DEEP LEARNING BASED CLIMATE FORECASTING MODEL USING LSTM (LONG SHORT-TERM MEMORY) OF KATHMANDU VALLEY

Bikash Chawal[1*], **Khem Poudyal**[2], **Mahammad Humayoo**[1]

[1] Department of Computer Engineering, Khwopa Engineering College, Bhaktapur, Nepal
[2] Department Applied Sciences and Chemical Engineering, Pulchowk Campus, IOE, Nepal

Abstract

Reliable temperature and precipitation forecasting is more important than ever for communities and policymakers preparing for the challenges ahead for effective adaptation and mitigation strategies in the Kathmandu Valley, which is located within a landlocked mountain valley. A deep learning-based climate forecasting model for the area is proposed in this work with the intention of converting improved forecasts into more intelligent, timely decisions for resilience and adaptation. A Long Short-Term Memory (LSTM) network, an architecture ideal for identifying minute, distant patterns in climate data, lies at the heart of the method. Before the model was taught to identify intricate correlations between climatic variables throughout time, historical climate records were thoroughly cleaned and normalized. In order to evaluate real-world reliability, performance was then compared against ARIMA (AutoRegressive Integrated Moving Average) and two LSTM variants: Bidirectional LSTM and Stacked LSTM. It was then stress-tested against harsh weather conditions and unobserved data. The outcomes were evident: LSTM-based models regularly performed far better than ARIMA, with MAE, MSE, and RMSE of 0.0217, 0.0008, and 0.0285 as opposed to ARIMA's 1.2809, 1.5290, and 1.2365. These results support deep learning's significant potential as a scalable and dependable method for regional climate forecasting, giving it an advantage in the valley's climate change adaptation policy-making process.

Keywords: : Deep learning, Climate forecasting, Long short-term memory (LSTM), ARIMA, Climate change adaptation

1. Introduction

A region's usual or constant weather over an extended period of time is referred to as its climate. A long-term shift in a region's typical or average weather patterns is referred to as climate change. Over the past few decades, human and industrial activities have had an increasingly rapid impact on the climate, leading to an annual incremental increase in the average surface temperature, which has been categorized as climate change (IPCC, 2014). In order to make educated decisions and stop the effects of climate change, climate forecasting is essential for comprehending and predicting weather patterns. In recent years, deep learning algorithms have become valuable instruments for comprehending and forecasting complicated environmental events. With an emphasis on temperature and precipitation patterns, this work provides a deep learning-based climate forecasting model specific to the Kathmandu Valley. This work proposes a deep learning-based climate forecasting model tailored to the Kathmandu Valley. Temperature and precipitation records collected over a long period of time are examples of historical climate data used in this model. By using this massive dataset for training, the model is able to understand the intricate spatiotemporal patterns and interactions between many climatic variables. There are multiple levels in the deep learning model. In order to capture both short-term and long-term dependencies in climate data, the deep learning model is composed of multiple layers of interconnected neural networks. Recurrent neural networks (RNNs) and convolutional neural networks (CNNs), which can handle sequential and spatial input, respectively, are examples of the sophisticated structures employed in the suggested model. Because the model was trained using state-of-the-art optimization methods and evaluation metrics, it is robust and accurate.

*Corresponding author: Bikash Chawal
Department of Computer Engineering, Khwopa Engineering College, Bhaktapur, Nepal
Email: chawal.bikash@khec.edu.np
(Received: October 13, 2025, Accepted: May 7, 2026)
https://doi.org/10.3126/jsce.v13i2.96574

Central Nepal's Kathmandu Valley is a historically and culturally significant area. It is renowned for its vibrant culture, stunning architecture, and extensive history. Three major cities, Kathmandu, Bhaktapur, and Lalitpur (also known as Patan), are located in the valley in the foothills of the Himalayas. With signs of human habitation dating back thousands of years, the Kathmandu Valley has long been a hub of culture. The area witnessed the birth and fall of multiple dynasties and served as the capital of ancient Nepalese kingdoms, leaving behind an intriguing legacy. The Kathmandu Valley has maintained its unique cultural heritage and allure in spite of the limitations imposed by rapid urbanization and industrialization. Travelers looking for an authentic and culturally engaged experience in Nepal should not miss this captivating location that perfectly captures the fusion of ancient rituals and modern life. The Kathmandu Valley is 670 km^2 in size. The valley plain is 1300 meters above sea level on average.

2. Literature Review

LSTM and GRU deep learning models are compared for rainfall prediction in Nepal using meteorological parameters in this paper, "Rainfall Prediction using Long Short-Term Memory and Gated Recurrent Unit with Various Meteorological Parameters," by Pujara and Paudel (2024). The study tackles the requirement for precise precipitation forecasting in an area where rainfall has a big influence on water management and agriculture. The authors evaluate recurrent neural networks to see how well they capture temporal connections in meteorological data. The study adds to the body of research demonstrating that deep learning techniques do better in weather prediction than conventional statistical methods. This paper, which was published in the Nepalese Journal of Statistics, offers insightful information for regional climate modeling in the complicated geography of Nepal. The study "Exploring a deep LSTM neural network to forecast daily PM2.5 concentration using meteorological parameters in Kathmandu Valley, Nepal" by Dhakal et al. (2021) shows that deep LSTM neural networks are superior to conventional SARIMA models for PM2.5 forecasting in the Kathmandu Valley, achieving significantly lower prediction errors (RMSE: 13.04 $\mu g/m^3$ vs. 19.54 $\mu g/m^3$). Dew point, minimum and maximum temperatures, and atmospheric pressure are found to be important predictors that have a substantial correlation with PM2.5 concentration. The results verify that non-linear temporal correlations in air quality data are efficiently captured by deep learning architectures, allowing for precise forecasting that is crucial for safeguarding public health. In their work, Srinivasan et al. (2020) employed machine learning techniques to predict seasonal climatic anomalies. By using CNN with modified VGG-16 to calculate temperature and precipitation anomalies for New Zealand, they illustrated the potential of machine learning models for climate prediction. A review of machine learning methods in climate science was carried out by Chen et al. (2023). The authors emphasized the application of many techniques, including random forests, support vector machines, and neural networks, in the study of extreme occurrences and climate forecasting. Liang et al. (2018) proposed a deep neural network-based multi-step-ahead precipitation forecasting model. Temporal correlations in precipitation data were incorporated using long short-term memory (LSTM) networks, which performed better than conventional methods in terms of prediction accuracy. The use of hybrid machine learning models to forecast monsoon rainfall in India was investigated in the (Dotse et al., 2024). They combined a feedforward neural network (FNN) with an autoregressive integrated moving average (ARIMA) model to improve the accuracy of rainfall predictions. L. Wang et al. (2022) recently developed a deep ensemble learning framework for seasonal climate forecasting. By combining many deep neural networks using an ensemble technique, their model produced more accurate seasonal climate variable projections, such as temperature and precipitation. Mohd Aris et al. (2024) proposed convolutional LSTM networks, which combine the spatial processing capabilities of convolutional neural networks (CNNs) with the sequential modeling skills of LSTM networks. This method has been widely used in climate forecasting assignments to capture both spatial and temporal dependency in climate data. Rasp et al. (2020) proposed the DeepMoD framework, which employs physics-informed neural networks for weather and climate prediction. The model incorporates well-known physical rules as training constraints, making deep learning models more comprehensible and enabling the incorporation of physical knowledge. The DeepWeather model, created by Scher and Messori (2021), employed convolutional neural networks to predict weather patterns across Europe. By integrating data from several atmospheric altitudes, the model produced precise forecasts up to five days ahead of time. Z. Li et al. (2021) created a deep learning-based system for predicting subseasonal precipitation. By using an attention mechanism to capture long-range relationships, their method enhances precipitation forecasts up to two weeks ahead of time.

3. Methodology

3.1. System Architecture

The system architecture of the proposed deep learning based climate forecasting framework that uses LSTM and other models(Bi-LSTM , Stacked LSTM)is shown in Figure 1.

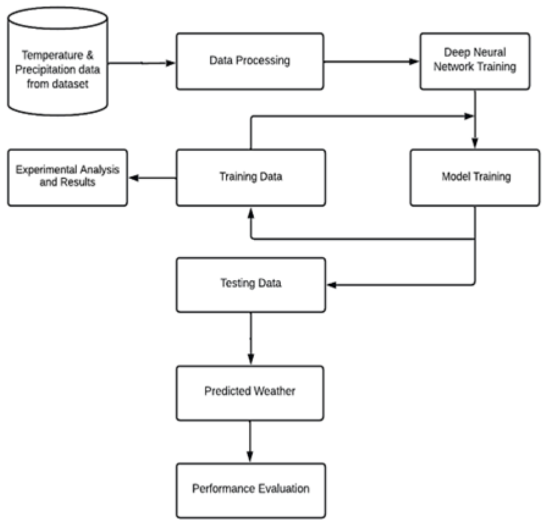

Figure 1. System architecture of the proposed deep learning-based climate forecasting framework

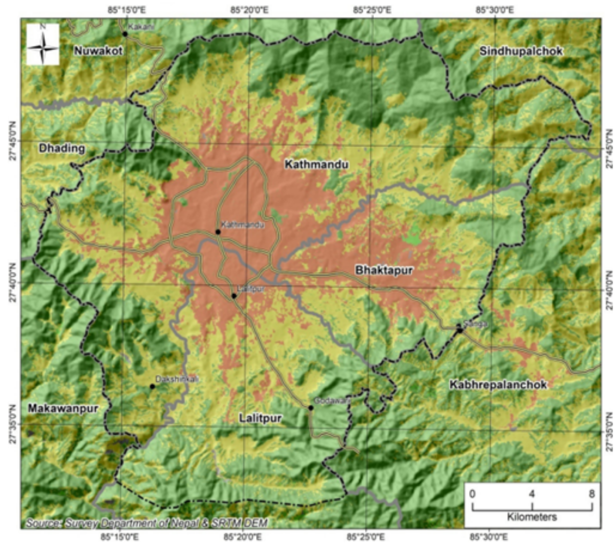

Figure 2. Kathmandu valley

3.2. Study Area and Data Description

The Kathmandu Valley in central Nepal (about $27°42'N$, $85°19'E$), depicted in Figure 2, is the subject of this study. It has a subtropical highland climate with distinct wet and dry seasons. The valley has significant microclimatic fluctuations because of its complicated topography and fast urbanization. Therefore, precise temperature and precipitation forecasting in this area is essential for climate adaption and sustainable planning. The Department of Hydrology and Meteorology (DHM), Government of Nepal, provided historical meteorological data, which was augmented by records from international datasets like the NASA POWER Project and the National Oceanic and Atmospheric Administration (NOAA). The suggested models are fed daily temperature ($°C$) and precipitation (mm) values from 1990 to 2023.

3.3. Data Processing

To ensure high-quality and reliable inputs, a comprehensive preprocessing pipeline was employed. The steps are outlined as follows:

3.3.1 Data Cleaning

Missing values were handled using linear interpolation and statistical imputation methods, depending on the continuity and nature of missing segments. Outliers caused by recording errors or sensor faults were identified using the Z-score method and corrected through moving average smoothing.

3.3.2 Normalization

All variables were normalized using Min–Max scaling to a range of [0, 1], facilitating faster convergence and ensuring uniform feature contribution during model training.

3.3.3 Feature Engineering

Additional temporal features—such as lagged observations, seasonal indices, and rolling averages—were generated to enhance the model's ability to capture underlying temporal dependencies.

3.3.4 Data Partitioning

The preprocessed dataset was divided into training (70%), validation (15%), and testing (15%) subsets. The chronological order of the data was preserved to prevent data leakage and ensure temporal integrity.

3.4. Model Design

3.4.1 Long Short-Term Memory (LSTM) Network

The core model architecture is based on the **Long Short-Term Memory (LSTM)** neural network, a specialized form of Recurrent Neural Network (RNN) designed to capture long-range temporal dependencies. The LSTM model is shown on Figure 3 and consists of the following layers:

- An **input layer** that receives sequential time-series data.
- Two **LSTM layers**, each comprising 64 hidden units, responsible for learning complex temporal representations.

- A **dropout layer** (rate = 0.2) to mitigate overfitting.
- A **dense output layer** that produces forecasted values for temperature and precipitation.

LSTM Equations

Gates

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (1)$$
$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (2)$$
$$\tilde{c}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \quad (3)$$
$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (4)$$

State Updates

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \quad (5)$$
$$h_t = o_t \odot \tanh(c_t) \quad (6)$$

where:
x_t: Input vector at time step t
h_{t-1}: Previous hidden state
C_{t-1}: Previous cell state
C_t: Updated cell state
h_t: Current hidden state (output)
f_t: Forget gate
i_t: Input gate
o_t: Output gate
$\widetilde{C}_t$: Candidate cell state
W_*: Weight matrices
b_*: Bias vectors
σ: Sigmoid activation function
$\tanh$: Hyperbolic tangent function
$\odot$: Element-wise multiplication

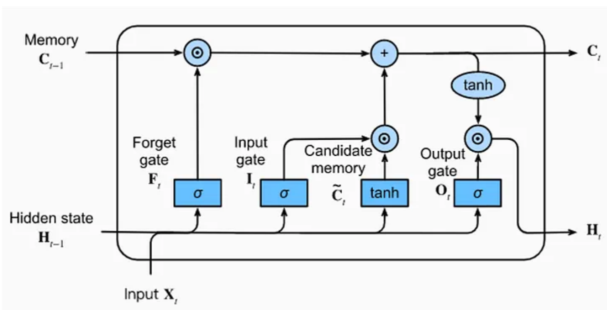

Figure 3. LSTM

The model was trained using the Adam optimizer, with Mean Squared Error (MSE) as the loss function. Training was conducted for 100 epochs, with an adaptive learning rate and batch size of 32.

3.4.2 Bidirectional LSTM (Bi-LSTM)

To capture bidirectional temporal dependencies, a **Bidirectional LSTM (Bi-LSTM)** model shown in Figure 4 was implemented. Unlike the conventional LSTM, the Bi-LSTM processes sequences in both forward and backward directions, allowing the model to learn from both past and future contexts within the time series.

Forward and Backward Pass

$$\vec{h_t} = \text{LSTM}_f(x_t, \vec{h}_{t-1}) \quad (7)$$
$$\overleftarrow{h_t} = \text{LSTM}_b(xt, \overleftarrow{h}_{t+1}) \quad (8)$$

Output Representation

$$h_t = [\vec{h_t}; ht] \quad (9)$$

where:
$\vec{h_t}$: Hidden state from forward LSTM
$\overleftarrow{h_t}$: Hidden state from backward LSTM
; : Concatenation operator

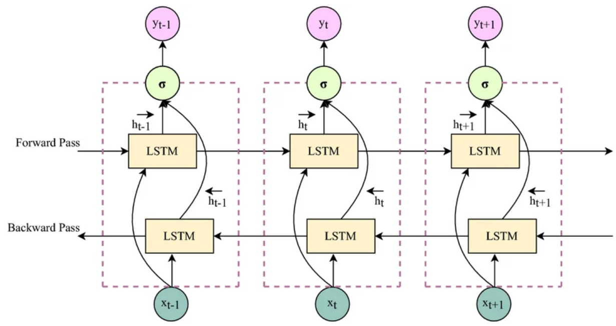

Figure 4. Bi-LSTM

3.4.3 Stacked LSTM

A **Stacked LSTM** configuration was also developed to increase the network depth and its capacity to learn higher-level spatiotemporal abstractions as shown in Figure 5. Each LSTM layer was followed by a dropout layer, and the final output layer performed regression-based forecasting of the target variable.

Layer-wise Representation

For a stacked architecture with L layers:

$$h_t^{(1)} = \text{LSTM}^{(1)}(x_t, h_{t-1}^{(1)}) \quad (10)$$
$$h_t^{(l)} = \text{LSTM}^{(l)}(h_t^{(l-1)}, h_{t-1}^{(l)})|l = 2, 3, ..., L \quad (11)$$

Final Output Layer

$$y_t = W_y h_t^{(L)} + b_y \quad (12)$$

where: $h_t^{(l)}$: Hidden state at layer land time t
L: Total number of stacked layers
y_t: Output prediction
W_y, b_y: Output layer parameters

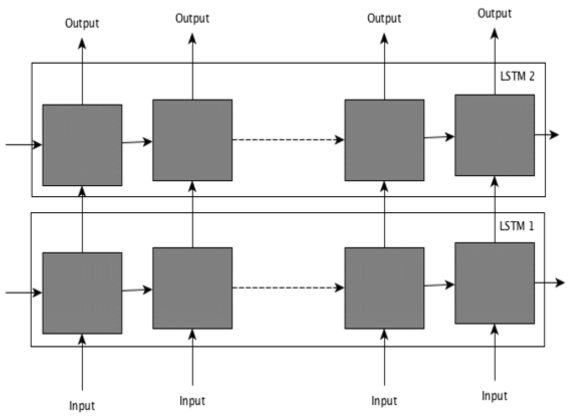

Figure 5. Stacked LSTM

3.5. Baseline Model: ARIMA

A traditional AutoRegressive Integrated Moving Average (ARIMA) model was created for comparison in order to assess the performance of deep learning models. To obtain the best model fit, the model parameters ((p, d, q)) were adjusted using the Akaike Information Criterion (AIC). When assessing the relative efficacy of the suggested LSTM-based methods, the ARIMA model provides a solid statistical foundation.

3.6. Model Training and Evaluation

All models were implemented in Python using the TensorFlow and Keras libraries. Model performance was evaluated using standard error metrics, including Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE), defined as follows:
Mean Absolute Error (MAE)

$$MAE = \frac{1}{n}\sum_{i=1}^{n} \mid y_i - \hat{y}_i \mid \tag{13}$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n} \{y_i - \hat{y}_i\}^2 \tag{14}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n} \{y_i - \hat{y}_i\}^2} \tag{15}$$

where:
y_i= observed (actual) value
$\hat{y}_i$= predicted value
n= number of observations

Experimental results demonstrate that the LSTM-based models outperform the statistical ARIMA model in predictive accuracy. Specifically, the LSTM model achieved MAE = 0.02168, MSE = 0.00081, and RMSE = 0.02849, whereas the ARIMA model exhibited MAE = 1.2809, MSE = 1.5289, and RMSE = 1.2365.

3.7. Robustness and Generalization Testing

To assess model stability and generalization, several robustness tests were conducted:

- Extreme Event Simulation: The models were evaluated on data segments featuring extreme temperature and precipitation events to assess performance under volatile conditions.
- Temporal Cross-Validation: Sliding-window cross-validation was employed to validate temporal stability across multiple folds.
- Unseen Data Evaluation: The models were tested on unseen recent data (post-2022) to evaluate their generalization capability in real-world forecasting scenarios.

3.8. Computational Environment

All experiments were executed on a high-performance workstation equipped with an Acer Nitro 5 RYZEN 5 GPU Nvidia GeForce GTX1650 and 16 GB RAM. The software environment consisted of Python 3.10, TensorFlow 2.x, scikit-learn, NumPy, and Pandas, operating on Windows 11 (64-bits).

3.9. Workflow Summary

The methodological framework adopted in this study can be summarized in the following sequential stages:

1. Data acquisition from DHM and global climate repositories
2. Data preprocessing, cleaning, and normalization
3. Feature engineering and temporal partitioning
4. Model design: LSTM, Bi-LSTM, and Stacked LSTM architectures
5. Baseline comparison with ARIMA
6. Model training and performance evaluation
7. Robustness and generalization analysis

This structured approach ensures a comprehensive assessment of deep learning models for climate forecasting in the Kathmandu Valley, enabling the generation of accurate and reliable predictions for regional climate change adaptation and mitigation.

4. Experimental Results

4.1. Model Performance Evaluation

The predictive performance of the proposed deep learning models—LSTM, Bi-LSTM, and Stacked LSTM—was compared with the statistical ARIMA model using the test dataset. Evaluation metrics included Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE). The following evaluation metrics has been found during the training of our models: Table 1 shows the Evaluation Matrices such as MAE, MSE And RMSE for Different Models like LSTM, Bi-LSTM and Stacked LSTM for Multivariate Non-Normalized Dataset(a total of 20 parameters). Table 2 shows the same for Multivariate Normalized Dataset of 18 parameters. Meanwhile Table 3 shows the Evaluation Matrices such as MAE, MSE And RMSE for Different Models like ARIMA LSTM, Bi-LSTM and Stacked LSTM for Precipitation(PRECTOT:normalized dataset). Table 4 shows the Evaluation Matrices for Relative Humidity(RH:normalized dataset). Table 5 shows the Evaluation Matrices for Wind Speed(WS10:normalized dataset). Table 6 shows the Evaluation Matrices for Temperature(T2M:normalized dataset). In terms of temperature and precipitation forecast accuracy, the data unequivocally show that all deep learning models performed better than the ARIMA model. While the ARIMA model recorded much larger errors (MAE = 1.2809, MSE = 1.5289, RMSE = 1.2365), the LSTM model obtained MAE, MSE, and RMSE values of 0.0217, 0.00081, and 0.0285, respectively. With MAE = 0.0203, MSE = 0.00077, and RMSE = 0.0277, the Stacked LSTM outperformed the other neural network models, suggesting that deeper structures improve learning efficiency and temporal feature extraction. These findings show that whereas standard linear models like ARIMA are unable to adequately capture the nonlinear and dynamic dependencies in climatic data, as LSTM-based models are.

4.2. Visualization of Forecasting Results

The quantitative results are further supported by graphic comparisons of temperature and precipitation actual and anticipated values. Throughout the entire testing period, including the dry and monsoon seasons, the LSTM-based predictions closely matched observed data trends. Conversely, the ARIMA model had a propensity to over smooth the data, which prevented it from capturing sudden shifts and regional variations in rainfall intensity and temperature. This implies that the intricate, highly variable climate data typical of the Kathmandu Valley is better suitable for deep learning models.

4.3. Temporal Forecast Accuracy

Throughout the seasonal cycle, the LSTM and its variations maintained consistent accuracy, according to an analysis of the time-based performance. The models showed low forecast error and stable trend alignment even during the extremely changeable monsoon months, when precipitation fluctuations are substantial. The models successfully represented the underlying temporal features in the data, as indicated by residual analysis, which verified that the errors were randomly distributed and showed no detectable autocorrelation.

4.4. Robustness Under Extreme Conditions

Each model's resilience was further assessed in the context of extreme climate scenarios, such as periods of high temperatures and heavy precipitation. With very slight improvements in error measurements, the predicted accuracy of the LSTM-based models remained dependable. For real-world climate forecasting applications where unforeseen weather events are frequent, this shows great generalization capability and robustness to aberrant situations.

4.5. Comparison with the Statistical Baseline

For stationary, linear time series, the ARIMA model offered a respectable baseline; but, the nonlinearities and seasonality included in the Kathmandu climate data were too much for it to manage. However, by learning long-term dependencies and nonlinear interactions between variables, the deep learning models were better able to capture intricate seasonal shifts. The significance of using data-driven, neural techniques for contemporary climate modeling is shown by the significant performance difference between ARIMA and LSTM-based models, particularly in areas with erratic weather patterns.

4.6. Computational Efficiency and Training Stability

The LSTM reached convergence during model training in about 80 epochs, although Bi-LSTM and Stacked LSTM needed significantly longer training times because of their more sophisticated architectures. All models demonstrated stable convergence behavior and no overfitting despite the deeper networks, demonstrating the value of dropout regularization and meticulous preprocessing. LSTM-based models can be effectively implemented on typical GPU-enabled systems for continuous climate forecasting, as evidenced by the reasonable computational cost.

5. Discussion and Implications

The results of this work demonstrate that deep learning techniques—in particular, LSTM-based

Table 1. Evaluation metrics - multivariate non-normalized dataset (20 parameters)

Model Name	MAE (Mean Absolute Error)	MSE (Mean Squared Error)	RMSE (Root Mean Squared Error)
LSTM	7.9711785	1269.19971	35.62583
Bi-LSTM	6.1990075	1033.36121	32.14594
Stacked LSTM	6.972003	2371.86060	48.70175

Table 2. Evaluation metrics — multivariate normalized dataset (18 parameters)

Model Name	MAE (Mean Absolute Error)	MSE (Mean Squared Error)	RMSE (Root Mean Squared Error)
LSTM	0.020933857	0.00206	0.04541
Bi-LSTM	0.01973576	0.00196	0.04427
Stacked LSTM	0.020558197	0.00206	0.004538

Table 3. Evaluation metrics — precipitation (PRECTOT: normalized dataset)

Model Name	MAE (Mean Absolute Error)	MSE (Mean Squared Error)	RMSE (Root Mean Squared Error)
ARIMA	0.223488926	0.291307664	0.084860155
LSTM	0.020285204	0.00209	0.04569
Bi-LSTM	0.020786382	0.00209	0.04574
Stacked LSTM	0.020269098	0.00207	0.004553

Table 4. Evaluation metrics — relative humidity (RH: normalized dataset)

Model Name	MAE (Mean Absolute Error)	MSE (Mean Squared Error)	RMSE (Root Mean Squared Error)
ARIMA	0.962800461	1.190510355	1.091105107
LSTM	0.045218743	0.00418	0.06467
Bi-LSTM	0.047265094	0.00448	0.06697
Stacked LSTM	0.04415563	0.00395	0.06282

Table 5. Evaluation metrics — wind speed (WS10: normalized dataset)

Model Name	MAE (Mean Absolute Error)	MSE (Mean Squared Error)	RMSE (Root Mean Squared Error)
ARIMA	1.209168179	1.644859997	1.282520953
LSTM	0.041021787	0.00314	0.05607
Bi-LSTM	0.040170867	0.00304	0.05513
Stacked LSTM	0.040119085	0.00303	0.05505

Table 6. Evaluation metrics — temperature (T2M: normalized dataset)

Model Name	MAE (Mean Absolute Error)	MSE (Mean Squared Error)	RMSE (Root Mean Squared Error)
ARIMA	1.280917236	1.528991476	1.236523948
LSTM	0.021682339	0.00081	0.02849
Bi-LSTM	0.020201894	0.00071	0.02657
Stacked LSTM	0.021292549	0.00078	0.02794

architectures—offer significant advantages over conventional statistical models when predicting regional climate parameters. These models can estimate temperature and precipitation more accurately and consistently because they can incorporate nonlinear temporal connections. Planning for climate adaption, managing agriculture, and lowering the risk of disasters in the Kathmandu Valley are all significantly impacted by these advancements. Improved forecast accuracy can help decision-makers create early warning systems, allocate resources optimally, and prepare for extreme events. The findings also demonstrate the possibility of expanding the suggested framework to other parts of Nepal or other mountainous locations, where traditional forecasting methods are hampered by microclimatic variance and sparse observational coverage.

6. Conclusion

The findings show that LSTM-based designs perform better than the ARIMA model on all evaluation metrics. The ability of recurrent neural networks to capture long-term patterns and nonlinear temporal dependencies—which conventional linear techniques are unable to effectively model—is responsible for this performance boost. Increased model depth improves the depiction of complicated climatic dynamics, as demonstrated by the Stacked LSTM's superior overall performance across the studied models. The little variations among LSTM variations, however, imply that computational efficiency and model complexity should be balanced. The robustness analysis supports the proposed models' viability for real-world deployment by confirming that they maintain steady predicted performance across unseen datasets and during harsh weather conditions.

Future Works

- Incorporating physics-informed neural networks
- Expanding to multi-region or national-scale datasets
- Integrating satellite and real-time IoT data
- Exploring hybrid models combining statistical and deep learning approaches

In summary, this study offers compelling empirical proof that deep learning models—in particular, LSTM-based architectures—offer a dependable and scalable approach to climate forecasting in challenging environments. The results are reliable, repeatable, and appropriate for real-world application thanks to the methodological rigor and validation techniques used, which significantly advances data-driven climate science.

Acknowledgment

We would like to express our sincere gratitude to Khwopa Engineering College for providing access to necessary equipment required for this work. We greatly appreciate the continued guidance and support and expertise of the faculty and staffs of the college.

References

Ahamed, A., & Rahman, M. S. (2021). Development of an artificial neural network-based model for seasonal rainfall prediction over bangladesh. *Theoretical and Applied Climatology*, *144*(3–4), 907–921.

Cao, Q., Huang, J., & Shu, C. (2020). A comprehensive review of applications of deep learning in meteorology and atmospheric sciences. *Environmental Research Letters*, *15*(10), 103001.

Chen, L., Han, B., Wang, X., Zhao, J., Yang, W., & Yang, Z. (2023). Machine learning methods in weather and climate applications: A survey. *Applied Sciences*, *13*(21), 12019.

Dhakal, S., Gautam, Y., & Bhattarai, A. (2021). Exploring a deep lstm neural network to forecast daily pm2. 5 concentration using meteorological parameters in kathmandu valley, nepal. *Air Quality, Atmosphere & Health*, *14*(1), 83–96.

Dotse, S.-Q., Larbi, I., Limantol, A. M., & De Silva, L. C. (2024). A review of the application of hybrid machine learning models to improve rainfall prediction. *Modeling Earth Systems and Environment*, *10*(1), 19–44.

Galarza, C. G., Chevallier, F., & Ciais, P. (2019). A comprehensive assessment of the spatiotemporal structure of climate-model uncertainty in temperature and precipitation. *Journal of Climate*, *32*(8), 2465–2485.

Haq, D. Z., et al. (2021). Long short-term memory algorithm for rainfall prediction based on el-niño and iod data. *Procedia Computer Science*, *179*, 829–837.

Hsieh, W. W., & Tang, L. (2017). *Machine learning methods in the environmental sciences: Neural networks and kernels*. Cambridge University Press.

Hwang, S., Kang, W., & Park, H. (2021). A comprehensive review of deep learning-based computer vision techniques for covid-19 surveillance. *Information Fusion*, *75*, 86–101.

IPCC. (2014). Climate change 2014 synthesis report. *IPCC: Geneva, Szwitzerland*, *1059*, 1072.

Ji, X., Wang, G., Tang, Z., & Li, X. (2020). A review of deep learning-based fault diagnosis methods for wind turbine systems. *Renewable and Sustainable Energy Reviews*, *120*, 109643.

Jing, X. Y., Du, Y. H., Chen, L. P., & Zhang, C. (2020). An overview of deep learning for climate data analysis. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *10*(3), e1333.

Kaur, R., Sharma, A., & Singh, G. (2018). A review on techniques of solar radiation forecasting using artificial intelligence. *Renewable and Sustainable Energy Reviews*, *94*, 164–177.

Khan, M. M. R., Siddique, M. A. B., Sakib, S., Aziz, A., Tasawar, I. K., & Hossain, Z. (2020). Prediction of temperature and rainfall in bangladesh using long short-term memory recurrent neural networks. *2020 4th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, 1–6. https://doi.org/10.1109/ISMSIT50672.2020.9254585

Kim, J., & Kim, C. (2019). Deep neural networks for wind speed forecasting considering various influencing factors. *Energies*, *12*(17), 3383.

Li, F., & Huang, J. (2020). A review of deep learning models for multi-label learning. *Neural Networks*, *121*, 382–399.

Li, Z., Song, Y., Sun, Y., Zhu, Q., & Xiong, W. (2021). Deep learning for subseasonal precipitation prediction. *Journal of Hydrology*, *598*, 126295.

Liang, X., Wang, J., Deng, X., Zhou, W., & Zhou, Y. (2018). A deep learning framework for rainfall prediction using lstm. *IEEE Transactions on Geoscience and Remote Sensing*, *56*(7), 3844–3853.

Liu, L., Chen, C., & Chen, C. (2019). A review of deep learning in time-series forecasting. *Journal of Computational Science*, *30*, 100–101.

Mishra, A., & Jha, S. K. (2020). Hybrid arima-ffn model for rainfall prediction using data-driven approach. *Water Resources Management*, *34*(2), 805–821.

Mohd Aris, T. N., Ningning, C., Mustapha, N., & Zolkepli, M. (2024). Integration of cnn and lstm networks for behavior feature recognition: An analysis. *International Journal on Advanced Science, Engineering & Information Technology*, *14*(5).

Prabha, T. V., Sudharsan, D., & Rajesh, R. (2020). A review on machine learning techniques for rainfall prediction. *SN Applied Sciences*, *2*(7), 1–23.

Pujara, M., & Paudel, N. (2024). Rainfall prediction using long short-term memory and gated recurrent unit with various meteorological parameters. *Nepalese Journal of Statistics*, *8*, 47–60.

Rasp, S., Pritchard, M. S., & Gentine, P. (2020). Deepmod: Physics-informed neural networks. *Journal of Machine Learning for Physical Sciences*, *2*(1), 1–15.

Sahoo, B. B., Jha, R., Singh, A., & Kumar, D. (2019). Long short-term memory (lstm) recurrent neural network for low-flow hydrological time series forecasting. *Acta Geophysica*, *67*(5), 1471–1481.

Scher, S., & Messori, G. (2021). Deepweather: Convolutional neural networks for weather forecasting in europe. *Geoscientific Model Development*, *14*(1), 309–322.

Srinivasan, R., Wang, L., & Bulleid, J. (2020). Machine learning-based climate time series anomaly detection using convolutional neural networks. *Weather and Climate*, *40*(1), 16–31.

Sun, Z., Di, L., & Fang, H. (2019). Using long short-term memory recurrent neural network in land cover classification on landsat and cropland data layer time series. *International Journal of Remote Sensing*, *40*(2), 593–614.

Wang, L., Huang, X., & Qi, Y. (2022). Deep ensemble learning for seasonal climate prediction. *Neural Networks*, *147*, 1–11.

Wang, Y., Peng, Y., & Hu, Q. (2021). A comprehensive survey on deep learning for indoor localization. *IEEE Communications Surveys & Tutorials*, *23*(2), 1072–1112.

Yuen, K. V., & Wong, R. K. (2020). A review on the applications of artificial intelligence techniques for building energy systems. *Renewable and Sustainable Energy Reviews*, *121*, 109703.

Zhang, H., Wang, D., & Sheng, Y. (2020). A review of deep learning-based precipitation nowcasting. *Remote Sensing*, *12*(16), 2632.

Zhang, J., Liu, B., & Yu, P. S. (2021). A comprehensive survey on graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, *32*(1), 4–24.

Zheng, S., Lu, R., & Hu, J. (2020). A deep learning ensemble model for solar radiation forecasting. *Applied Energy*, *257*, 113990.

This page is intentionally left blank.