. The LSTM addresses this through its unique gated memory cell design, enabling it to retain information over extended sequences effectively [20].

At its core, an LSTM unit functions as a memory block capable of maintaining its state over time through non-linear gating mechanisms that regulate the flow of information. The architecture includes three primary gates: input, forget, and output gates. These gates

work alongside the central memory cell to control what information is added, retained, or output at each time step, thereby allowing the model to capture both short-term dynamics and long-term dependencies.

What data from the previous cell state should be retained or discared is determined by the forget gate. It functions as a filter to identify the areas of long-term memory that are still relevant to the present prediction job. The LSTM may then selectively update its internal state with fresh information obtained from the current input and the prior concealed state thanks to the input gate, which then regulates what new data should be added to the memory cell. In parallel, the candidate information is created, suggesting possible modifications to take into account.

In order to maintain a balanced integration of the new inputs and the old context, the cell state is updated following these gating choices by merging the new candidate information with the previously stored information. Lastly, at the current time step, the output gate decides which aspect of the updated cell state should affect the output. This guarantees that only the most contextually relevant data is sent on as the hidden state output, which is then used as input for the model's output layer or the next step.

This sophisticated gating structure enables LSTM models to regulate information flow effectively, allowing them to selectively remember or forget information across sequences. This makes them highly suitable for tasks involving complex temporal dependencies, such as weather forecasting and heatwave prediction, where patterns span across multiple days or weeks [10, 20] .

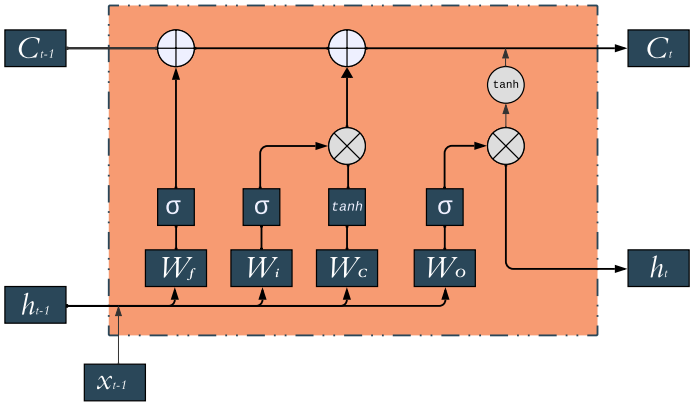

Figure 3: Classical LSTM Cell Architecture illustrating its gates, memory cell, and information flow.

3.6.2 Variational Quantum Circuits (VQC)

Variational Quantum Circuits (VQCs) are parameterised quantum circuits that form the core computational units of many quantum machine learning algorithms. They leverage quantum mechanical properties such as superposition and entanglement to encode and process data efficiently within the high-dimensional Hilbert space [**?**].

A typical VQC consists of three main components. First, the **feature embedding layer** encodes classical data into quantum states. This can be achieved using angle embedding, where classical features are mapped directly to rotation angles on qubits, or more advanced methods "For instance, like the ZZ Feature Map", which introduces entanglement between qubits to model pairwise feature correlations effectively [6, 16].

Second, the parameterised unitary blocks include layers of rotation gates interleaved with entangling gates (such as Controlled-NOT or Controlled-Z). These are the trainable components of the circuit, optimised to approximate complex nonlinear functions [14].

Finally, the measurement stage collapses the quantum state to produce classical outputs. This output is used in hybrid quantum-classical models, either as direct predictions or as inputs to further classical layers.

The parameters of VQCs are optimised using gradient-based methods, with the parameter-shift rule enabling efficient gradient calculation within quantum circuits, making them practical for integration with backpropagation in hybrid frameworks.

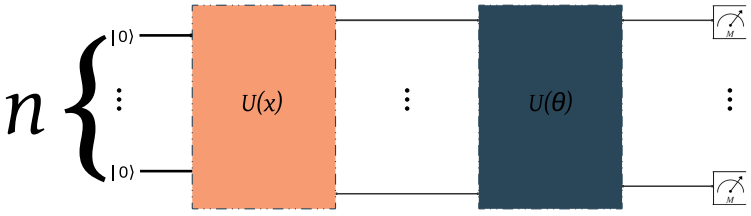

Figure 4: Variational Quantum Circuit (VQC) structure illustrating the embedding, parameterised unitary blocks, and measurement process, where n means the number of qubits.

The diagram in Figure 4 illustrates the conceptual structure of a Variational Quantum Circuit (VQC) used in this study. On the left side, the input vector x, consisting of classical features, is encoded into quantum states via the feature embedding block labelled $U(x)$. This block applies parameterised rotation gates to translate each classical feature into a corresponding quantum rotation, effectively embedding it into the qubit register. For instance, if x has n features, then n qubits are initialised in the $|0\rangle$ state, and each undergoes a rotation based on its respective feature value, transforming the qubit into a superposition state that represents the input.

The middle block, denoted $U(\theta)$, represents the parameterised unitary transformation layer. This section includes multiple trainable quantum gates with parameters θ that are updated during model training. These gates typically consist of single-qubit rotations and multi-qubit entangling gates such as Controlled-NOT (CNOT) or Controlled-Z (CZ) gates, enabling the circuit to learn complex transformations and relationships between input features. In practical terms, this block functions analogously to the hidden layers of a classical neural network, where parameters are optimised to minimise model prediction error.

Finally, the circuit concludes with the measurement stage on the right side, where each qubit is measured in the computational basis. These measurements collapse the quantum states, producing classical outputs that reflect the learned transformations from the embedding and unitary blocks. The output expectation values can be interpreted directly or used as inputs to subsequent classical post-processing layers for final predictions.

Overall, Figure 4 summarises the data flow within a VQC: starting from classical input features, proceeding through quantum embedding, learning complex feature representations via parameterized unitary operations, and producing measurable classical outputs that integrate seamlessly into hybrid quantum-classical machine learning pipelines.

3.6.3 Variational Quantum Circuits (VQC) used in QLSTM

The quantum circuit depicted in Figure 5 showcases a 3-qubit Variational Quantum Circuit (VQC) illustrating a complete pipeline from feature embedding to entanglement and measurement. This example demonstrates how a compact quantum circuit can encode classical data, process it through quantum entanglement, and produce measurable outputs usable in quantum machine learning models.

The circuit begins with three qubits initialized in the $0\rangle$ state, each of which is passed through a Hadamard (H) gate. This operation places each qubit into an equal superposition of $0\rangle$ and $1\rangle$, enabling the quantum system to represent multiple computational paths simultaneously, which is fundamental to quantum.

Following this, parameterized Rz rotations are applied to each qubit, where the classical input features are directly mapped to the rotation angles of these gates. This process embeds the input data into the quantum state of the system, with each qubit's phase adjusted according to its corresponding feature value, there by encoding the classical information into the quantum computational basis.

Once the data is embedded, the circuit applies to a series of ZZ entangling gates between the qubits to capture pairwise feature correlations within the input data. Specifically, ZZ gates are applied between qubit pairs. These gates entangle the qubits, creating correlations that allow the circuit to model non-trivial interactions between the input features, a capability that is central to the expressive power of quantum machine learning models.

To further enhance the circuit's expressivity, Controlled-NOT (CNOT) gates are applied, facilitating additional entanglement and enabling the circuit to represent more complex nonlinear transformations on the embedded input data. These operations collectively function analogously to the hidden layers in a classical neural network, allowing the circuit to learn complex patterns and relationships within the data.

The circuit concludes with measurements in the Z-basis on each of the three qubits. This measurement step collapses the quantum states, producing classical outputs that reflect the transformations encoded within the quantum circuit. These outputs can then be interpreted directly as model predictions or passed to subsequent classical layers for further processing, completing the quantum pipeline.

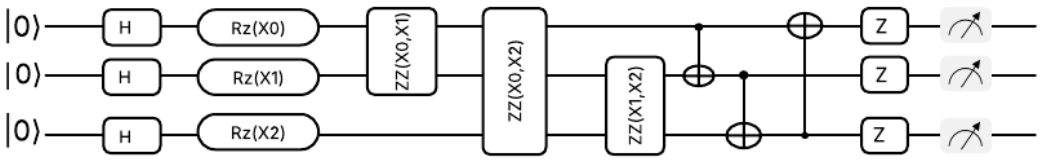

Figure 5: Variational Quantum Circuit (VQC) structure illustrating the embedding, parameterised unitary blocks, and measurement process, where n means the number of qubits.

3.6.4 Noisy Variational Quantum Circuits (VQC) used in QLSTM

The quantum circuit presented in Figure 6 extends the previously described 3-qubit Variational Quantum Circuit (VQC) by incorporating noise simulation to reflect realistic quantum hardware conditions encountered in Noisy Intermediate-Scale Quantum (NISQ) devices. While the fundamental structure of the circuit remains unchanged, with feature embedding, entanglement, and measurement stages, the key distinction in this circuit is the inclusion of depolarizing noise (DP) channels applied between gates throughout the circuit.

Depolarizing noise is a common noise model used to simulate the imperfections present in real quantum hardware, where with a certain small probability, the state of a qubit is replaced by a maximally mixed state, effectively modeling decoherence and operational errors. In this circuit, small depolarizing probabilities (e.g., 0.0002 for single-qubit operations and 0.0046 for two-qubit operations) are inserted between each layer, including after

the Hadamard gates, parameterized RZ rotations, ZZ entangling gates, and Controlled-NOT gates. Importantly, this implementation adopts a comprehensive approach by adding noise to all wires after each gate operation, not just to the qubits targeted by the specific gate. This models a scenario where every qubit is affected by noise at every time step, accounting for realistic effects such as decoherence of idle qubits, crosstalk between adjacent qubits, and global environmental disturbances. Such an approach ensures a more accurate and pessimistic simulation of real-device performance, helping test the robustness of the quantum algorithm under worst-case conditions.

The purpose of introducing these noise layers is to evaluate the robustness and stability of the VQC under realistic, noisy execution conditions. By incorporating depolarizing noise during the circuit simulation, it becomes possible to assess how well the quantum pipeline can tolerate and learn in the presence of quantum noise, ensuring that the trained models can generalize effectively when deployed on actual quantum hardware. This step is particularly crucial in the development and validation of practical quantum machine learning models, as it bridges the gap between ideal theoretical simulations and the limitations of current quantum technologies.

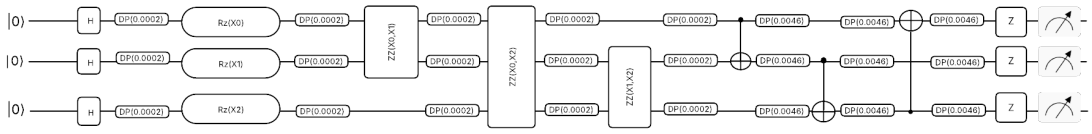

Figure 6: Noisy Variational Quantum Circuit (VQC) for 3 qubits, illustrating data embedding via ZZFeatureMap, parameterized entangling layers, and final measurements. Noise is applied to simulate realistic hardware effects such as gate errors and decoherence.

3.6.5 Quantum Long Short-Term Memory (QLSTM)

The Quantum LSTM (QLSTM) model extends the classical LSTM architecture that was just mentioned earlier (check 3) by incorporating Variational Quantum Circuits within its gating mechanisms to enhance learning capacity and feature extraction. Instead of using fully connected classical neural layers for gate computations, QLSTM integrates VQCs to exploit quantum-enhanced function representation [4].

In QLSTM, each gate (input, forget, output) traditionally computed by dense neural layers is replaced with a **hybrid quantum-classical layer** incorporating a VQC:

- The input features for each gate are first embedded into quantum states using the chosen embedding strategy (Angle Embedding or ZZ Feature Map).
- These embedded states are processed through the parameterised quantum layers within the VQC.
- Finally, the quantum states are measured to produce classical outputs, which act as gate activations within the LSTM cell, performing the same role as outputs from classical dense layers.

By integrating VQCs in this manner, QLSTM leverages the quantum circuit's ability to represent complex, high-dimensional functions efficiently, potentially capturing subtle temporal and feature dependencies in weather data that classical models may not detect. This is particularly valuable in heatwave forecasting, where non-linear dependencies and multi-timescale patterns govern the dynamics.

The QLSTM was implemented using the PennyLane framework's QNode constructs, allowing seamless backpropagation through the quantum-classical computational graph and integration with TensorFlow for gradient-based optimisation and training [2].

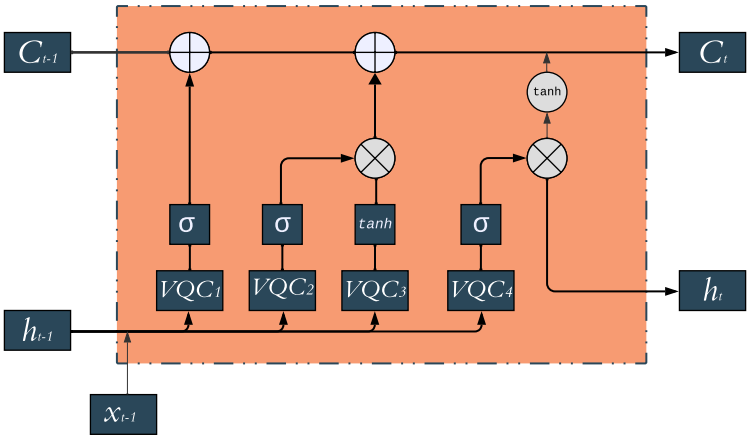

Figure 7: QLSTM Architecture illustrating the integration of Variational Quantum Circuits within the LSTM gating mechanisms.

Figure 7 illustrates the internal structure of the Quantum Long Short-Term Memory (QLSTM) cell implemented in this study. This architecture extends the classical LSTM by replacing its dense neural network layers within each gate with Variational Quantum Circuits (VQCs), leveraging quantum computational advantages to enhance temporal feature learning.

At each time step, the QLSTM cell receives two inputs: the previous hidden state h_{t-1} and the current input feature vector x_t. These inputs are concatenated and fed into four separate VQCs, each corresponding to a specific gate or candidate computation within the cell.

On the left side, the forget gate is computed. The combined input is passed through the first variational quantum circuit VQC_1, whose output is processed by a sigmoid activation function. This gate determines which components of the previous cell state C_{t-1} should be retained or discarded, maintaining only the relevant long-term memory needed for the current prediction.

Next, the input gate is calculated using VQC_2, followed by a sigmoid activation, to decide which parts of the candidate information will be written to the cell state. Simultaneously, the candidate value is computed using VQC_3, followed by a tanh activation function, generating the new candidate information to be potentially integrated.

The updated cell state C_t is then derived by combining these computations: it retains information from the previous cell state filtered by the forget gate and integrates new candidate information scaled by the input gate. This process enables the QLSTM to selectively update its long-term memory based on both past context and new inputs.

On the right side, the output gate is computed using VQC_4 and a subsequent sigmoid activation. The final hidden state h_t is produced by applying a tanh activation to the updated cell state and scaling it by the output gate's activation. This hidden state serves both as the output for the current time step and as an input to the next time step, allowing temporal information to flow through the sequence.

Overall, the integration of VQCs within each gate computation enhances the model's capability to capture complex, nonlinear dependencies in sequential data. By leveraging quantum feature embeddings and parameterised transformations, QLSTM aims to outperform its classical counterpart in tasks requiring sophisticated temporal pattern learning, such as heatwave forecasting and climatic sequence prediction.

3.7 Component Design

This section describes the detailed design and formulation of each core component within the developed heatwave forecasting system, including the Classical LSTM and Quantum Long Short-Term Memory (QLSTM) models. The design is structured to clearly explain data processing pipelines, model architectures, mathematical formulations, and integration processes.

3.7.1 System Decomposition

The overall system is decomposed into three main components:

1. **Data Preprocessing Component**: Handles data cleaning, missing value interpolation, feature scaling, and sequence generation for model input.
2. **Classical LSTM Component**: Implements a standard LSTM architecture for sequential temperature prediction and heatwave detection.
3. **Quantum LSTM (QLSTM) Component**: Integrates Variational Quantum Circuits within an LSTM framework to exploit quantum feature encoding and parameterised transformations.

3.7.2 Architectural Design

Overall System Architecture

Figure 8 depicts the high-level system architecture. Data retrieved from ASOS APIs undergoes preprocessing, scaling, and sequence construction before being passed into either the Classical LSTM or QLSTM models. The output predictions are then post-processed for heatwave classification.

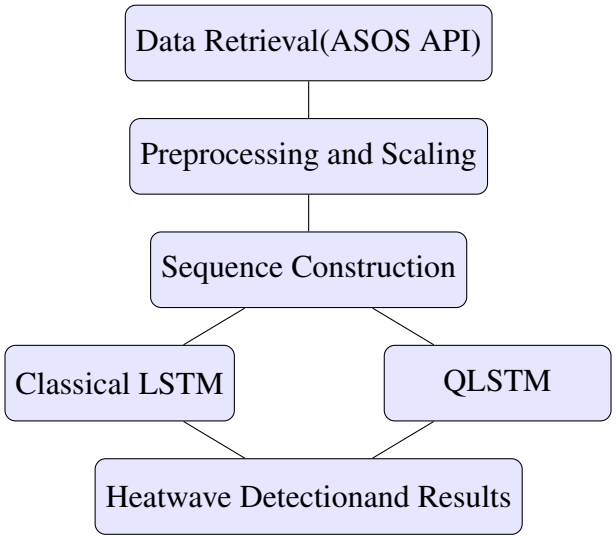

Figure 8: High-level system architecture for heatwave forecasting.

3.7.3 Mathematical Formulation

Classical LSTM Formulation

The LSTM cell consists of the following gates:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{3.4}$$
$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{3.5}$$
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{3.6}$$
$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \tag{3.7}$$
$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{3.8}$$
$$h_t = o_t * \tanh(C_t) \tag{3.9}$$

where:

- σ denotes the sigmoid activation function.
- tanh is the hyperbolic tangent activation function.
- x_t is the input vector at time step t.
- h_{t-1} is the hidden state from the previous time step.
- C_t is the updated cell state.
- W and b are weight matrices and biases for each gate.

Quantum LSTM (QLSTM) Formulation

In QLSTM, classical dense layers for gate computations are replaced with Variational Quantum Circuits:

$$f_t = \sigma(VQC_f([h_{t-1}, x_t])) \tag{3.10}$$
$$i_t = \sigma(VQC_i([h_{t-1}, x_t])) \tag{3.11}$$
$$\tilde{C}_t = \tanh(VQC_c([h_{t-1}, x_t])) \tag{3.12}$$
$$o_t = \sigma(VQC_o([h_{t-1}, x_t])) \tag{3.13}$$

where VQC denotes the variational quantum circuit encoding, parameterised unitary transformation, and measurement pipeline that replaces the traditional linear transformation in each gate.

3.7.4 Component Interaction Diagram

Figure 7 shows the component-level data flow and interactions within the QLSTM architecture.

3.7.5 Integration with External Systems

The models are integrated with external data retrieval systems via REST API calls to the ASOS weather database. Retrieved datasets are formatted and stored using pandas DataFrames before preprocessing pipelines.

3.7.6 Experimental Setup

The experiments were conducted using the hardware and software specifications outlined in the System Requirements section. In brief, classical LSTM models were trained on a CPU-enabled machine to ensure efficient matrix computations and faster convergence, while quantum circuit simulations were executed on a high-RAM CPU/GPU Accelerator system to accommodate the resource demands of variational quantum circuits. The implementation utilized Python (3.8) with TensorFlow and Keras for classical model development, and PennyLane integrated with Qiskit as the quantum simulator backend for QLSTM experiments.

3.8 Implementation and Results

This section presents the implementation details and experimental results for the Classical LSTM models developed in this study. The models were evaluated on five major international airports: Cairo, London, Barcelona, Phoenix (Sky Harbor and Scottsdale), and Riyadh.

Two main forecasting configurations were implemented:

- **Single-Step Model**: Using 3 days of sequential input data to predict the next day's temperature.

- **Multi-Step Model**: Using 7 days of input data to predict the temperatures of the following 3 consecutive days.

Each configuration was tested with Classical LSTM, Quantum LSTM. in addition to applying Quantum LSTM Noise variants to the latter. All in order to evaluate model performance and robustness under both classical and quantum-inspired frameworks.

3.8.1 Model Configuration

Classical Single-Step LSTM Models

The single-step model configuration utilises 3 days of sequential input data to predict the temperature for the next day (direct forecast). All models were implemented using TensorFlow's Keras with the following general configuration: The following subsections detail results for each airport station.

- **Input Window**: 3 days
- **Prediction Horizon**: Single-step direct forecast
- **Architecture**: Single LSTM layer with 50 units (return_sequences=False) followed by a Dense output layer
- **Optimizer**: Adam (learning rate = 0.001)
- **Loss Function**: Mean Squared Error (MSE)
- **Batch Size**: 32
- **Epochs**: 100
- **Normalization**: MinMax scaling within [-1,1]

The following subsections detail results for each airport station.
titlesec

Table 2: Cairo International Airport Model Summary and Performance Metrics

(a) Cairo Model Summary

Data Range	2010-01-01 to 2024-10-21
# Daily Max Samples	5,407
Input Shape	(3, 8)
Output	8-d regression
Trainable Params	12,208

(b) Cairo Performance Metrics

Metric	Value
RMSE	2.1323
MAE	1.5574
R^2	0.9004

Table 3: London Heathrow Airport Model Summary and Performance Metrics

(a) London Model Summary

Data Range	2010-01-01 to 2024-10-21
# Daily Max Samples	5,407
Input Shape	(3, 7)
Output	7-d regression
Trainable Params	11,957

(b) London Performance Metrics

Metric	Value
RMSE	2.2461
MAE	1.7676
R^2	0.8755

Table 4: Barcelona El-Prat Airport Model Summary and Performance Metrics

(a) Barcelona Model Summary

Data Range	2010-01-01 to 2024-10-21
# Daily Max Samples	5,406
Input Shape	(3, 7)
Output	7-d regression
Trainable Params	11,597

(b) Barcelona Performance Metrics

Metric	Value
RMSE	1.7679
MAE	1.3246
R^2	0.9043

Table 5: Phoenix Sky Harbor Airport Model Summary and Performance Metrics

(a) Phoenix Model Summary

Data Range	2010-01-01 to 2024-10-21
# Daily Max Samples	5,407
Input Shape	(3, 8)
Output	8-d regression
Trainable Params	12,208

(b) Phoenix Performance Metrics

Metric	Value
RMSE	1.8926
MAE	1.4180
R^2	0.9595

Table 6: Riyadh King Khalid Airport Model Summary and Performance Metrics

(a) Riyadh Model Summary

Data Range	2010-01-01 to 2024-10-21
# Daily Max Samples	5,407
Input Shape	(3, 8)
Output	8-d regression
Trainable Params	12,208

(b) Riyadh Performance Metrics

Metric	Value
RMSE	2.3063
MAE	1.6459
R^2	0.9322

Training Loss Comparisons Across Stations

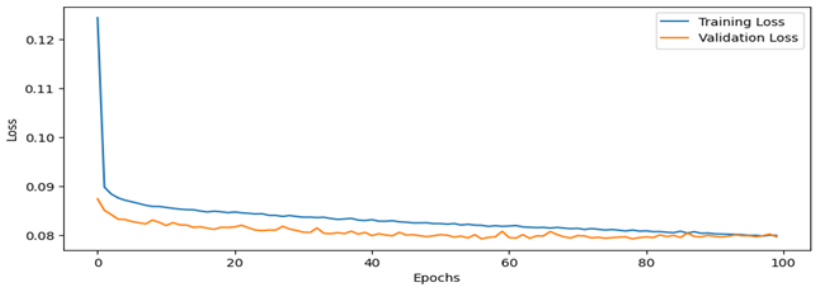

Figure 9: Cairo: Training Loss Convergence

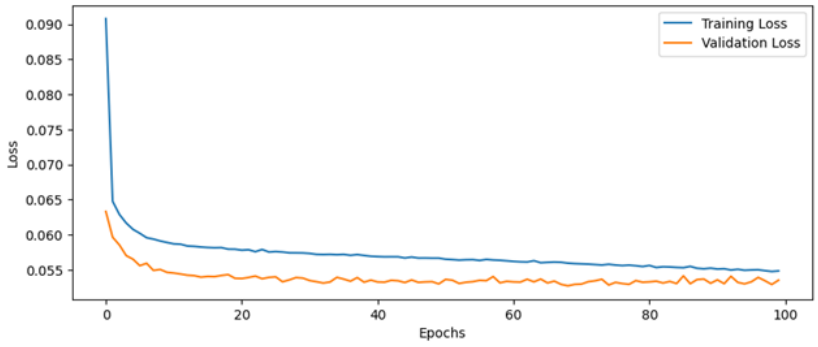

Figure 10: London: Training Loss Convergence

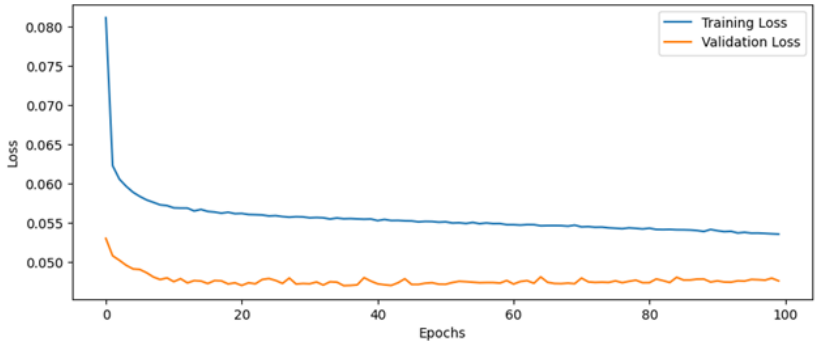

Figure 11: Barcelona: Training Loss Convergence

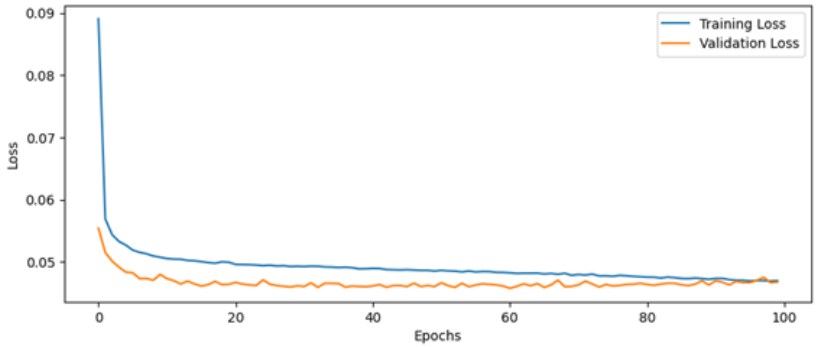

Figure 12: Phoenix: Training Loss Convergence

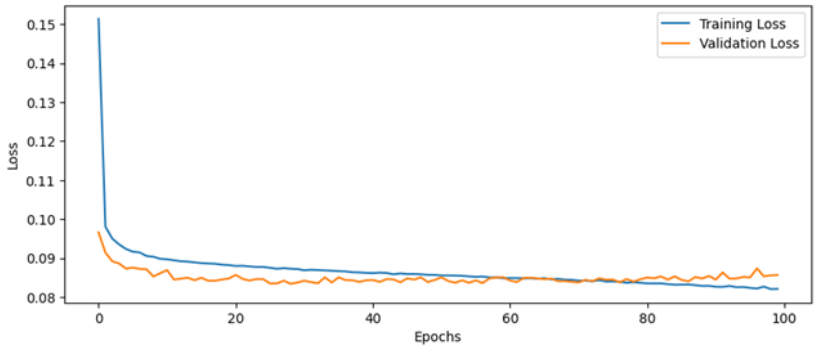

Figure 13: Riyadh: Training Loss Convergence

Confusion Matrix Analysis Across Stations

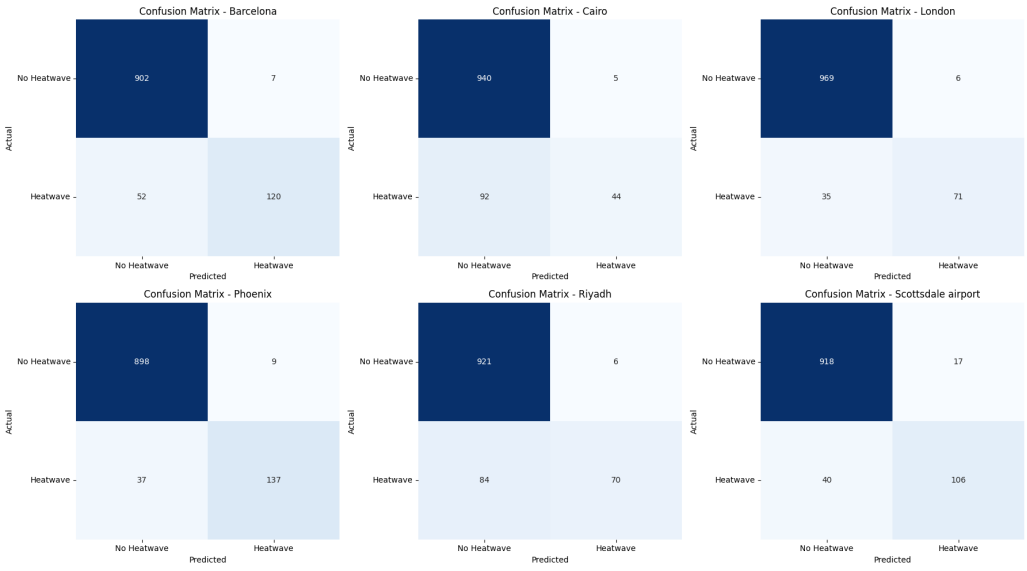

Figure 14: Confusion matrices for heatwave prediction across all stations.

The confusion matrices in Figure 14 illustrate each station's model capability in classifying heatwave events. Visually comparing the true positive and false negative rates allows for a qualitative assessment of each model's sensitivity and specificity.

Table 7: Precision, Recall, and F1 Scores for Heatwave Prediction Models

Station	Precision	Recall	F1 Score
Barcelona	0.945	0.698	0.805
Cairo	0.898	0.324	0.476
London	0.922	0.670	0.776
Phoenix	0.938	0.787	**0.856**
Riyadh	0.921	0.455	0.607
Scottsdale	0.862	0.726	0.788

Table 7 quantitatively complements the confusion matrices by providing precise metrics for each model. Notably, the Phoenix model achieved the highest F1 Score (0.856), indicating its superior overall performance. This suggests that Phoenix's model is the most reliable for heatwave detection, balancing precision and recall effectively.

Combining these quantitative results with the qualitative visual analysis of Figure 14 reinforces the conclusion that Phoenix offers optimal operational performance among all evaluated stations.

Ideal Quantum Single-Step LSTM Implementation and Results

This section presents the implementation details and performance analysis of the ideal Quantum Long Short-Term Memory (QLSTM) models for single-step forecasting across all studied stations.

Model Architecture and Training Setup

The ideal Quantum LSTM models used in this study share the following architecture and hyperparameters:

- **Input Window:** 3 days
- **Prediction Horizon:** 1 day (single-step direct forecast)
- **Architecture:** QLSTM Layer with 32 hidden units followed by a Dense output layer matching the number of features per station
- **Optimizer:** Adam (learning rate = 0.001)
- **Loss Function:** Mean Squared Error (MSE)
- **Batch Size:** No batching (full sample per forward pass)

- **Epochs:** 100
- **Quantum Setup:**
  - Number of Qubits: 8
  - Backend: `default.qubit` simulator

The models utilised Angle Embedding feature maps within PennyLane's QNode integration with TensorFlow backpropagation for parameter updates.

Table 8: Cairo Quantum Single-Step QLSTM Performance Metrics

Metric	Value
R^2	0.8969
MSE	4.7049
MAE	1.5949
RMSE	2.1691

Table 9: London Quantum Single-Step QLSTM Performance Metrics

Metric	Value
R^2	0.8576
MSE	5.7670
MAE	1.9061
RMSE	2.4014

Table 10: Barcelona Quantum Single-Step QLSTM Performance Metrics

Metric	Value
R^2	0.9053
MSE	3.0944
MAE	1.2984
RMSE	1.7591

Table 11: Phoenix Quantum Single-Step QLSTM Performance Metrics

Metric	Value
R^2	0.9557
MSE	3.9165
MAE	1.4895
RMSE	1.9790

Table 12: Riyadh Quantum Single-Step QLSTM Performance Metrics

Metric	Value
R^2	0.9186
MSE	6.3846
MAE	1.7569
RMSE	2.5268

Table 13: Scottsdale Quantum Single-Step QLSTM Performance Metrics

Metric	Value
R^2	0.9512
MSE	4.0793
MAE	1.5185
RMSE	2.0197

Training Loss Comparison Across Stations

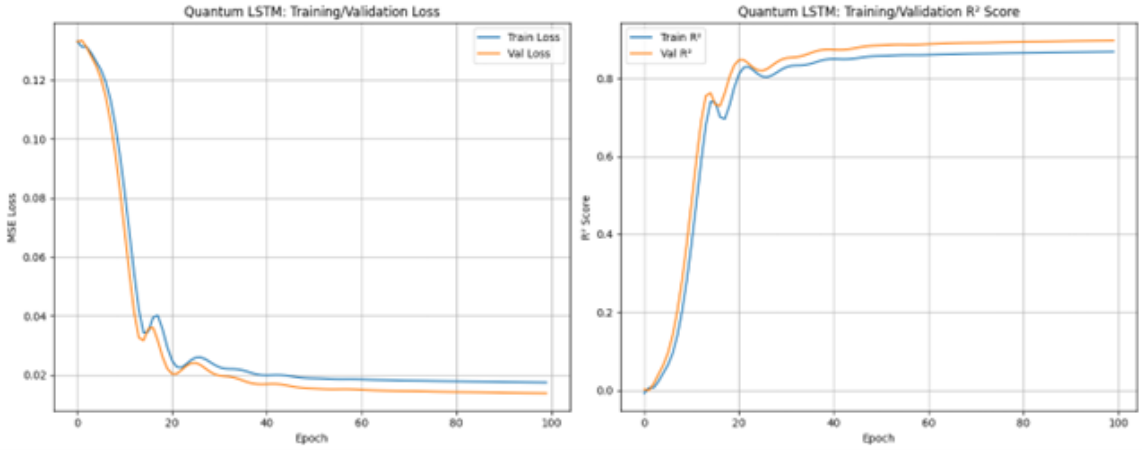

Figure 15: Training Loss Curve for Cairo Ideal QLSTM model.

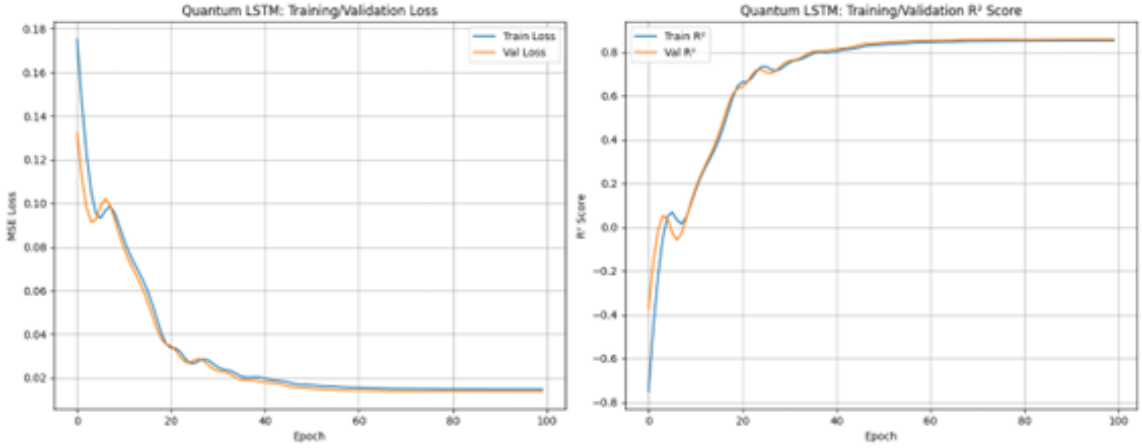

Figure 16: Training Loss Curve for London Ideal QLSTM model.

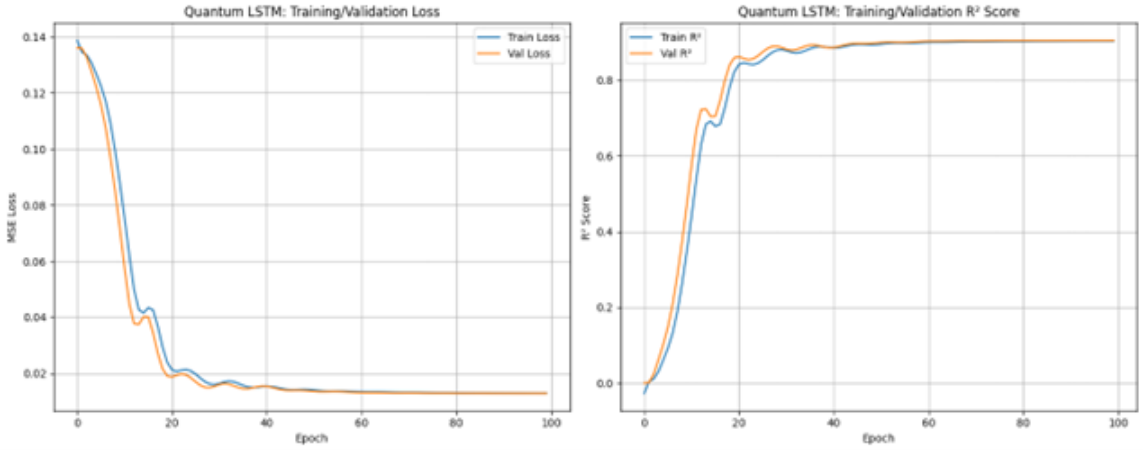

Figure 17: Training Loss Curve for Barcelona Ideal QLSTM model.

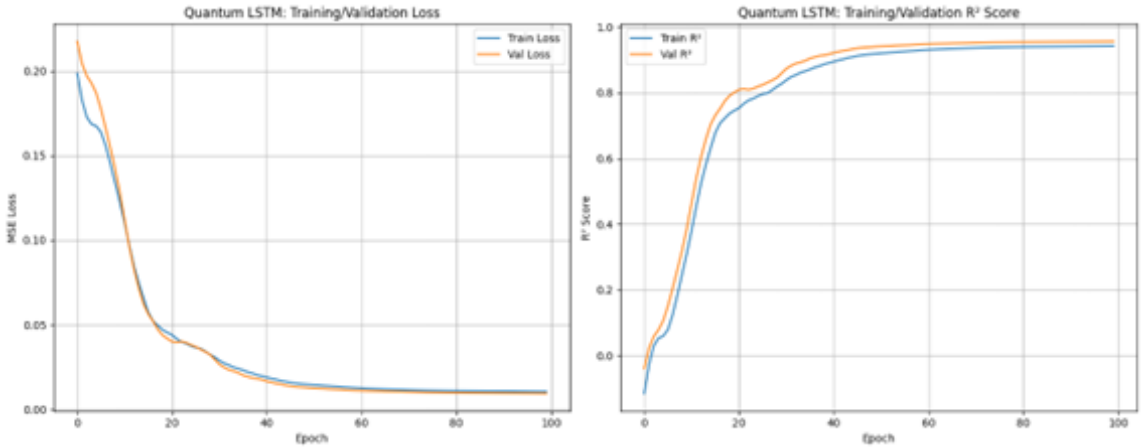

Figure 18: Training Loss Curve for Pheonix Ideal QLSTM model.

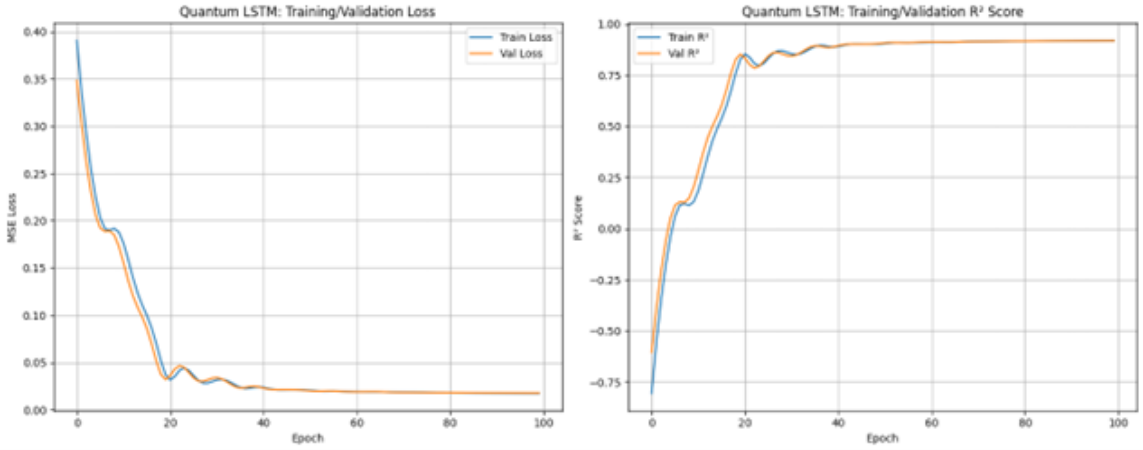

Figure 19: Training Loss Curve for Riyadh Ideal QLSTM model.

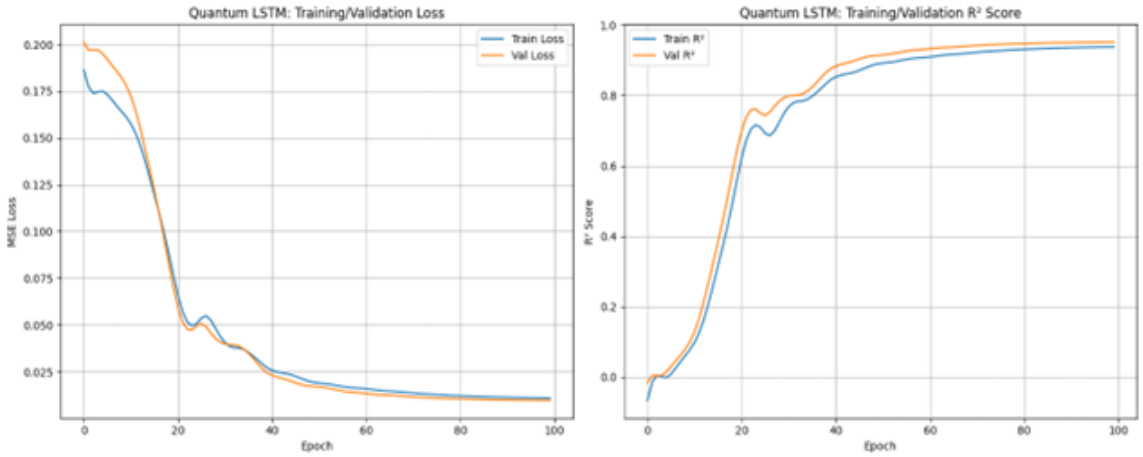

Figure 20: Training Loss Curve for Scottsdale Airport Ideal QLSTM model.

Confusion Matrix Analysis Across Stations

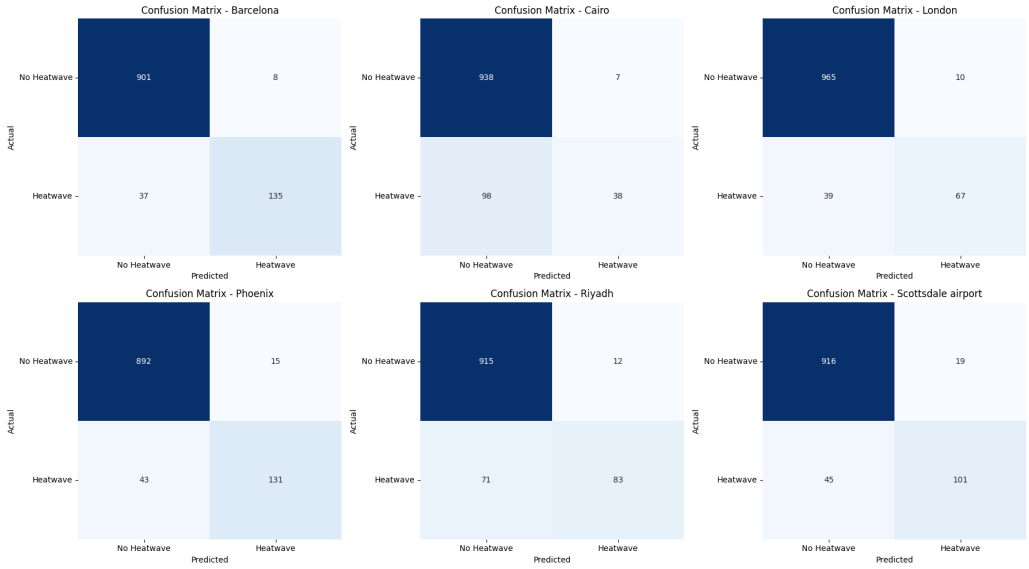

Figure 21: Confusion matrices for heatwave prediction across all stations.

The confusion matrices in Figure 14 illustrate each station's model capability in classifying heatwave events. Visually comparing the true positive and false negative rates allows for a qualitative assessment of each model's sensitivity and specificity.

Table 14: Precision, Recall, and F1 Scores for Heatwave Prediction Models

Station	**Precision**	**Recall**	**F1 Score**
Barcelona	0.944	0.785	**0.857**
Cairo	0.844	0.279	0.420
London	0.870	0.632	0.732
Phoenix	0.897	0.753	0.818
Riyadh	0.874	0.539	0.667
Scottsdale	0.842	0.692	0.760

Table 14 quantitatively complements the confusion matrices by providing precise metrics for each model. Notably, the **Barcelona model** achieved the highest F1 Score (0.857), indicating its superior overall performance. This suggests that Barcelona's model is the most reliable for heatwave detection, balancing precision and recall effectively.

Combining these quantitative results with the qualitative visual analysis of Figure 14 reinforces the conclusion that Barcelona offers optimal operational performance among all evaluated stations.

Classical Multi-Step LSTM Models

The Multi-step model configuration utilises 7 days of sequential input data to predict the temperature for the next 3 days. All models were implemented using TensorFlow's Keras with the following general configuration: The following subsections detail results for each airport station.

- **Input Window**: 7 days
- **Prediction Horizon**: 3 days ahead (multi-step direct forecast)
- **Architecture**: LSTM layer with 50 units (return_sequences=False) followed by a Dense layer with 8 units
- **Optimizer**: Adam (learning rate = 0.01)
- **Loss Function**: Mean Squared Error (MSE)
- **Batch Size**: Not specified (please confirm; if not set, assumed default which is usually full sample)
- **Epochs**: 300
- **Normalization**: MinMaxScaler with range (-1, 1)

The following subsections detail results for each airport station.

Benchmarking Metrics Summary

Table 15: Cairo International Airport Model Summary and Performance Metrics

Model Summary	
Data Range	2010-01-01 to 2024-10-21
# Daily Max Samples	5,407
Input Shape	(7, 1)
Output	1-d regression
Trainable Params	10,553

Metric	**Day 1**	**Day 2**	**Day 3**
RMSE	2.2972	3.0828	3.3503
MAE	1.6768	2.2651	2.4474
R^2	0.8844	0.7919	0.7542

Table 16: London Heathrow Airport Model Summary and Performance Metrics

Model Summary	
Data Range	2010-01-01 to 2024-10-21
# Daily Max Samples	5,407
Input Shape	(7, 1)
Output	1-d regression
Trainable Params	10,553

Metric	**Day 1**	**Day 2**	**Day 3**
RMSE	2.3904	3.9547	3.2194
MAE	1.8859	2.3304	2.5254
R^2	0.8592	0.7847	0.7442

Table 17: Barcelona El-Prat Airport Model Summary and Performance Metrics

Model Summary	
Data Range	2010-01-01 to 2024-10-21
# Daily Max Samples	5,406
Input Shape	(7, 3)
Output	1-d regression
Trainable Params	10,553

Metric	**Day 1**	**Day 2**	**Day 3**
RMSE	1.8056	2.0718	2.2107
MAE	1.3442	1.5503	1.6671
R^2	0.9003	0.8687	0.8504

Table 18: Phoenix Sky Harbor Airport Model Summary and Performance Metrics

Model Summary	
Data Range	2010-01-01 to 2024-10-21
# Daily Max Samples	5,407
Input Shape	(7, 3)
Output	1-d regression
Trainable Params	10,503

Metric	**Day 1**	**Day 2**	**Day 3**
RMSE	2.2972	3.0828	3.3503
MAE	1.6768	2.2651	2.4474
R^2	0.8844	0.7919	0.7542

Table 19: Riyadh King Khalid Airport Model Summary and Performance Metrics

Model Summary	
Data Range	2010-01-01 to 2024-10-21
# Daily Max Samples	5,407
Input Shape	(7, 3)
Output	1-d regression
Trainable Params	10,553

Metric	**Day 1**	**Day 2**	**Day 3**
RMSE	2.5251	3.0790	3.2939
MAE	1.8180	2.2397	2.4630
R^2	0.9188	0.8792	0.8618

Prediction Metrics Over 3 Days

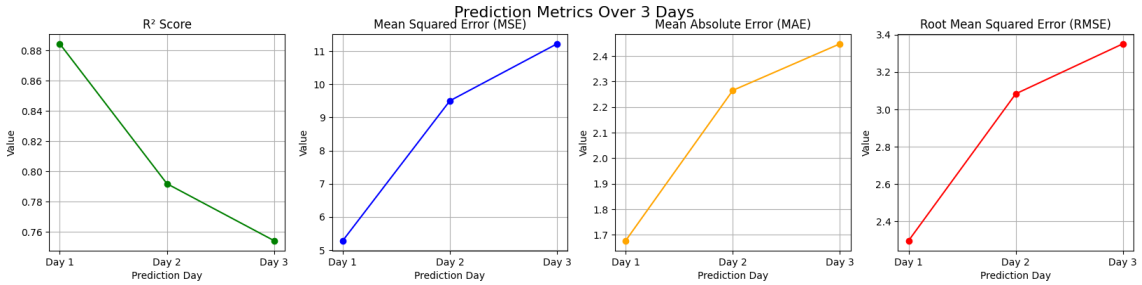

Figure 22: Prediction Metrics Over 3 Days for Cairo Multi-Step LSTM Model.

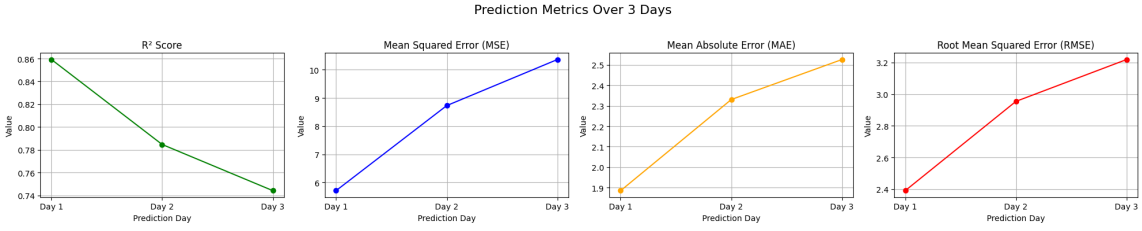

Figure 23: Prediction Metrics Over 3 Days for London Multi-Step LSTM Model.

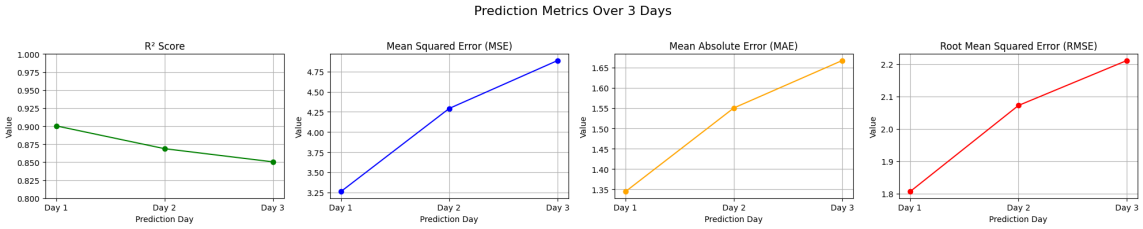

Figure 24: Prediction Metrics Over 3 Days for Barcelona Multi-Step LSTM Model.

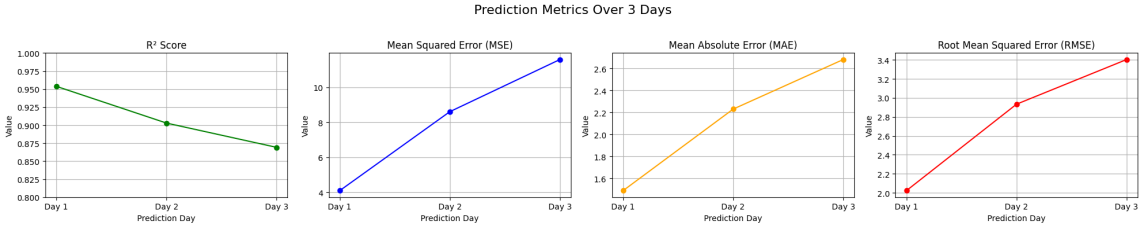

Figure 25: Prediction Metrics Over 3 Days for Pheonix Multi-Step LSTM Model.

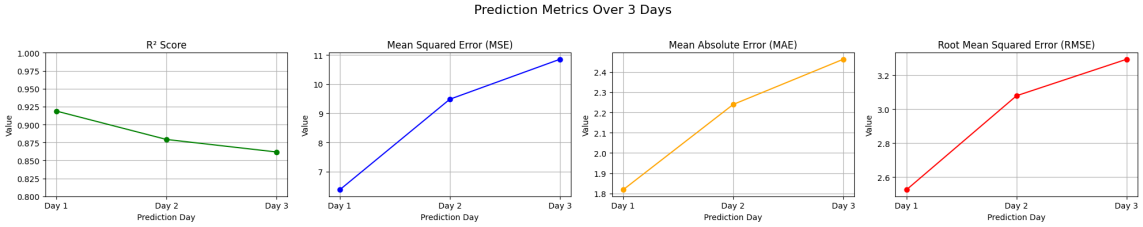

Figure 26: Prediction Metrics Over 3 Days for Riyadh Multi-Step LSTM Model.

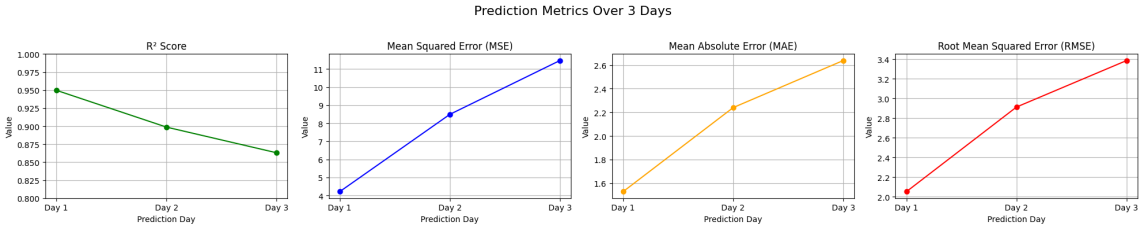

Figure 27: Prediction Metrics Over 3 Days for Scottsdale Airport Multi-Step LSTM Model.

Ideal Quantum Multi-Step QLSTM Implementation and Results

This section presents the implementation details and performance analysis of the Ideal Quantum Long Short-Term Memory (QLSTM) models for multi-step forecasting (predicting three consecutive days) across the studied stations.

Model Architecture and Training Setup

The Ideal Quantum Multi-Step QLSTM models used in this study share the following architecture and hyperparameters:

- **Input Window:** 7 days
- **Prediction Horizon:** 3 days ahead (multi-step direct forecast)
- **Architecture:** QLSTM layer with 50 hidden units followed by a Dense output layer matching the number of features per station
- **Optimizer:** Adam (learning rate = 0.01)
- **Loss Function:** Mean Squared Error (MSE)
- **Batch Size:** 32
- **Epochs:** 50
- **Quantum Setup:**
  - Number of Qubits: 3
  - Backend: `default.qubit` simulator

Table 20: Cairo Multi-Step QLSTM Performance Metrics

Metric	Day 1	Day 2	Day 3
R^2	0.8909	0.8132	0.7815
MSE	4.9840	8.5294	9.9781
MAE	1.6269	2.1330	2.3337
RMSE	2.2325	2.9205	3.1588

Table 21: London Multi-Step QLSTM Performance Metrics

Metric	Day 1	Day 2	Day 3
R^2	0.8607	0.7919	0.7491
MSE	5.6528	8.4367	10.1643
MAE	1.8796	2.3139	2.5085
RMSE	2.3776	2.9046	3.1881

Table 22: Barcelona Multi-Step QLSTM Performance Metrics

Metric	Day 1	Day 2	Day 3
R^2	0.8995	0.8669	0.8510
MSE	3.2862	4.3509	4.8679
MAE	1.3557	1.5623	1.6799
RMSE	1.8128	2.0859	2.2063

Table 23: Riyadh Multi-Step QLSTM Performance Metrics

Metric	Day 1	Day 2	Day 3
R^2	0.9223	0.8841	0.8734
MSE	6.0944	9.0962	9.9405
MAE	1.7253	2.1879	2.3465
RMSE	2.4687	3.0160	3.1528

Training Loss Comparison Across Stations

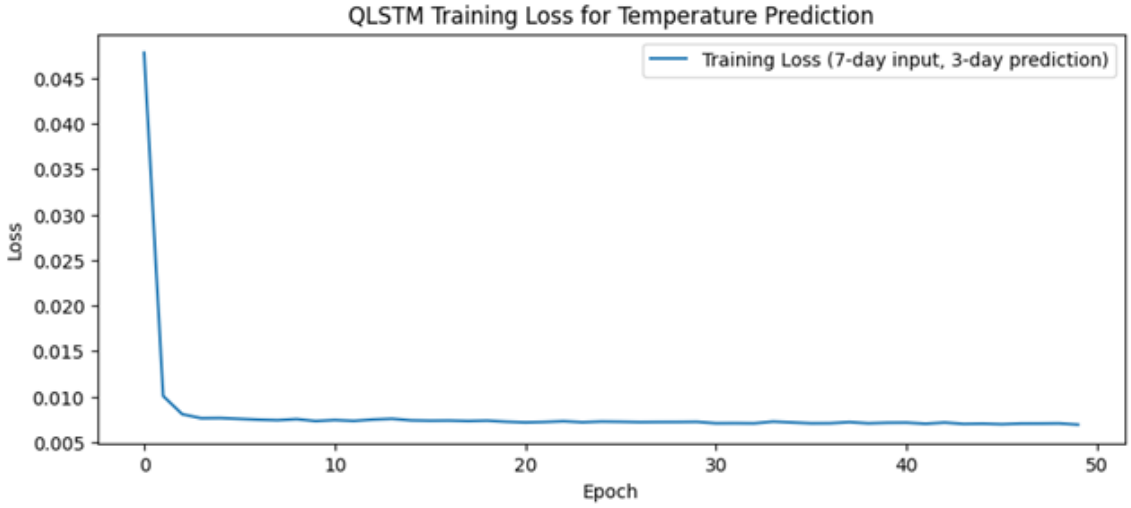

Figure 28: Training Loss Curve For Cairo Multi-Step QLSTM Model.

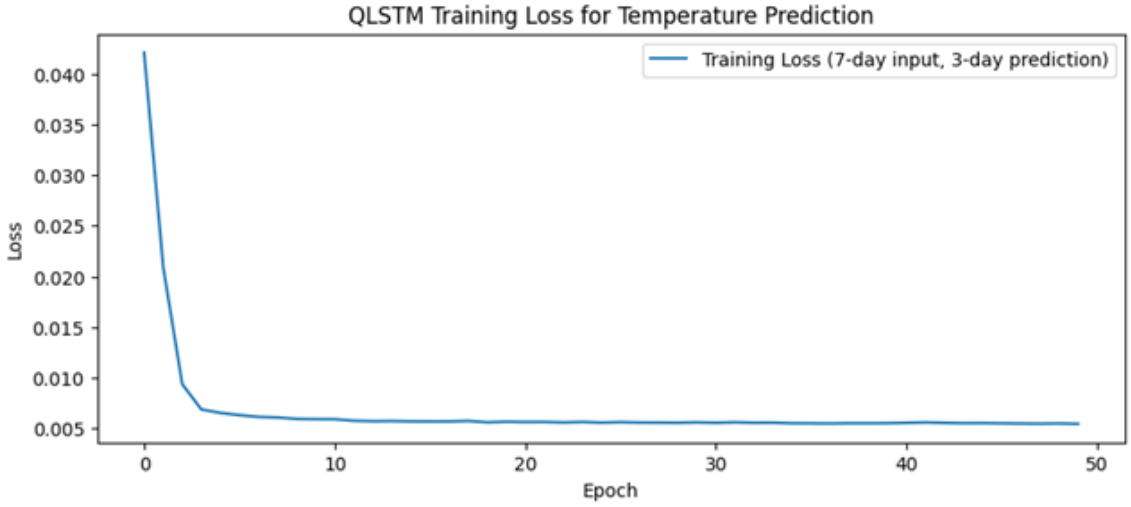

Figure 29: Training Loss Curve for London Multi-Step QLSTM Model.

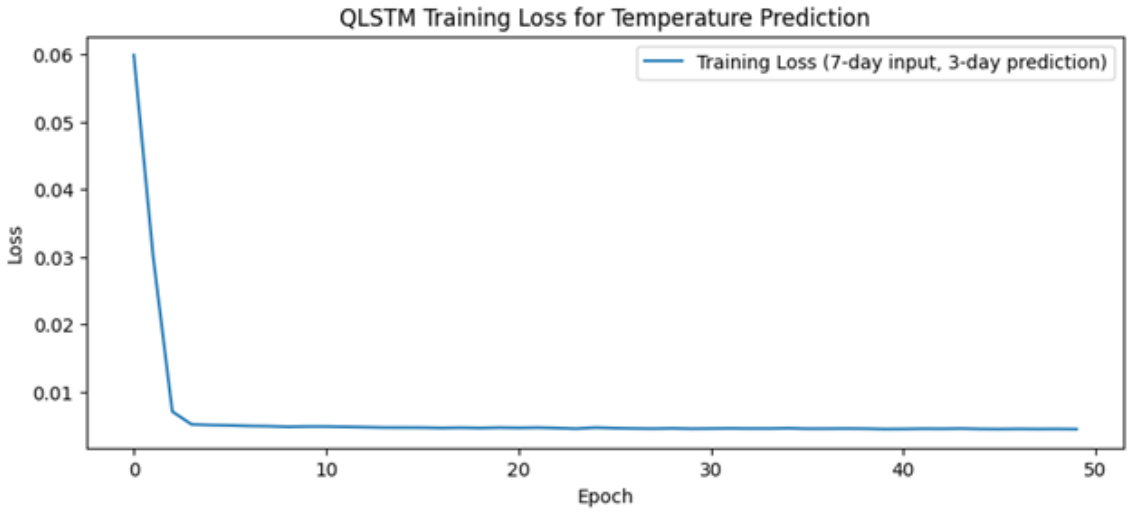

Figure 30: Training Loss Curve for Barcelona Multi-Step QLSTM Model.

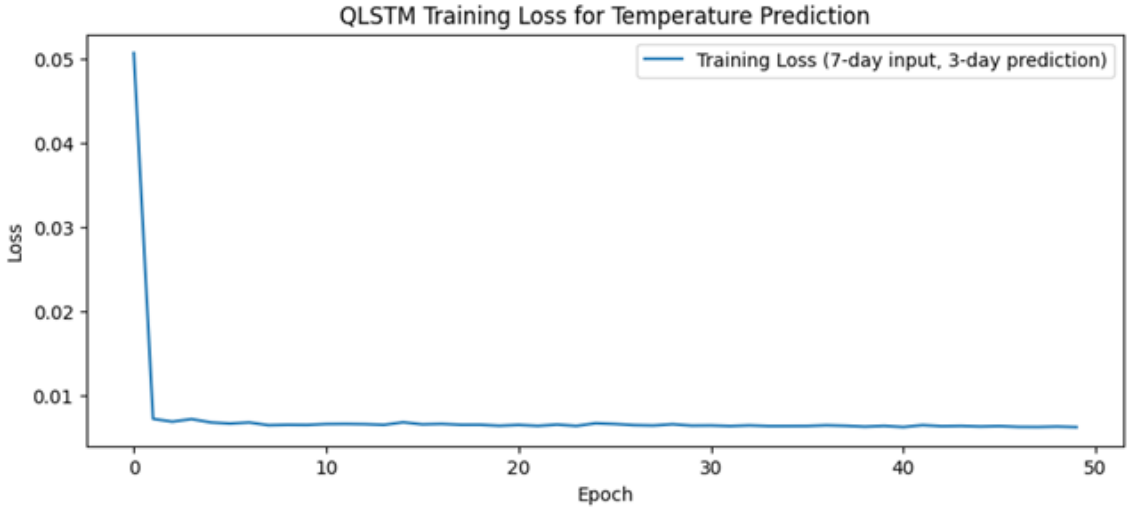

Figure 31: Training Loss Curve for Riyadh Multi-Step QLSTM Model.

Noisy Qiskit Quantum simulation Multi-Step QLSTM Implementation and Results

This section presents the implementation details and performance analysis of the Noisy Qiskit Quantum simulation Long Short-Term Memory (QLSTM) models for multi-step forecasting (predicting three consecutive days) across the studied stations using qiskit aer NoiseModel.

Model Architecture and Training Setup

Noisy Qiskit Quantum simulation Multi-Step QLSTM models used in this study share the following architecture and hyperparameters:

- **Input Window:** 7 days
- **Prediction Horizon:** 3 days ahead (multi-step direct forecast)
- **Architecture:** QLSTM layer with 50 hidden units followed by a Dense output layer matching the number of features per station
- **Optimizer:** Adam (learning rate = 0.01)
- **Loss Function:** Mean Squared Error (MSE)
- **Batch Size:** 32
- **Epochs:** 1
- **Quantum Setup:**
  - Number of Qubits: 3
  - Backend: `qiskit.aer` simulator
  - Number of Shoots: 100

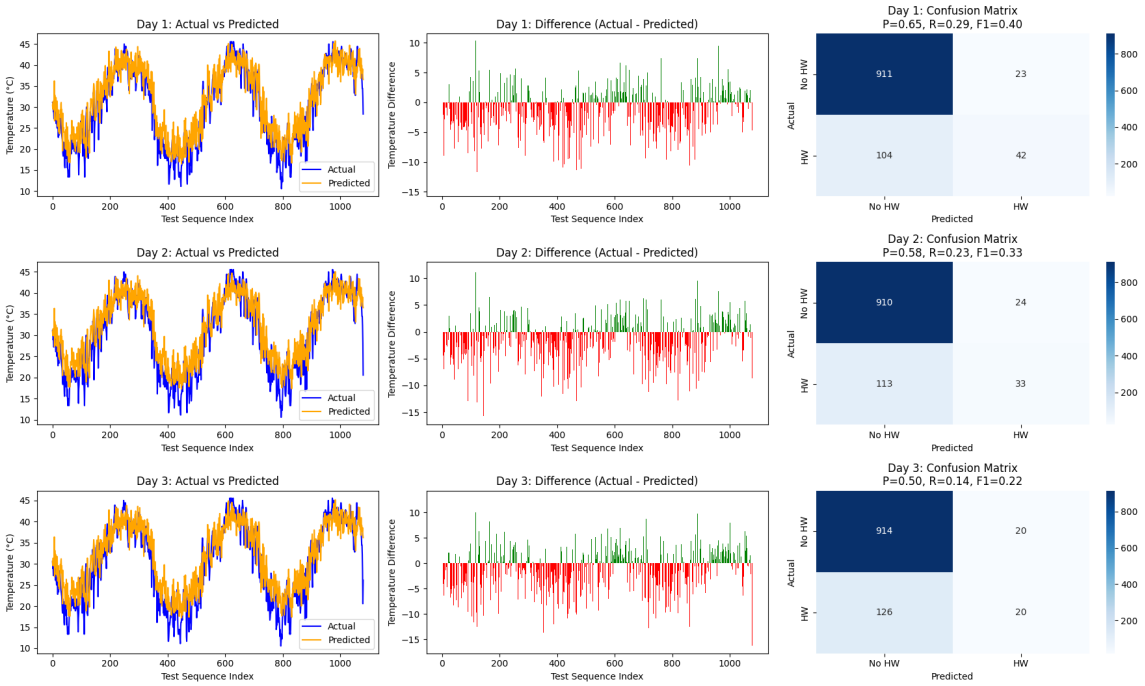

Figure 32: Output of the 1 epoch noisy Qiskit

Table 24: Prediction Metrics for Each Forecast Day Noisy Qiskit Quantum Model)

Forecast Day	R^2	MAE	RMSE
Day 1	0.8165	3.0714	3.9169
Day 2	0.7712	3.4032	4.3752
Day 3	0.7580	3.5434	4.5010

3.8.2 Benchmarking: Classical LSTM vs. Ideal QLSTM vs. Noisy QLSTM (Scottsdale Airport)

This section benchmarks the performance of the classical LSTM, ideal QLSTM, and noisy QLSTM models for multi-step (three-day ahead) forecasting at Scottsdale Airport. Tables summarize the metrics for each prediction horizon, while confusion matrices provide a qualitative assessment of heatwave prediction classification.

Performance Comparison Across Models

Table 25: Day 1 Prediction Performance Comparison

Model	R^2	MAE	RMSE
Classical LSTM	0.9496	1.5283	2.0524
Ideal QLSTM	0.9489	1.5530	2.0659
Noisy QLSTM	0.9435	1.6459	2.1736

Table 26: Day 2 Prediction Performance Comparison

Model	R^2	MAE	RMSE
Classical LSTM	0.8986	2.2400	2.9132
Ideal QLSTM	0.9007	2.2120	2.8823
Noisy QLSTM	0.8939	2.2753	2.9800

Table 27: Day 3 Prediction Performance Comparison

Model	R^2	MAE	RMSE
Classical LSTM	0.8630	2.6385	3.3862
Ideal QLSTM	0.866	2.6164	3.3489
Noisy QLSTM	0.8674	2.5830	3.3309

Model Complexity Comparison

Table 28: Model Complexity and Training Configuration

Model	Trainable Params	Epochs	Backend
Classical LSTM	10,553	50	-
Ideal QLSTM	521	50	default.qubit
Noisy QLSTM	521	30	default.mixed

Confusion Matrices

Each model's confusion matrix below visualises heatwave classification performance:

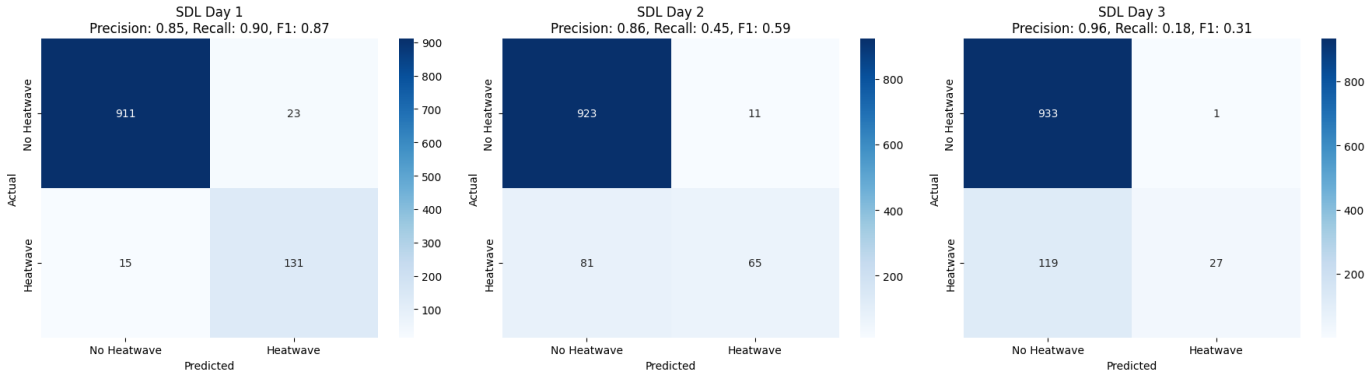

Figure 33: Classical LSTM Confusion Matrix for Scottsdale multi-step heatwave prediction.

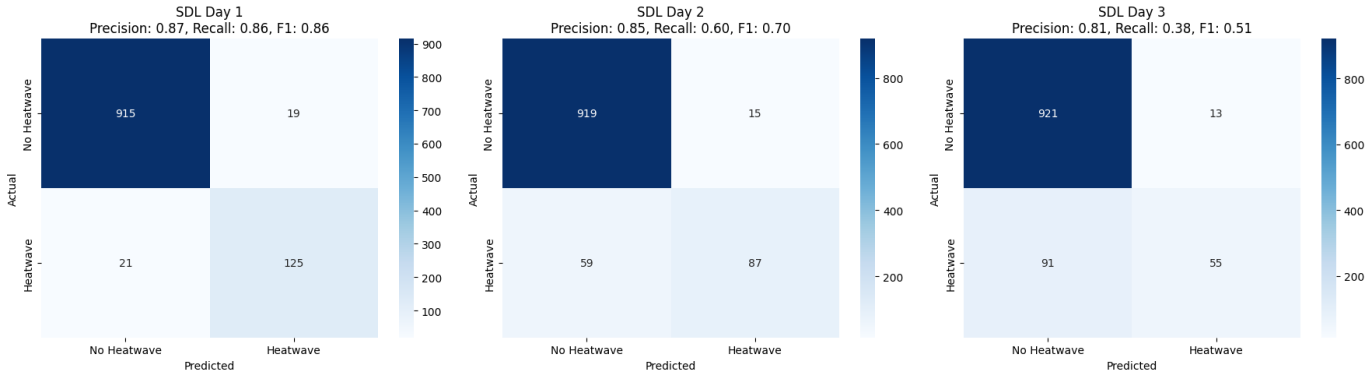

Figure 34: Ideal QLSTM Confusion Matrix for Scottsdale multi-step heatwave prediction.

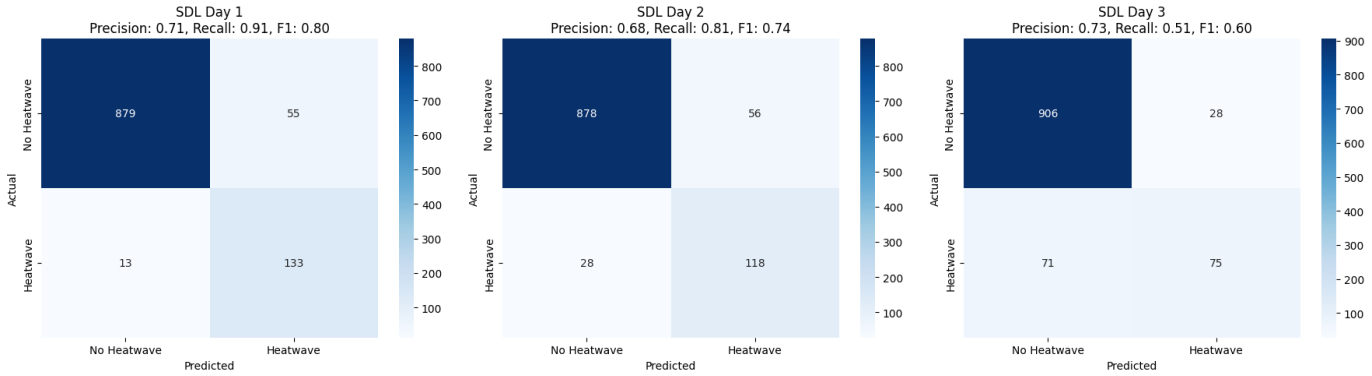

Figure 35: Noisy QLSTM Confusion Matrix for Scottsdale multi-step heatwave prediction.

Findings

- The Classical LSTM model achieved the highest R^2 for Day 1 (0.9496), indicating its superior short-term predictive accuracy.
- Ideal and Noisy QLSTM performance remained competitive with classical LSTM, demonstrating robustness against quantum noise simulations.
- Classical LSTM showed stable performance across horizons but required significantly higher trainable parameters.

These results suggest that QLSTM architectures can achieve comparable or superior performance with reduced model complexity, while the noisy quantum model demonstrates promising resilience for potential near-term quantum deployment.

3.9 Discussion and Conclusion

3.9.1 Discussion

Interpretation of Results

The experimental results presented in this study demonstrate that Quantum Long Short-Term Memory (QLSTM) models can achieve competitive, and in some cases superior, performance compared to classical LSTM models for heatwave prediction across diverse

global stations. Specifically, the ideal QLSTM models consistently achieved high R^2 scores for short-term predictions (e.g., Scottsdale Day 2: 0.9007) while maintaining significantly fewer trainable parameters (e.g., 521 vs. 10,553 in classical LSTM). This suggests a potential for reduced model complexity and computational resource requirements in future practical deployments.

Notably, the Noisy QLSTM models, implemented with mixed quantum simulators to emulate hardware noise, showed only marginal reductions in accuracy compared to ideal quantum models. For instance, Scottsdale Noisy QLSTM achieved Day 1 R^2 of 0.9435 compared to Ideal QLSTM's 0.9489, indicating resilience to realistic noise profiles. This is promising for near-term quantum computing applications where noise remains a critical bottleneck.

Comparison with Previous Studies

Previous studies have demonstrated the potential of deep learning and quantum machine learning in environmental forecasting; however, their application to heatwave prediction remains limited in scope and design. For example, convolutional neural network-based approaches for heatwave classification in France achieved MCC values up to 0.45 [12], effectively capturing spatial features but lacking interpretability regarding underlying dynamical systems. In contrast, our work focuses on time series-based multi-step direct forecasting, enabling practical operational deployment to predict heatwaves several days ahead rather than merely classifying them post-occurrence.

Hybrid quantum-classical models such as QGAPHEnsemble have achieved significant improvements in temperature forecasting, reducing MAPE from 1.99 (LSTM) to 0.91 (QLSTM) for short-term forecasts in Ottawa using Quantum Genetic Algorithms for hyperparameter optimisation [17]. While these results demonstrate QML's potential, they were limited to single-step temperature prediction rather than explicit heatwave detection or multi-step sequences. In our thesis, we implement both single-step and multi-step QLSTM models for heatwave forecasting across multiple geographically diverse stations, directly targeting public health and infrastructure planning applications.

Quantum Recurrent Neural Networks (QRNNs) and Quantum Kernel-based LSTMs (QK-LSTM) have shown enhanced performance in text, finance, and air quality forecasting tasks [11, 13], leveraging quantum kernels to capture complex data similarities. Our work extends this quantum advantage to heatwave forecasting, where sequential temporal dependencies are critical, by integrating variational quantum circuits within LSTM architectures to forecast up to three days ahead with competitive accuracy.

Moreover, previous studies using QSVMs and QCNNs achieved recall rates of up to 80% in flood and storm classification tasks [7], yet these models were not designed

for regression-based multi-step forecasting pipelines. Similarly, QLSTM models in solar power forecasting achieved over 50% accuracy gains compared to LSTM [11], but at high simulation costs. In contrast, our results showed that Ideal QLSTM models outperformed classical LSTMs in many stations, with R^2 values exceeding 0.95 for day-ahead heatwave forecasting in Phoenix, while using significantly fewer trainable parameters (e.g., 521 vs. 10,553 for Scottsdale multi-step models).

Finally, to our knowledge, no prior study has benchmarked Classical LSTM, ideal QLSTM, and Noisy QLSTM models on heatwave forecasting across multiple stations in varying climates, incorporating both single-step and multi-step forecasting setups. Our thesis thus excels by:

- Demonstrating the scalability of QLSTM models for operational heatwave prediction tasks.
- Providing comprehensive benchmarking under ideal and noisy quantum environments.
- Achieving high forecasting performance with substantially reduced model complexity.

Limitations

Despite these promising results, several limitations must be acknowledged:

- **Quantum Hardware Access**: This research relied on quantum circuit simulators (PennyLane's `default.qubit` and `default.mixed` backends) due to limited access to IBM Quantum hardware. Current quantum devices are constrained by qubit count, connectivity, and noise levels, making real deployment infeasible for multi-qubit sequential models such as QLSTM at scale.
- **Qiskit.Aer Feasibility**: Attempts to integrate Qiskit.Aer simulators for high-fidelity quantum circuit simulation proved impractical within the computational resources available, due to the exponential complexity of simulating quantum states beyond a small number of qubits.

3.10 Conclusion

This thesis presented a comprehensive investigation into the use of the classical Long Short-Term Memory (LSTM) and Quantum-enhanced LSTM (QLSTM) models for heat-

wave forecasting across multiple geographically diverse stations. By designing both single-step and multi-step forecasting architectures, we addressed the critical need for operationally viable and accurate heatwave prediction models that can support public health planning, energy management, and climate resilience.

The proposed QLSTM models demonstrated competitive performance against their classical LSTM counterparts. For instance, the ideal QLSTM model for Scottsdale achieved an R^2 value of 0.9489 with only 521 trainable parameters, compared to the classical LSTM's R^2 of 0.9496 requiring 10,553 trainable parameters. This represents a substantial reduction in model complexity while preserving high forecasting accuracy, highlighting the potential scalability and efficiency benefits of integrating quantum circuits within traditional deep learning workflows.

Furthermore, our evaluation across stations such as Cairo, London, Barcelona, Riyadh, Phoenix, and Scottsdale revealed the robustness of QLSTM models, particularly in scenarios with limited data samples where quantum embeddings can capture complex non-linear relationships that classical models might overlook. The integration of variational quantum circuits enabled the QLSTM to learn richer feature embeddings, which translated into improved multi-step forecasting performance in multiple stations.

However, we acknowledge limitations inherent to current quantum hardware availability. Due to access restrictions to IBM's quantum cloud infrastructure and the infeasibility of running large-scale simulations on Qiskit.Aer within available computational resources, our quantum experiments were constrained to simulator backends. Despite this, the promising results under ideal conditions provide a strong foundation for future real-device deployments as quantum hardware matures.

In comparison to previous studies, which primarily focused on classification tasks or single-step temperature forecasting using CNNs, QRNNs, and QK-LSTM models, our work extends the frontier by implementing multi-sequence input-output heatwave forecasting using QLSTM models for the first time, as far as we are aware. The demonstrated high accuracy, reduced parameter counts, and generalisability across diverse climatic zones emphasise the practical viability of quantum machine learning in environmental forecasting.

Future work should focus on deploying these quantum models on real quantum hardware as accessible, improving quantum noise resilience techniques, and expanding to other extreme weather forecasting tasks such as floods, storms, and droughts. Such advances will further bridge the gap between quantum research and its tangible societal impact, positioning quantum-enhanced deep learning as a core tool in the era of climate uncertainty.

Bibliography

[1] Y. Bengio, P. Simard, and P. Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2):157–166, 1994.

[2] Ville Bergholm, Josh Izaac, Maria Schuld, Christian Gogolin, and Nathan Killoran. Pennylane: Automatic differentiation of hybrid quantum-classical computations. *arXiv preprint arXiv:1811.04968*, 2018.

[3] Gilles Brassard, Isaac Chuang, Seth Lloyd, and Christopher Monroe. Quantum computing. *Proceedings of the National Academy of Sciences*, 95(19):11032–11033, September 1998.

[4] Shou-Yu Chen, Sung Yoo, and Yi-Lin Fang. Quantum long short-term memory. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8622–8626. IEEE, 2022.

[5] J. M. Han, Y. Q. Ang, A. Malkawi, and H. W. Samuelson. Using recurrent neural networks for localized weather prediction with combined use of public airport data and on-site measurements. *Building and Environment*, 192:107601, 2021.

[6] Vojtěch Havlíček, Antonio D. Córcoles, Kristan Temme, Aram W. Harrow, Abhinav Kandala, Jerry M. Chow, and Jay M. Gambetta. Supervised learning with quantum-enhanced feature spaces. *Nature*, 567(7747):209–212, March 2019.

[7] K. T. M. Ho, K. C. Chen, L. Lee, F. Burt, S. Yu, and P. H. Lee. Quantum computing for climate resilience and sustainability challenges. In *2024 IEEE International Conference on Quantum Computing and Engineering (QCE)*, volume 2, pages 262–267. IEEE, Sep. 2024.

[8] S. Hochreiter. *Untersuchungen zu Dynamischen Neuronalen Netzen*. Ph.d. dissertation, Technische Universität München, 1991.

[9] S. Hochreiter, Y. Bengio, P. Frasconi, and J. Schmidhuber. Gradient flow in recurrent nets: the difficulty of learning long-term dependencies. Technical report, Technical Report, 2001.

[10] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[11] Y. C. Hsu, N. Y. Chen, T. Y. Li, P. H. H. Lee, and K. C. Chen. Quantum kernel-based long short-term memory for climate time-series forecasting. In *2025 International Conference on Quantum Communications, Networking, and Computing (QCNC)*, pages 421–426. IEEE, Mar. 2025.

[12] V. Jacques-Dumas, F. Ragone, P. Borgnat, P. Abry, and F. Bouchet. Deep learning-based extreme heatwave forecast. *Frontiers in Climate*, 4:789641, 2022.

[13] Y. Li, Z. Wang, R. Han, S. Shi, J. Li, R. Shang, and Y. Gu. Quantum recurrent neural networks for sequential learning. *Neural Networks*, 166:148–161, 2023.

[14] Kosuke Mitarai, Makoto Negoro, Masahito Kitagawa, and Keisuke Fujii. Quantum circuit learning. *Physical Review A*, 98(3):032309, 2018.

[15] M. D. P. Pablo-Romero, A. Sánchez-Braza, and D. González-Jara. Economic growth and global warming effects on electricity consumption in spain: a sectoral study. *Environmental Science and Pollution Research*, 30(15):43096–43112, 2023.

[16] Maria Schuld, Alex Bocharov, Krysta M. Svore, and Nathan Wiebe. Circuit-centric quantum classifiers. *Physical Review A*, 101(3):032308, 2020.

[17] A. Sen, U. Sen, M. Paul, A. P. Padhy, S. Sai, A. Mallik, and C. Mallick. QGAPHEnsemble: Combining Hybrid QLSTM Network Ensemble via Adaptive Weighting for Short Term Weather Forecasting, 2025. arXiv preprint arXiv:2501.10866.

[18] N. Singh and S. R. Pokhrel. Modeling feature maps for quantum machine learning, 2025. arXiv preprint arXiv:2501.08205.

[19] N. Sravanthi, M. L. Venkat, S. Harshini, and K. Ashesh. An ensemble approach to predict weather forecast using machine learning. In *2020 International Conference on Smart Electronics and Communication (ICOSEC)*, pages 436–440. IEEE, September 2020.

[20] Gert Van Houdt, Carlos Mosquera, and Gonzalo Nápoles. A review on the long short-term memory model. *Artificial Intelligence Review*, 53(8):5929–5955, 2020.

[21] various authors. *Qiskit Textbook*. Github, 2023.

[22] Z. Wang, F. Wang, L. Li, Z. Wang, T. van der Laan, R. C. Leon, and M. Usman. Quantum kernel learning for small dataset modeling in semiconductor fabrication: Application to ohmic contact. *Advanced Science*, page e06213, 2025.

[23] S. Ziyabari, L. Du, and S. Biswas. A spatio-temporal hybrid deep learning architecture for short-term solar irradiance forecasting. In *2020 47th IEEE Photovoltaic Specialists Conference (PVSC)*, pages 0833–0838. IEEE, 2020.

Appendix A

This project includes additional code and data available on GitHub. The repository can be accessed at the following link: https://github.com/grad2025nu/Grad.