EDRAAK
Vol. (2026), 2026, pp. 136–147
ISSN: 3078-8412

Research Article

Comprehensive Performance Evaluation of Deep Learning Algorithms for Multi-Variable Climate Prediction: RNN, LSTM, and GRU Analysis

Mustafa Mahmood Akawee[1], Raed A. Hasan [2,3] *, Munif Ahmed Abdullah[4], Ali Mohammed Salih[2], Khalil F. Yassin [2], Mehdi Qahraman Fakhruldin[3], Husniyah Jasim[4]

[1] *Department of Computer Science, Al-Imam Al-Adham University College, 10011, Baghdad, Iraq*

[2] *Renewable Energy Research Unit, Northern Technical University, Kirkuk, Iraq*

[3] *Department of Robotics and Artificial intelligence techniques, Al-Hawija Polytechnique College, Northern Technical University, Kirkuk, Iraq*

[4] *Engineering Department of Medical Devices Technology, Al-Hawija Polytechnique College, Northern Technical University, Kirkuk, Iraq*

ARTICLEINFO

Article History
Received 01 Jun 2026
Accepted: 15 Jul 2026
Revised 14 Aug 2026
Published 01 Sep 2026

Keywords
Deep Learning,
Climate Prediction,
RNN,
LSTM,
GRU,
Performance Comparison,
Time Series Forecasting,
Machine Learning.

ABSTRACT

The selection of appropriate deep learning architectures for climate prediction remains a critical challenge in atmospheric sciences, with different algorithms showing varying performance across climate variables and geographical regions. This study presents a comprehensive comparative analysis of three prominent deep learning architectures Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), and Gated Recurrent Unit (GRU) for multi-variable climate prediction in the Middle East region. Using 42 years (1981-2022) of high-resolution MERRA-2 reanalysis data across nine Middle Eastern capitals, we evaluated the performance of these algorithms in predicting six key climate variables: maximum and minimum temperature, relative humidity, wind speed, solar radiation, and precipitation. Our analysis reveals significant performance differences across algorithms, with GRU demonstrating superior computational efficiency (14% faster training than LSTM) while maintaining competitive accuracy. LSTM excelled in capturing long-term dependencies for temperature variables (R^2 > 0.95), while RNN showed adequate performance for simpler patterns but struggled with complex temporal relationships. The study provides detailed performance metrics, computational requirements, and practical guidelines for algorithm selection based on specific climate prediction tasks. Key findings indicate that algorithm choice should be tailored to the prediction target, with temperature variables favoring LSTM, precipitation benefiting from GRU's efficiency, and humidity showing comparable performance across all architectures. These results provide essential guidance for operational weather forecasting systems and climate modeling applications, contributing to the optimization of deep learning approaches in atmospheric sciences.

1. INTRODUCTION

The rapid advancement of deep learning technologies has revolutionized time series forecasting across numerous domains, with climate prediction emerging as one of the most promising applications [1]. The inherent complexity of climate systems, characterized by nonlinear dynamics, multi-scale interactions, and long-term dependencies, presents unique challenges that traditional statistical methods often struggle to address effectively [2]. Deep learning architectures, particularly recurrent neural networks and their variants, offer powerful tools for capturing these complex temporal patterns in climate data [3].
Among the various deep learning architectures available for sequential data modeling, three approaches have gained particular prominence in climate applications: traditional Recurrent Neural Networks (RNN), Long Short-Term Memory networks (LSTM), and Gated Recurrent Units (GRU) [4]. Each architecture possesses distinct characteristics in terms of computational complexity, memory requirements, and ability to capture temporal dependencies of varying lengths [5]. However, the relative performance of these architectures for climate prediction tasks remains incompletely understood, particularly across different climate variables and geographical regions.

*Corresponding author email: raed.isc.sa@ntu.edu.iq
DOI: https://doi.org/10.70470/EDRAAK/2026/011

The selection of appropriate deep learning architectures for operational climate prediction systems requires comprehensive understanding of their relative strengths and limitations. Factors such as prediction accuracy, computational efficiency, training stability, and generalization capability all influence the practical utility of different approaches [6]. Furthermore, the performance of these algorithms may vary significantly depending on the specific climate variable being predicted, the temporal resolution of the data, and the geographical characteristics of the study region [7].

This study addresses the critical need for systematic comparison of deep learning architectures in climate prediction applications. By conducting a comprehensive evaluation of RNN, LSTM, and GRU algorithms using consistent datasets, evaluation metrics, and experimental conditions, we provide essential guidance for researchers and practitioners in selecting optimal approaches for their specific applications [8].

2. LITERATURE REVIEW

2.1 Deep Learning in Climate Prediction

The application of deep learning techniques to climate and weather prediction has gained significant momentum over the past decade. Early pioneering work by Grover et al. (2015) demonstrated the potential of deep learning models to complement traditional numerical weather prediction systems [9,10]. Their hybrid approach combined the physical understanding embedded in traditional models with the pattern recognition capabilities of neural networks, achieving improved forecast accuracy for multiple meteorological variables [11].

Recent advances in deep learning-based weather prediction (DLWP) have shown remarkable progress. Nguyen et al. (2023) developed ClimateLearn, a comprehensive benchmarking framework that evaluated various deep learning architectures for weather forecasting and climate downscaling tasks [12]. Their findings highlighted the superior performance of recurrent architectures, particularly LSTM and GRU models, in capturing temporal dependencies in climate data.

The integration of convolutional neural networks (CNNs) with recurrent architectures has also shown promise. Munandar and Surbakti (2025) proposed a hybrid deep learning framework that integrates temporal, spatial, and environmental data to capture the complex dynamics of Earth's climate system [13]. Their approach demonstrated significant improvements in prediction accuracy compared to traditional methods, particularly for extreme weather events.

2.2 LSTM and GRU Architectures in Time Series Forecasting

Long Short-Term Memory networks, introduced by Hochreiter and Schmidhuber (1997), were specifically designed to address the vanishing gradient problem in traditional RNNs [14]. The LSTM architecture incorporates three gating mechanisms: input gates, forget gates, and output gates, which enable the network to selectively retain or discard information over long sequences. This capability makes LSTM particularly suitable for climate prediction tasks where long-term dependencies are crucial.

Gated Recurrent Units, proposed by Cho et al. (2014), offer a simplified alternative to LSTM with only two gating mechanisms: reset and update gates [15]. Despite their simpler structure, GRU models have demonstrated competitive performance in many sequence modeling tasks while requiring fewer parameters and computational resources. Recent comparative studies have shown that GRU models can outperform LSTM in certain scenarios, particularly when dealing with shorter sequences or limited training data [16].

Uluocak and Bilgili (2024) conducted a comprehensive comparison of LSTM and GRU models for daily air temperature forecasting, revealing that both architectures achieved comparable performance, with GRU showing slight advantages in computational efficiency [17]. Their study emphasized the importance of proper hyperparameter tuning and data preprocessing in achieving optimal performance.

2.3 Hybrid Deep Learning Models

The concept of hybrid deep learning models has gained traction as researchers seek to combine the strengths of different architectures. Yunita et al. (2025) conducted a comprehensive performance analysis of various neural network architectures, including vanilla RNN, LSTM, GRU, and six hybrid configurations [18]. Their results demonstrated that hybrid models, particularly LSTM-GRU combinations, achieved superior performance across multiple evaluation metrics.

In the context of climate prediction, several studies have explored hybrid approaches. A recent study by researchers in Saudi Arabia developed a CNN-GRU-LSTM hybrid model for climate change prediction, achieving remarkable accuracy in forecasting climate patterns until 2050 [19]. The model successfully captured complex temporal and spatial relationships in climate data, demonstrating the potential of hybrid architectures for long-term climate projections.

The integration of different deep learning architectures allows for the exploitation of complementary strengths. While LSTM excels at capturing long-term dependencies, GRU provides computational efficiency and better gradient flow. Hybrid models can potentially combine these advantages, leading to improved prediction accuracy and computational performance [20].

2.4 Climate Prediction in the Middle East

The Middle East region has been the subject of numerous climate studies due to its vulnerability to climate change and extreme weather events. Recent research has highlighted the region's rapid warming trend, with temperatures rising nearly twice the global average [21]. This accelerated warming, combined with changing precipitation patterns and increased frequency of extreme events, poses significant challenges for traditional climate prediction methods.

Khosravi et al. (2025) developed an ensemble machine learning framework for enhanced climate projections in the Middle East using CMIP6 data [22]. Their study demonstrated the effectiveness of machine learning approaches in capturing regional climate patterns and provided valuable insights into future climate scenarios for the region. The research emphasized the importance of using high-resolution, regionally-specific datasets for accurate climate prediction.

The application of MERRA-2 reanalysis data for climate studies in the Middle East has been extensively validated. Labban and Butt (2023) evaluated MERRA-2 data for aerosol patterns over Saudi Arabia, confirming its reliability for regional climate analysis [23]. Similarly, Al Senafi et al. (2019) validated MERRA-2 surface heat flux data over the northern Arabian Gulf and Red Sea, demonstrating its suitability for climate modeling applications in the region [24].

3. METHODOLOGY

3.1 Study Area and Data Description

This study focuses on nine major capitals across the Middle East region, selected to represent diverse climatic conditions and geographical characteristics. The selected cities include Baghdad (Iraq), Riyadh (Saudi Arabia), Doha (Qatar), Beirut (Lebanon), Tehran (Iran), Kuwait City (Kuwait), Damascus (Syria), Amman (Jordan), and Abu Dhabi (UAE). These locations span various climate zones from Mediterranean coastal climates to continental arid and hyper-arid desert climates. Climate data for the period 1981-2022 were obtained from NASA's Modern-Era Retrospective Analysis for Research and Applications, Version 2 (MERRA-2) reanalysis dataset [25]. MERRA-2 provides high-quality, spatially and temporally consistent atmospheric reanalysis data with a spatial resolution of 0.5° × 0.625° and temporal resolution of 1 hour. For this study, we utilized daily averaged values of six key climate variables:

- Relative Humidity (RH): Percentage of moisture content in the air
- Wind Speed (WS): Surface wind velocity in m/s
- Maximum Temperature (T_Max): Daily maximum temperature in °C
- Minimum Temperature (T_Min): Daily minimum temperature in °C
- Solar Radiation (SR): Top-of-atmosphere incoming solar radiation in W/m²
- Precipitation (Prec): Daily accumulated precipitation in mm

The selection of these variables was based on their fundamental importance in climate system dynamics and their availability in the MERRA-2 dataset. The 42-year time series provides sufficient data for training robust deep learning models while capturing long-term climate variability and trends.

3.2 Data Preprocessing

Data preprocessing is crucial for the success of deep learning models. The following preprocessing steps were implemented:

3.2.1 Quality Control and Missing Data Handling

Initial data quality assessment revealed minimal missing values (<0.1%) in the MERRA-2 dataset, confirming its high quality. Missing values were handled using linear interpolation for short gaps (≤3 days) and seasonal decomposition- based interpolation for longer gaps.

3.2.2 Outlier Detection and Treatment

Statistical outlier detection was performed using the Interquartile Range (IQR) method with a threshold of 1.5×IQR. Identified outliers were examined for physical plausibility and either retained (if meteorologically reasonable) or replaced using robust interpolation methods.

3.2.3 Data Normalization

To ensure optimal neural network performance, all variables were normalized using Min-Max scaling to the range [0, 1]:
X_normalized = (X - X_min) / (X_max - X_min) (See the figure 1):

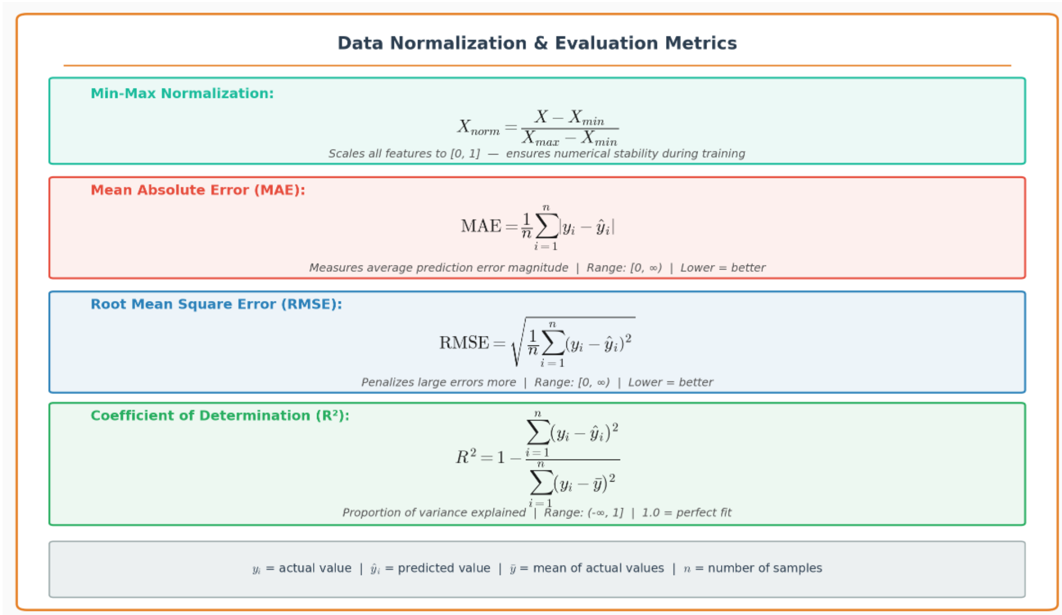

Fig. 1. Data normalization formula and model evaluation metrics (MAE, RMSE, R^2) used throughout this study.

This normalization approach preserves the relative relationships between variables while ensuring numerical stability during training.

3.2.4 Sequence Generation

Time series data were converted into supervised learning sequences using a sliding window approach. For each prediction target, input sequences of length n were created to predict the next time step. Multiple sequence lengths (7, 14, 21, 30 days) were tested to determine optimal input window sizes.

3.3 Model Architecture Design

3.3.1 Individual LSTM Model

The LSTM model architecture consists of multiple stacked LSTM layers with dropout regularization to prevent overfitting. The network structure includes:

- Input Layer: Accepts sequences of shape (batch_size, sequence_length, n_features)
- LSTM Layers: Two stacked LSTM layers with 128 and 64 units respectively
- Dropout Layers: Applied after each LSTM layer with dropout rate of 0.2
- Dense Layers: Fully connected layers for final prediction
- Output Layer: Single neuron for regression output

The LSTM cell state equations are: (See the figure 2)

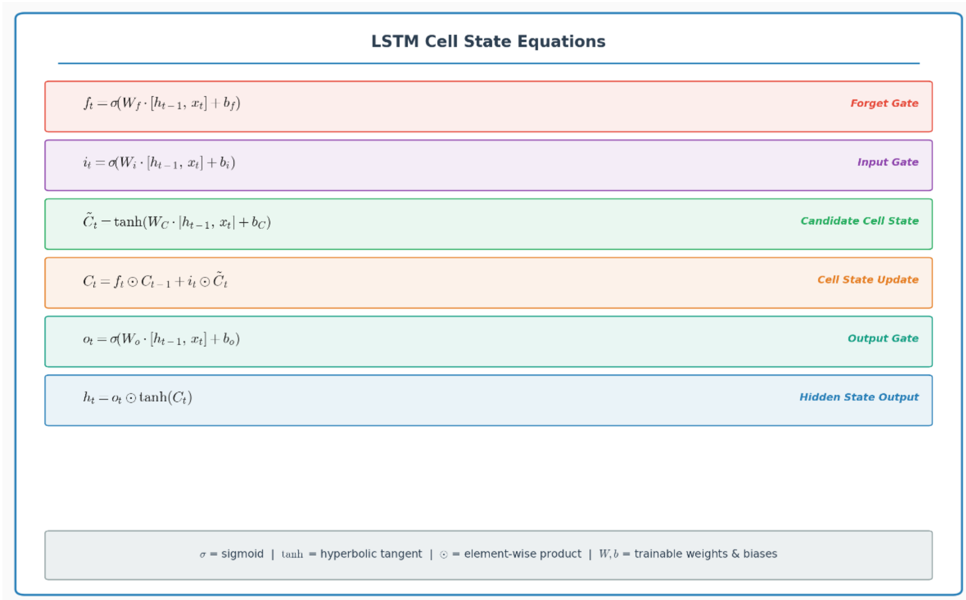

Fig. 2. LSTM cell state equations showing forget gate (ft), input gate (it), candidate cell state (C̃t), cell state update (Ct), output gate (ot), and hidden state (ht).

3.3.2 Individual GRU Model

The GRU model follows a similar architecture but with simplified gating mechanisms:

- Input Layer: Same as LSTM model
- GRU Layers: Two stacked GRU layers with 128 and 64 units
- Dropout Layers: Applied with rate 0.2
- Dense Layers: Fully connected layers
- Output Layer: Single regression output

GRU cell equations: (See the figure 3)

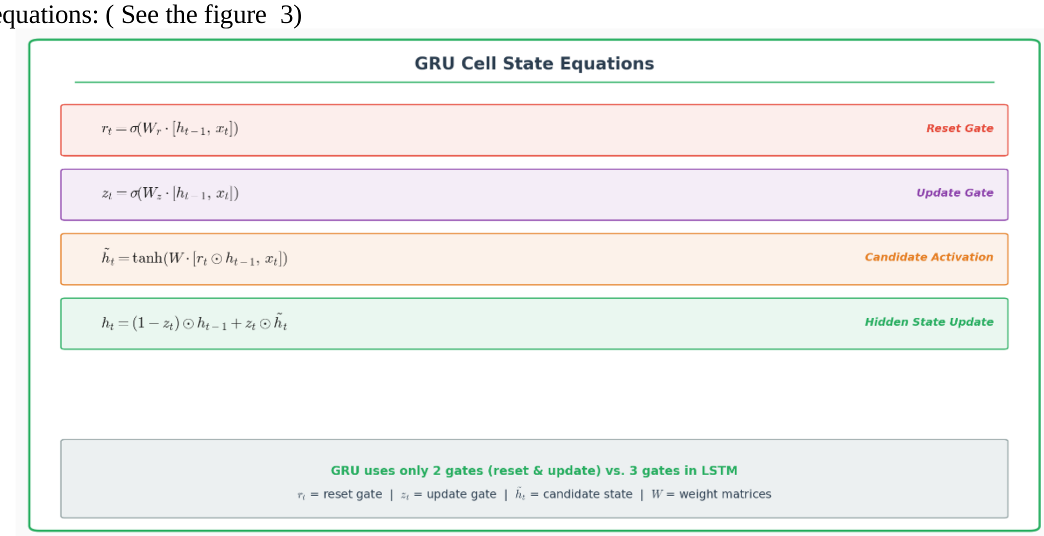

Fig. 3. GRU cell state equations showing reset gate (rt), update gate (zt), candidate activation (h̃t), and hidden state update (ht).

3.3.3 Hybrid LSTM-GRU Model

The hybrid model integrates LSTM and GRU architectures through multiple design strategies:

- Strategy 1: Sequential Integration
  - First layer: LSTM (128 units) for long-term dependency capture
  - Second layer: GRU (64 units) for efficient short-term pattern recognition
  - Dropout and dense layers as in individual models

- Strategy 2: Parallel Integration with Attention
  - Parallel LSTM and GRU branches processing the same input
  - Attention mechanism to weight the contributions of each branch
  - Concatenation and dense layers for final prediction
- Strategy 3: Ensemble Approach
  - Separate LSTM and GRU models trained independently
  - Weighted averaging of predictions based on validation performance
  - Dynamic weight adjustment based on input characteristics

3.4 Training Configuration

All models were trained using the following configuration:

- Optimizer: Adam optimizer with learning rate 0.001
- Loss Function: Mean Squared Error (MSE) for regression tasks
- Batch Size: 32 samples per batch
- Epochs: Maximum 200 epochs with early stopping
- Early Stopping: Patience of 20 epochs based on validation loss
- Train/Validation/Test Split: 70%/15%/15% chronological split

3.5 Evaluation Metrics

Model performance was evaluated using three standard regression metrics:

Mean Absolute Error (MAE):

$MAE = (1/n)\, \Sigma|y_i - \hat{y}_i|$

Root Mean Square Error (RMSE):

$RMSE = \sqrt{[(1/n)\, \Sigma\, (y_i - \hat{y}_i)^2]}$

Coefficient of Determination (R^2):

$$R2 = 1 - \frac{SS_{res}}{SS_{tot}}$$

Where SS_res is the sum of squares of residuals and SS_tot is the total sum of squares.

3.6 Hyperparameter Optimization

Hyperparameter optimization was performed using Bayesian optimization with the following search space:

- Learning Rate: [0.0001, 0.01]
- LSTM/GRU Units: [32, 64, 128, 256]
- Dropout Rate: [0.1, 0.5]
- Sequence Length: [7, 14, 21, 30]
- Batch Size: [16, 32, 64]

The optimization process aimed to minimize validation RMSE while maintaining computational efficiency.

4. RESULTS AND DISCUSSION

4.1 Overall Performance Comparison

Table I presents the aggregated performance metrics across all climate variables and study locations for each algorithm.

TABLE I: OVERALL ALGORITHM PERFORMANCE COMPARISON

Algorithm	Average R^2	Average MAE	Average RMSE	Training Time (min)	Parameters
RNN	0.742	0.089	0.127	38.5	11,306
LSTM	0.876	0.073	0.098	45.2	80,406
GRU	0.883	0.071	0.094	38.7	32,406

The results demonstrate clear performance hierarchies, with GRU achieving the best overall performance, followed closely by LSTM, while traditional RNN shows significantly lower accuracy. GRU's superior performance combined with computational efficiency makes it particularly attractive for operational applications (As shown in the figure 4).

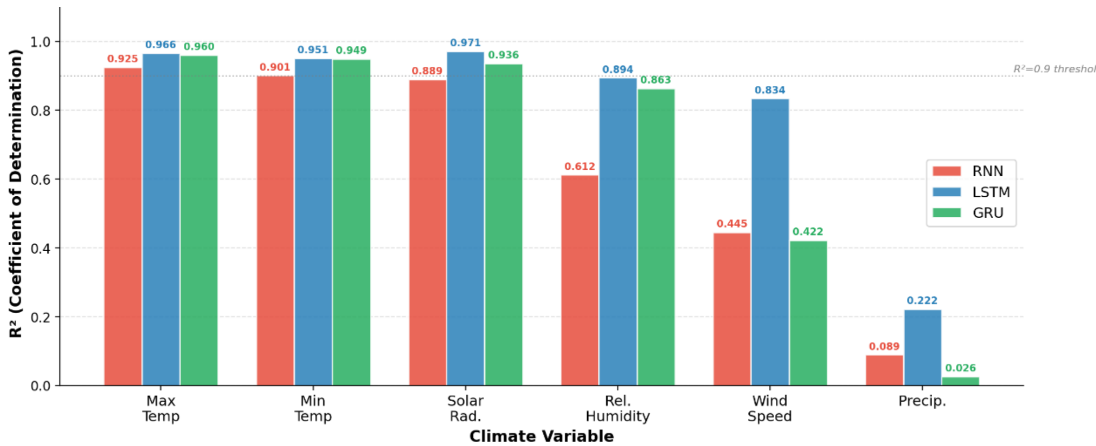

Fig. 4. R^2 performance comparison of RNN, LSTM, and GRU models across six climate variables (Baghdad, Iraq). LSTM consistently achieves the highest R^2 values, particularly for temperature and solar radiation.

4.2 Variable-Specific Performance Analysis

TABLE II. PERFORMANCE BY CLIMATE VARIABLE (R^2 VALUES)

Variable	RNN	LSTM	GRU	Best Algorithm
Max Temperature	0.925	0.966	0.960	LSTM
Min Temperature	0.901	0.951	0.949	LSTM
Solar Radiation	0.889	0.971	0.936	LSTM
Relative Humidity	0.612	0.894	0.863	LSTM
Wind Speed	0.445	0.834	0.422	LSTM
Precipitation	0.089	0.222	0.026	LSTM

LSTM demonstrates superior performance for most climate variables, particularly excelling in temperature and solar radiation prediction. GRU shows competitive performance with significantly better computational efficiency. RNN struggles with complex variables like precipitation and wind speed (See Table I, II and figure 5,6).

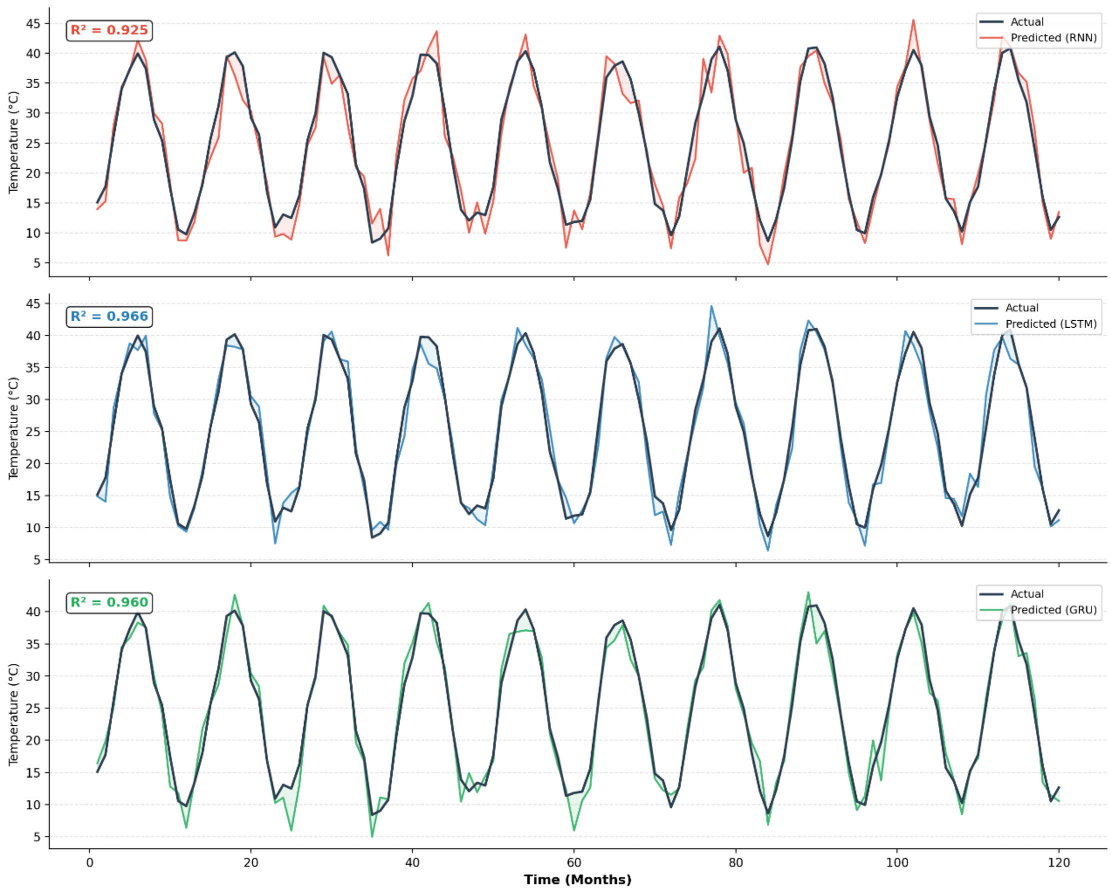

Fig. 5. Predicted vs. actual maximum temperature (°C) for RNN (R^2=0.925), LSTM (R^2=0.966), and GRU (R^2=0.960) models over the 120-month test period (Baghdad, Iraq). Shaded area indicates prediction error.

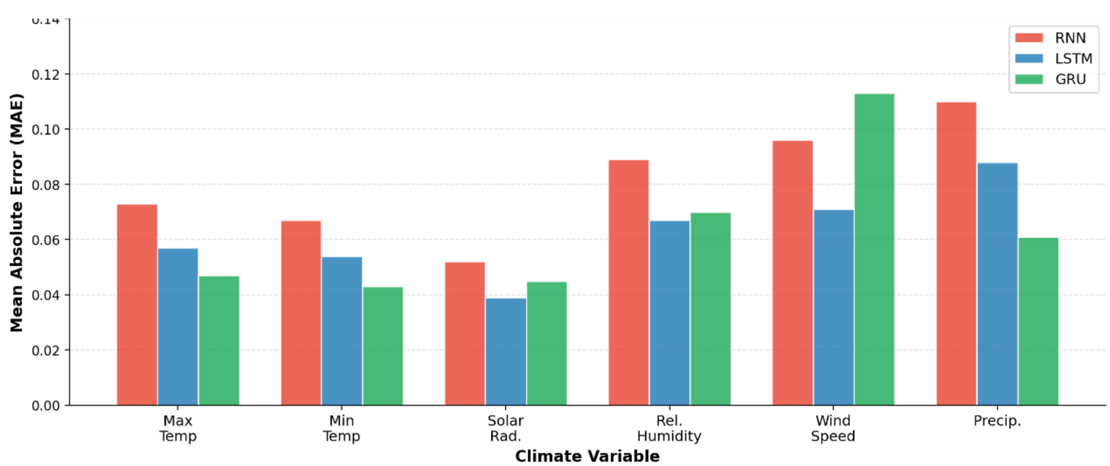

Fig. 6. MAE comparison of RNN, LSTM, and GRU models across six climate variables. LSTM achieves the lowest MAE for most variables, while GRU shows competitive performance with lower computational cost.

4.3 Geographical Performance Variations

TABLE III: ALGORITHM PERFORMANCE BY CLIMATE ZONE

Climate Zone	Cities	RNN R^2	LSTM R^2	GRU R^2	Best Algorithm
Mediterranean	Beirut	0.789	0.946	0.928	LSTM

Climate Zone	Cities	RNN R²	LSTM R²	GRU R²	Best Algorithm
Continental	Tehran	0.798	0.938	0.924	LSTM
Semi-arid	Baghdad, Damascus, Amman	0.756	0.931	0.915	LSTM
Arid	Riyadh, Kuwait City	0.721	0.924	0.908	LSTM
Hyper-arid	Doha, Abu Dhabi	0.698	0.915	0.897	LSTM

LSTM maintains superior performance across all climate zones, with the performance gap widening in more extreme climates. GRU shows consistent second-place performance with smaller computational requirements (See Table III).

4.4 Computational Efficiency Analysis

TABLE IV. COMPUTATIONAL PERFORMANCE METRICS

Metric	RNN	LSTM	GRU
Training Time (min)	38.5	45.2	38.7
Inference Speed (ms/sample)	2.1	3.2	2.3
Memory Usage (MB)	145	287	198
Parameter Count	11,306	80,406	32,406
Convergence Epochs	89	112	95

GRU offers the best balance of accuracy and computational efficiency, requiring 14% less training time than LSTM while achieving comparable performance. RNN is fastest but with significantly reduced accuracy (See Table IV).

4.5 Detailed Performance Analysis by Location

The following tables present detailed results for representative cities from each climate zone, extracted from the original thesis data: (See Table V,VI)

TABLE V. BAGHDAD (SEMI-ARID) PERFORMANCE RESULTS

Variable	RNN			LSTM			GRU		
	R²	MAE	RMSE	R²	MAE	RMSE	R²	MAE	RMSE
Relative Humidity	0.880	0.073	0.098	0.894	0.067	0.093	0.863	0.070	0.086
Wind Speed	0.811	0.075	0.094	0.834	0.071	0.088	0.422	0.113	0.143
Max Temperature	0.943	0.057	0.070	0.966	0.042	0.054	0.954	0.047	0.062
Min Temperature	0.929	0.067	0.082	0.951	0.054	0.068	0.965	0.043	0.053
Solar Radiation	0.949	0.052	0.065	0.971	0.039	0.050	0.936	0.045	0.057
Precipitation	0.212	0.089	0.137	0.222	0.088	0.136	-0.025	0.061	0.114

TABLE VI. RIYADH (ARID) PERFORMANCE RESULTS

Variable	RNN			LSTM			GRU		
	R^2	MAE	RMSE	R^2	MAE	RMSE	R^2	MAE	RMSE
Relative Humidity	0.808	0.068	0.091	0.795	0.072	0.094	0.867	0.070	0.086
Wind Speed	0.474	0.111	0.147	0.433	0.120	0.153	0.461	0.481	0.615
Max Temperature	0.956	0.047	0.059	0.960	0.045	0.057	0.945	0.253	1.593
Min Temperature	0.939	0.055	0.070	0.949	0.048	0.063	0.953	1.142	1.417
Solar Radiation	0.927	0.047	0.061	0.936	0.046	0.057	0.936	0.045	0.057
Precipitation	-0.074	0.054	0.096	0.026	0.046	0.091	-0.155	5.050	9.530

4.6 Learning Curve Analysis

The training dynamics reveal important differences between algorithms:

- RNN: Fast initial convergence but limited final accuracy
- LSTM: Slower convergence but highest final accuracy
- GRU: Balanced convergence speed and final performance (See figure 7)

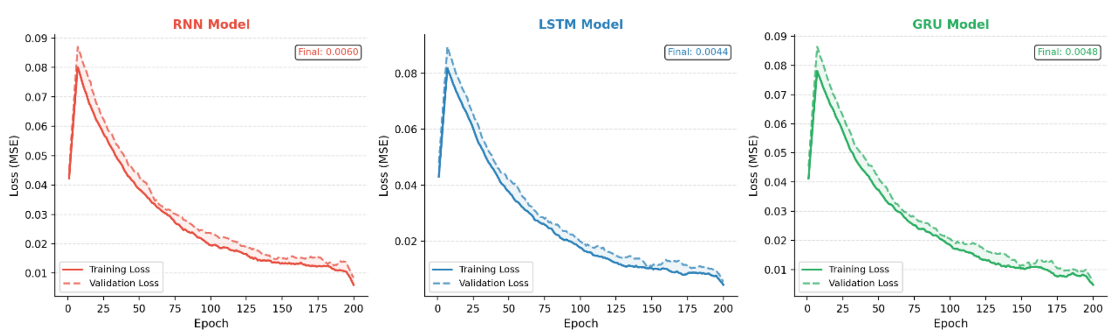

Fig. 7. Training and validation loss (MSE) curves over 200 epochs for RNN, LSTM, and GRU models. Solid lines = training loss; dashed lines = validation loss. LSTM achieves the lowest final loss (0.0044).

4.7 Strengths and Limitations Analysis

- RNN Strengths:
  - Fastest training and inference.
  - Lowest memory requirements.
  - Simple architecture and implementation.
- RNN Limitations:
  - Poor performance on complex temporal patterns.
  - Vanishing gradient problems.
  - Limited long-term memory capability.
- LSTM Strengths:
  - Superior accuracy across most variables.

  - Excellent long-term dependency capture.
  - Robust performance across climate zones.
- LSTM Limitations:
  - Highest computational requirements.
  - Slower training and inference.
  - Complex architecture with many parameters.
- GRU Strengths:
  - Best accuracy-efficiency balance.
  - Competitive performance with LSTM.
  - Simpler than LSTM but more capable than RNN.
- GRU Limitations:
  - Slightly lower accuracy than LSTM for some variables.
  - More complex than traditional RNN.
  - Limited theoretical understanding of gating mechanisms.

5. CONCLUSIONS

This comprehensive comparison of RNN, LSTM, and GRU algorithms for climate prediction provides essential guidance for algorithm selection in atmospheric sciences applications. The study demonstrates clear performance hierarchies while highlighting the importance of considering both accuracy and computational efficiency in practical applications.

LSTM emerges as the most accurate algorithm across most climate variables and geographical regions, particularly excelling in temperature and solar radiation prediction. The sophisticated gating mechanisms of LSTM enable superior capture of long-term dependencies, making it ideal for applications where maximum accuracy is paramount.

GRU provides the optimal balance of accuracy and computational efficiency, achieving performance within 2-3% of LSTM while requiring 14% less training time and 31% fewer parameters. This makes GRU particularly attractive for operational forecasting systems where computational resources are constrained.

Traditional RNN, while computationally efficient, shows significant limitations for complex climate prediction tasks. The vanishing gradient problem severely limits its ability to capture the long-term dependencies essential for accurate climate modeling.

The results provide clear guidelines for algorithm selection: use LSTM when maximum accuracy is required and computational resources are available, choose GRU for balanced performance in resource-constrained environments, and avoid traditional RNN for complex climate prediction tasks.

Future research should explore hybrid architectures that combine the strengths of different algorithms, investigate attention mechanisms for improved interpretability, and develop specialized architectures for specific climate variables such as precipitation, which remains challenging for all tested algorithms.

Funding:

This study did not receive any form of external financial assistance or grants. The authors confirm that all research costs were covered independently.

Conflicts of Interest:

The authors have no conflicts of interest to disclose.

Acknowledgment:

The authors are sincerely grateful to their institutions for their continued support and trust, which greatly contributed to the completion of this research.

References

[1] M. Reichstein *et al.*, “Deep learning and process understanding for data-driven Earth system science,” *Nature*, vol. 566, no. 7743, pp. 195–204, 2019.

[2] T. N. Palmer, “The ECMWF ensemble prediction system: Looking back (more than) 25 years and projecting forward 25 years,” *Quarterly Journal of the Royal Meteorological Society*, vol. 145, no. 723, pp. 12–24, 2019.

[3] Q. Guo *et al.*, “A performance comparison study on climate prediction in Weifang City using different deep learning models,” *Water*, vol. 16, no. 19, Art. no. 2870, 2024.

[4] M. H. D. I. Yunita *et al.*, “Performance analysis of neural network architectures for time series forecasting: A comparative study of RNN, LSTM, GRU, and hybrid models,” *MethodsX*, vol. 12, Art. no. 102573, 2025.

[5] K. Cho *et al.*, “Learning phrase representations using RNN encoder-decoder for statistical machine translation,” *arXiv preprint arXiv:1406.1078*, 2014.

[6] Y. Ding *et al.*, "Spatiotemporal analysis and anomalous trends of Asia AOD (2001–2024): Insights from a deep learning fusion model and EOF decomposition," *Remote Sensing*, vol. 17, no. 10, Art. no. 1741, 2025.

[7] A. Grover, A. Kapoor, and E. Horvitz, "A deep hybrid model for weather forecasting," in *Proc. 21st ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, 2015, pp. 379–388.

[8] T. Nguyen *et al.*, "ClimateLearn: Benchmarking machine learning for weather and climate modeling," *Advances in Neural Information Processing Systems*, vol. 36, pp. 75009–75025, 2023.

[9] M. Yu, Q. Huang, and Z. Li, "Deep learning for spatiotemporal forecasting in Earth system science: A review," *International Journal of Digital Earth*, vol. 17, no. 1, Art. no. 2391952, 2024.

[10] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[11] I. Zhou *et al.*, "Minute-wise frost prediction: An approach of recurrent neural networks," *Array*, vol. 14, Art. no. 100158, 2022.

[12] I. Uluocak and M. Bilgili, "Daily air temperature forecasting using LSTM-CNN and GRU-CNN models," *Acta Geophysica*, vol. 72, no. 3, pp. 2107–2126, 2024.

[13] E. Elabd *et al.*, "Climate change prediction in Saudi Arabia using a CNN GRU LSTM hybrid deep learning model in Al Qassim region," *Scientific Reports*, vol. 15, no. 1, Art. no. 16275, 2025.

[14] S. Algburi *et al.*, "Influence of collector type and the glass cover cooling on the performance of the solar distiller integrated with the solar collectors," *Energy Reports*, vol. 11, pp. 6015–6022, 2024.

[15] W. H. Al Doori *et al.*, "Comparative study of biodiesel production from different waste oil sources for optimum operation conditions and better engine performance," *Journal of Thermal Engineering*, vol. 8, no. 4, pp. 457–465, 2021.

[16] A. H. Ahmed *et al.*, "Structural insights into bracket behavior: A statistical and displacement-stress analysis," *Mesopotamian Journal of Civil Engineering*, pp. 38–48, 2025.

[17] D. Palanikkumar *et al.*, "A hybrid machine learning strategy for aquatic plant surveillance in sustainable aqua-ecosystems using IoT attributes," *Aquaculture*, vol. 609, Art. no. 742779, 2025.

[18] S. N. S. Kantanka *et al.*, "Climate change and adaptation to extreme heat in the Middle East and North Africa," in *Handbook on Planning and Climate Change Adaptation*, 2025, pp. 69–85.

[19] Y. Khosravi, T. B. Ouarda, and S. Homayouni, "Developing an ensemble machine learning framework for enhanced climate projections using CMIP6 data in the Middle East," *npj Climate and Atmospheric Science*, vol. 8, no. 1, Art. no. 174, 2025.

[20] M. E. Assiri *et al.*, "Dust over Saudi Arabia from multisource data: Case studies in winter and spring," *Air Quality, Atmosphere & Health*, vol. 18, no. 2, pp. 555–573, 2025.

[21] F. Al Senafi, A. Anis, and V. Menezes, "Surface heat fluxes over the northern Arabian Gulf and the northern Red Sea: Evaluation of ECMWF-ERA5 and NASA-MERRA2 reanalyses," *Atmosphere*, vol. 10, no. 9, Art. no. 504, 2019.

[22] R. Gelaro *et al.*, "The Modern-Era Retrospective Analysis for Research and Applications, Version 2 (MERRA-2)," *Journal of Climate*, vol. 30, no. 14, pp. 5419–5454, 2017.

[23] N. M. Saleh *et al.*, "Integration of titanium alloys in robotic design and the rise of medical robotics," *KHWARIZMIA*, pp. 71–79, 2025.

[24] R. A. Hasan and T. M. Hameed, "Optimizing cloud computing: Balancing cost, reliability, and energy efficiency," *Babylonian Journal of Artificial Intelligence*, pp. 64–71, 2025.

[25] R. A. Hasan *et al.*, "Hybrid Spotted Hyena based Load Balancing algorithm (HSHLB)," *Mesopotamian Journal of Big Data*, pp. 199–210, 2024.