Attentional Bi-LSTM for Multivariate Time Series Forecasting on Edge Devices: A Case Study on NanoPi Neo Plus2

Navid Hajizadeh
Department of Computer Engineering
Ferdowsi University of Mashhad
navid.hajizadeh@um.ac.ir

Saeed Yazdani
Department of Computer Engineering
Ferdowsi University of Mashhad
sa.yazdani@um.ac.ir

Sara Ershadi-Nasab
Department of Computer Engineering
Ferdowsi University of Mashhad
ershadinasab@um.ac.ir
corresponding author

***Abstract*—This study presents a resource-aware deep learning pipeline for next-step multivariate time series forecasting, featuring an attentional bidirectional long short-term memory architecture. The proposed model is designed to capture both forward and backward temporal dependencies while dynamically focusing on the most salient time steps using a Bahdanau-style attention mechanism. We first evaluate the method on the widely used Jena Climate dataset, then extend the study to a larger real-world meteorological dataset from California. This combination provides both a standard benchmark and a more challenging real-world test case, where the model demonstrates its superior predictive accuracy compared to state-of-the-art models, including CNN-RNN, CNN-LSTM, and Stacked-LSTM baselines. To ensure practical applicability in real-world, resource-constrained environments, the entire model is optimized and deployed on a NanoPi Neo Plus2 board—an ARM-based 64-bit single-board computer with limited computational resources. Our implementation leverages lightweight inference techniques and efficient model quantization to enable on-device prediction without cloud connectivity. The resulting system achieves competitive forecasting performance with minimal latency and power consumption, showcasing the feasibility of edge-AI solutions for environmental monitoring and smart sensing applications. Both quantitative and qualitative analyses confirm the effectiveness and interpretability of the proposed approach.The source code for this project is publicly available at: https://github.com/NavidH95/attentional-bilstm-for-edge-forecasting/tree/main.**

***Index Terms*—Attention Mechanism, Attentional Bi-LSTM, Bidirectional LSTM, Edge AI, Edge Deployment, Embedded Deep Learning, Multivariate Time Series Forecasting, NanoPi Neo Plus2, Real-Time Environmental Monitoring, Resource-Constrained Inference**

I. Introduction

Multivariate time series forecasting is a critical task in various real-world domains, including environmental monitoring, energy systems, finance, and industrial automation [1]–[3]. Accurate forecasting enables timely decision-making, anomaly detection, and resource optimization [4], [5]. However, real-world time series data often exhibit complex temporal dependencies, nonlinearity, and high dimensionality, making effective forecasting particularly challenging [6], [7].

Recent advances in deep learning, especially recurrent neural networks (RNNs) [8] and their variants like long short-term memory (LSTM) [9] networks, have demonstrated strong performance in capturing temporal patterns [10]–[12]. Bidirectional LSTM (Bi-LSTM) [13] architectures further enhance sequence modeling by considering both past and future contexts [14]. Nevertheless, not all historical time steps contribute equally to a prediction task. Attention mechanisms have emerged as a powerful complement to RNNs, allowing models to dynamically focus on the most informative time points [15].

In this study, we propose an attentional bidirectional LSTM architecture for next-step multivariate time series forecasting. Our model integrates a Bi-LSTM backbone with a Bahdanau-style attention mechanism [16] to selectively weigh hidden states and extract salient temporal features. We validate the model on two datasets: (i) the widely used Jena Climate dataset [17], which serves as a standard benchmark, and (ii) a larger real-world meteorological dataset from California containing 35,088 hourly samples with 20+ environmental and astronomical variables. This dual evaluation highlights performance on both a controlled benchmark and a more complex, real-world setting, with comparisons against traditional and deep learning-based baselines [5].

Beyond algorithmic performance, practical deployment of deep learning models on resource-constrained edge devices remains a key barrier to real-world adoption. To bridge this gap, we implement and deploy our proposed model on the NanoPi Neo Plus2 [18], a compact ARM-based single-board computer with limited compute and memory resources. Our deployment demonstrates that high-accuracy time series forecasting can be achieved efficiently at the edge, without relying on cloud servers or high-end GPUs.

The main contributions of this work are as follows:

- We design a resource-efficient attentional Bi-LSTM model for accurate multivariate time series forecasting.
- We validate the model on both a standard benchmark dataset (Jena Climate) and a large-scale real-world weather dataset (California), demonstrating robustness across simple and complex forecasting tasks.
- We introduce an edge deployment framework and demonstrate successful inference on the NanoPi Neo Plus2 platform [18].

- We provide a comprehensive experimental evaluation and compare our model with several state-of-the-art methods on the Jena Climate dataset [17].
- We offer insights into model interpretability through attention weight visualization, highlighting which temporal features drive predictions.

This paper is organized as follows: Section II reviews the most relevant studies and highlights the key differences between existing approaches and our proposed method; Section III describes the proposed methodology; Section IV presents the experimental setup, evaluation results, and comparative analyses; Section V describes the hardware setup and real-time deployment aspects of our system; and Section VI concludes with future directions.

II. Related Work

Recent advances in time series forecasting have leveraged a wide range of machine learning and deep learning techniques. In this section, we categorize and review the most relevant works in the literature, including hybrid deep learning models, standard recurrent architectures, and traditional machine learning approaches.

Hybrid Deep Learning Models have gained popularity due to their ability to capture both spatial and temporal dependencies in complex multivariate time series data. Models such as CNN-RNN [1], CNN-LSTM [14], and Recurrence Plot combined with CNN [7] exploit convolutional layers for local feature extraction and recurrent layers for temporal modeling. These architectures have demonstrated improved forecasting accuracy in various environmental and energy-related datasets. Another study proposes a highly efficient hybrid weather prediction model based on deep learning, combining various neural networks to improve prediction accuracy, especially for dynamic and complex environmental datasets [5]. These advancements highlight the growing importance of hybrid models in environmental forecasting. Recent work on Bi-LSTM networks with attention mechanisms has also shown superior predictive performance for time series forecasting, such as in the study of environmental monitoring systems implemented on resource-constrained edge devices.

Standard Recurrent Architectures like the standard LSTM [10] and stacked-LSTM [12] have also been extensively studied for sequence modeling tasks. While effective in capturing long-term dependencies, these models often lack mechanisms for distinguishing the relative importance of different time steps, which limits their interpretability and adaptability in dynamic environments.

Traditional Machine Learning and Other Methods include classical algorithms such as the Levenberg-Marquardt neural network [6], [15], Wavelet-SVM [3], Stacked denoising autoencoders (SDAE) [19]. These methods often require extensive feature engineering and may struggle to model nonlinear temporal interactions, particularly in high-resolution or noisy datasets. Specifically, SVR has been found to outperform other machine learning methods in the prediction of long-term air temperatures in Australia and New Zealand, showing the importance of capturing climate change patterns with statistical methods [4]. Nevertheless, lightweight statistical approaches (e.g., LMS, ARIMA, SVR) may offer competitive performance for simple one-step forecasting tasks but struggle to maintain accuracy in multivariate, high-resolution datasets like ours. This motivates the use of attention-enhanced deep learning architectures that balance accuracy, interpretability, and deployment efficiency.

In contrast to these existing methods, our proposed model integrates a BidireLSTM architecture with an attention mechanism, enabling the model to capture temporal dependencies from both directions while dynamically focusing on the most informative time steps. Furthermore, we extend the existing body of work by deploying the model on a resource-constrained embedded platform (NanoPi Neo Plus2), demonstrating its practical applicability for real-time edge-AI forecasting tasks.

III. Methodology

This section details our proposed model for next-step multivariate time series forecasting. We first describe the data preparation and problem formulation, followed by a detailed explanation of our Attentional Bidirectional LSTM architecture.

A. Data Preparation and Problem Formulation

The model was developed and evaluated on two multivariate time series datasets: (i) the Jena Climate dataset ($D = 14$ features), which serves as a benchmark, and (ii) a large-scale California weather dataset containing 35,088 hourly records with $D = 22$ features (including temperature, humidity, windspeed, cloud cover, UV index, astronomical variables, etc.). To address disparate feature scales and ensure numerical stability, we normalized each feature vector $\mathbf{x}$ to a $[0, 1]$ range using Min-Max normalization. The scaling parameters were derived solely from the training set to prevent data leakage.

$$\mathbf{x}_{\text{scaled}} = \frac{\mathbf{x} - \mathbf{x}_{\min}}{\mathbf{x}_{\max} - \mathbf{x}_{\min}} \tag{1}$$

Subsequently, the forecasting task was framed as a supervised learning problem via a sliding window approach. Input sequences $\mathbf{X}_i$ of a fixed length $L = 128$ were generated to predict the subsequent time steps. In our experiments, we evaluate both single-step forecasting ($\mathbf{y}i = \mathbf{x}i + 1$) and multi-step forecasting horizons (e.g., 3, 6, and 24 hours ahead), to assess robustness in short-term and longer-term prediction scenarios. To augment the dataset, a stride of $S = 64$ was employed, creating a 50% overlap between consecutive sequences. This approach balances dataset size and computational efficiency, ensuring sufficient temporal coverage while reducing redundancy.

B. Proposed Model: Attentional Bi-LSTM Network

The overall architecture of our proposed model is illustrated in Figure 1. The model follows a sequence-to-vector design, beginning with a Bi-LSTM layer that processes the input sequence to extract rich contextual features from both forward

and backward temporal directions. The complete set of hidden states from this layer is then passed to a Bahdanau-style attention mechanism. This layer computes attention weights to produce a single, fixed-length context vector that selectively summarizes the most salient information from the input sequence. Finally, this context vector is fed into a dense feed-forward layer to generate the multivariate prediction for the next time step.

We propose a sequence-to-vector architecture designed to capture complex temporal patterns by identifying salient historical features. The core of our model is a Bi-LSTM network, which processes input sequences from both forward ($\overrightarrow{\mathbf{h}_t}$) and backward ($\overleftarrow{\mathbf{h}_t}$) directions. The resulting hidden states are concatenated, $\mathbf{h}_t = [\overrightarrow{\mathbf{h}_t}; \overleftarrow{\mathbf{h}_t}]$, to create a rich, dual-context representation. To enable the model to dynamically focus on the most relevant time steps, we integrated a Bahdanau-style attention mechanism. This layer computes a context vector ($\mathbf{c}$) as a weighted sum of the Bi-LSTM hidden states. The mechanism calculates alignment scores (e_t) which are then normalized via a softmax function to produce attention weights (α_t).

$$e_t = \mathbf{v}^T \tanh(\mathbf{W}_h \mathbf{h}_t + \mathbf{W}_s \mathbf{s}) \tag{2}$$

$$\alpha_t = \frac{\exp(e_t)}{\sum_{k=1}^{L} \exp(e_k)} \tag{3}$$

The context vector, $\mathbf{c} = \sum_{t=1}^{L} \alpha_t \mathbf{h}_t$, serves as a distilled summary that is passed to the final prediction layer. The model is trained end-to-end by minimizing the mean squared error (MSE) loss using the Adam optimizer.

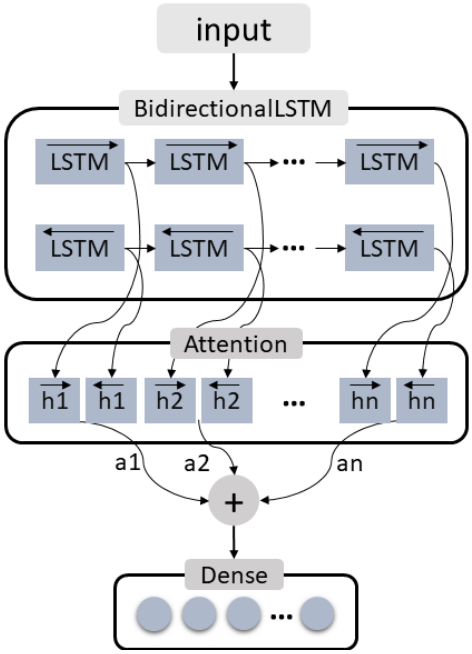

Fig. 1: Architecture of the proposed model.

TABLE I: Model Hyperparameters used for training the attentional Bi-LSTM model. Most parameters were kept consistent across datasets, except the number of epochs.

Hyperparameter	Jena Climate	California
Sequence Length (L)	128	128
Stride (S)	64	64
Hidden Dimension (per Bi-LSTM layer)	64	64
Bi-LSTM Layer Count	2	2
Dropout Probability	0.2	0.2
Optimizer	Adam	Adam
Learning Rate	0.001	0.001
Batch Size	64	64
Number of Epochs	20	50

IV. Experiments and Evaluation

This section details the experimental framework designed to evaluate the performance of our proposed model. We first outline the experimental setup and evaluation protocol, followed by quantitative and qualitative results.

A. Experimental Setup

All experiments were conducted on the Jena Climate dataset. The model was implemented using PyTorch and trained on a system equipped with an NVIDIA Tesla T4 GPU. The key hyperparameters for our proposed model are listed in Table I.

B. Evaluation Protocol

To rigorously assess the model's generalization performance while respecting temporal dependencies, a specialized evaluation protocol was imperative. Standard k-fold cross-validation, which involves random data shuffling, is unsuitable for time-series forecasting as it would cause data leakage. Consequently, we adopted a **walk-forward validation** methodology. The dataset was first chronologically partitioned, with the initial 85% of data allocated for training and validation, and the final 15% reserved as a completely unseen hold-out test set. This protocol was applied independently to both datasets: the Jena Climate dataset for benchmark comparison, and the California weather dataset to evaluate performance on a more complex, real-world scenario. Within the 85% training partition, we implemented a 5-fold 'TimeSeriesSplit' scheme. In this scheme, the training window progressively expands to include data from the previous validation block, ensuring the model is always tested on future data. Informed by this validation process, a final model was trained on the entire 85% partition. Its performance was then conclusively evaluated on the hold-out test set using **mean squared error (MSE)**, **root mean squared error (RMSE)**, **mean absolute error (MAE)**, and **coefficient of determination (R^2)** as our primary metrics.

C. Quantitative Results

The learning progression of the final model is depicted in Figure 3. The training loss (blue line) shows a consistent and smooth decrease, indicating that the model is effectively learning from the training data. Critically, the validation loss (orange line), evaluated on the unseen test set, also converges and remains stable at a low value without significant

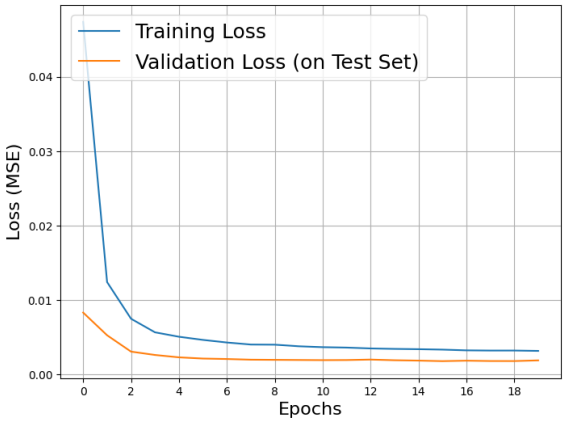

Fig. 2: Training and validation loss curves for the Jena benchmark

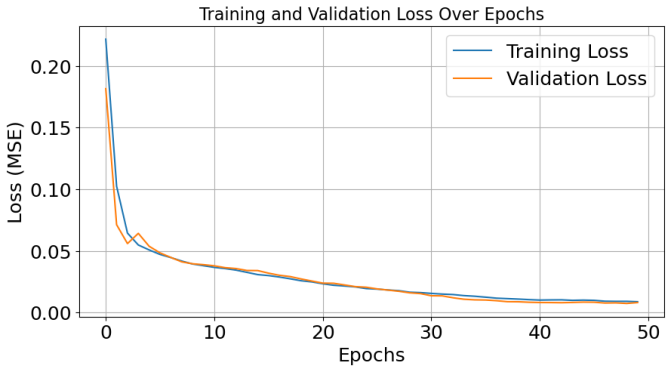

Fig. 3: Training and validation loss curves for the California real-world datasets

divergence from the training loss. This convergence pattern demonstrates a stable training process and suggests that the model has generalized well without succumbing to significant overfitting. Similar learning and convergence patterns were observed for the California dataset, demonstrating that the attentional Bi-LSTM can effectively capture complex temporal dependencies in large, multivariate real-world time series.

To contextualize the performance of our proposed attentional Bi-LSTM model, we conduct a comprehensive comparative analysis against a diverse range of models from the existing literature. The selection of these baseline models spans several categories, including traditional machine learning methods, standalone recurrent architectures, and state-of-the-art hybrid deep learning models. The performance metrics for these models are cited directly from their respective publications, as reported on the same Jena Climate dataset. In addition, we evaluate our proposed attentional Bi-LSTM on the California dataset to provide a real-world assessment. On the California dataset, the model maintains strong predictive performance with an R^2 of 0.79, despite the dataset's higher dimensionality and more irregular temporal patterns. This confirms that the attentional Bi-LSTM effectively captures

TABLE II: Comparison of the proposed attentional Bi-LSTM model with other models in the literature

Model	MSE	RMSE	MAE	R^2	Reported by
Attentional Bi-LSTM	**0.001809**	**0.042531**	**0.013926**	**0.9284**	Proposed method (Jena dataset)
Attentional Bi-LSTM	**0.006407**	**0.080044**	**0.055011**	**0.7915**	Proposed method (California dataset)
CNN-RNN	0.035	0.189	0.126	0.987	A. Utku and U. Can [1]
Levenberg–Marquardt	1.96550	–	–	0.98881	Dombaycı and Gölcü [2]
Wavelet-SVM	0.0937	–	–	–	Liu et al. [3]
Levenberg–Marquardt	–	1.53	1.27	0.995	Kisi and Shiri [6]
SDAE	–	1.38	–	–	Hossain et al. [19]
SVR, MLP	–	–	0.7232	–	Salcedo–Sanz et al. [4]
Stacked-LSTM	1.5365	1.236	0.9056	0.9692	Li et al. [10]
LSTM	–	1.04	–	0.984	Li et al. [11]
CNN–LSTM	–	1.97	1.02	–	Hou et al. [14]
Recurrence Plot + CNN + Binarised	0.718	–	0.696	–	Fister et al. [7]

complex dependencies across multiple meteorological variables, highlighting its practical utility for real-world forecasting tasks.

Table II presents a direct comparison of our model's performance against these established benchmarks, demonstrating that our proposed attentional Bi-LSTM model achieves a significantly lower error across all primary metrics. The model obtains an **MSE of 0.001809**, an **RMSE of 0.042531**, an **MAE of 0.013926**, and $\boldsymbol{R^2}$ **of 0.9284**. These results represent a substantial improvement over the next-best performing model, the CNN-RNN [1], showcasing the superior predictive accuracy of our approach. The superior performance underscores the efficacy of combining a bidirectional context with a dynamic attention mechanism for this forecasting task.

D. Comparative Analysis

This section provides a deeper analysis of these comparative results. A closer examination reveals several key insights. When compared to other sophisticated deep learning architectures like CNN-RNN [1] and stacked-LSTM [12], our model's superiority is particularly evident. We attribute this significant performance gain to the synergistic effect of its core architectural components. First, the bidirectional nature of the LSTM layers allows the model to build a more comprehensive contextual understanding of the input sequence by processing information from both past and future directions. This creates a richer representation than what is available to unidirectional models like the standard LSTM [10]. Second, and more critically, the integrated attention mechanism empowers the model to dynamically weigh the importance of each time step within this rich context. Unlike architectures that may treat all historical data with uniform importance, our model learns to focus on the most salient temporal features for the prediction task, a behavior we explore qualitatively in the next section. This ability to intelligently filter and prioritize information is a distinct advantage and a primary driver of the model's reduced error rates.

In summary, the comparative analysis confirms the quantitative superiority of our proposed architecture and suggests that its unique combination of bidirectional context encoding and an attention-based focus mechanism is highly effective for high-resolution climate forecasting, as demonstrated across both the benchmark Jena dataset and the larger, real-world California dataset.

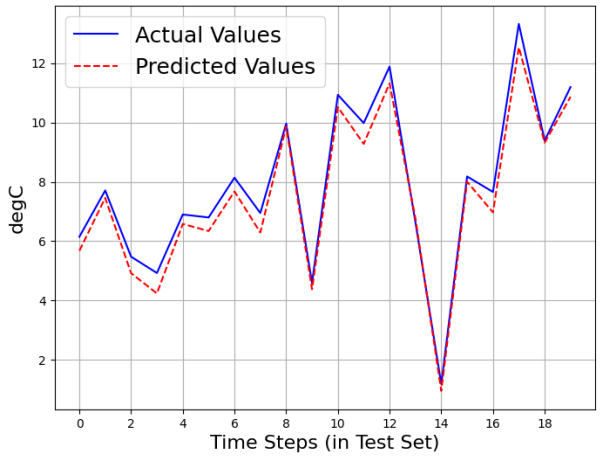

Fig. 4: Predicted vs. Actual values for the T(degC) feature in the Jena dataset.

E. Qualitative Analysis

Beyond quantitative metrics, a qualitative analysis was performed to visually assess the model's behavior and interpretability. First, to confirm the model's forecasting accuracy, we visualized its predictions against the ground-truth values for representative features on the test set. As depicted in Figure 4, the predicted values (red dashed line) demonstrate high fidelity, closely tracking the fluctuations and trends of the actual time series (blue solid line). Likewise, Figure 5 presents the predicted vs. actual values for the temperature feature in the California dataset. The model effectively captures the real-world fluctuations and trends, confirming its generalizability to larger, more complex datasets. This visual evidence corroborates the strong quantitative results and confirms the model's capability to capture the complex dynamics of the climate data.

Second, to understand the model's decision-making process, we visualized the attention weights (α_t) assigned to the input sequence for a representative prediction, as shown in Figure 6. The plot clearly reveals that the model has learned to allocate the vast majority of its attention to the most recent time steps. This behavior is highly logical, as it indicates the model has successfully identified the strong short-term auto-correlation present in the data. Rather than treating all historical data points equally, the attention mechanism dynamically focuses on the most salient temporal segment, which validates our architectural choice and highlights the model's interpretability. Figure 7 shows the attention weights for a representative California prediction. As with the Jena dataset, the model assigns higher attention to the most informative time steps and covariates, demonstrating consistent prioritization patterns across datasets and reinforcing the interpretability of the attentional Bi-LSTM architecture.

F. Ablation Study

To systematically investigate the contribution of each key component of our proposed architecture, we conducted a comprehensive ablation study. We evaluated the performance

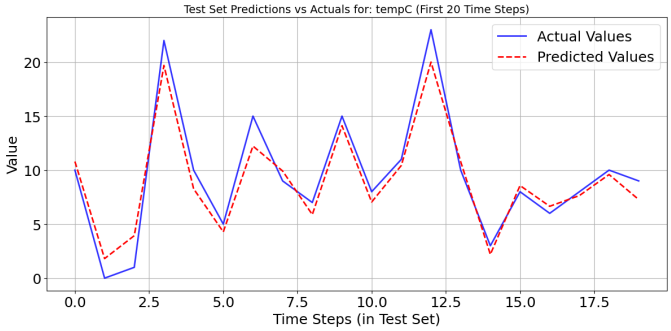

Fig. 5: Predicted vs. Actual values for the Temperature (degC) feature in the California weather dataset.

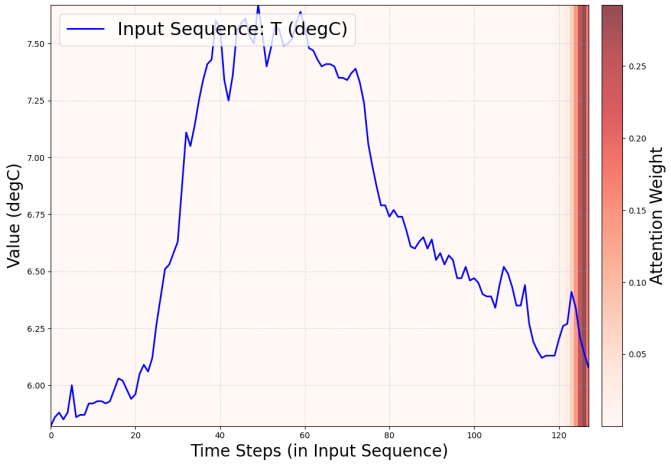

Fig. 6: Visualization of attention weights for a single prediction on the Jena dataset.

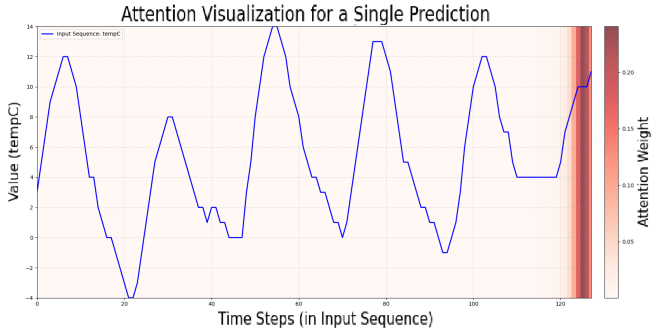

Fig. 7: Attention weights visualization for a single prediction on the California dataset. The model prioritizes the most informative time steps across multiple meteorological variables.

of our full attentional Bi-LSTM model against three ablated versions: (i) a standard Bi-LSTM model without the attention mechanism, (ii) a unidirectional attentional LSTM, and (iii) a vanilla (unidirectional) LSTM. This analysis allows us to isolate the impact of bidirectionality and the attention mechanism. The results are summarized in Table III. We additionally evaluated the ablated models on the California dataset. While

TABLE III: Performance comparison of the proposed attentional Bi-LSTM model with baseline LSTM variants on the Jena Climate dataset

Model	MSE	RMSE	MAE	R^2	Reported by
Attentional Bi-LSTM	**0.001913**	**0.043736**	**0.018867**	**0.9284**	Proposed method
Bi-LSTM	0.001952	0.044185	0.020738	0.9229	Our implementation
Vanilla LSTM	0.002128	0.046132	0.022769	0.9117	Our implementation
Attentional LSTM	0.004252	0.065204	0.039729	0.7827	Our implementation

absolute errors are higher due to more irregular temporal patterns, the relative improvements from bidirectionality and attention remain consistent, confirming that both architectural components generalize effectively to real-world, large-scale multivariate data.

The Critical Role of Bidirectionality: The most significant performance degradation occurs when bidirectionality is removed. The unidirectional attentional LSTM shows a dramatic increase in error (e.g., MSE increases by over 122% compared to our full model). This starkly demonstrates that encoding contextual information from both past and future time steps is fundamental to accurately capturing the complex temporal dynamics of the dataset. The richer representation provided by the Bi-LSTM serves as a powerful foundation for the model.

The Fine-Tuning Effect of Attention: The benefit of the attention mechanism is also clearly evident. Comparing our full model to the standard Bi-LSTM, we observe a consistent improvement across all error metrics. For instance, the MAE is reduced from 0.020738 to 0.018867. This indicates that once a rich bidirectional context is established, the attention layer provides an effective and valuable fine-tuning step, allowing the model to dynamically focus on the most salient features within that context for a more precise prediction.

In conclusion, the ablation study validates our architectural design choices. The findings suggest that while both components are beneficial, establishing a comprehensive bidirectional context is the primary driver of performance. The attention mechanism then acts as a powerful enhancement, further refining the model's predictive capabilities and leading to the superior overall results of our proposed method.

G. Model Efficiency and Versatility

Our proposed model is highly efficient in terms of both complexity and computational performance, demonstrating its practicality for diverse deployment scenarios. With only **175,118 trainable parameters** and a final size of **688 KB**, the model is lightweight. This efficiency is further highlighted by its performance across different hardware platforms. On a high-performance **NVIDIA Tesla T4 GPU**, the model achieves exceptional inference speed, processing data at over **19360 samples/second** with an average latency of just **0.0517 ms** per sample.

To assess its viability for edge computing, the model was also deployed on a **NanoPi microcontroller**. On this resource-constrained device, it maintained a practical performance, achieving a throughput of approximately **20 samples/second** with an average latency of **48.2 ms** per sample. This combination of high predictive accuracy, low model complexity, and strong performance on both high-end servers and low-power edge devices confirms that our proposed model is not only effective but also highly versatile for real-world applications.

V. Hardware Implementation and Real-Time Deployment

To evaluate the feasibility of deploying AI-powered forecasting models in real-world environmental monitoring scenarios, we implemented a complete embedded system using the NanoPi NEO Plus2 board. This section details both the hardware architecture and the software configuration necessary to support real-time inference at the edge.

A. System Architecture

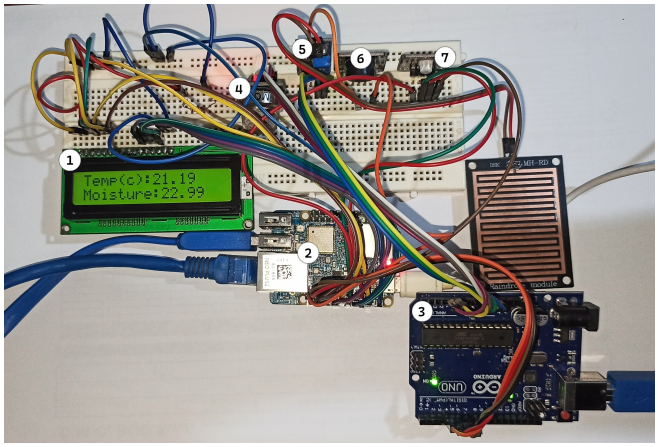

Fig. 8: Real-time environmental monitoring system architecture. The system consists of: (1) I2C LCD display for user feedback, (2) NanoPi NEO Plus2 for AI inference and decision-making, (3) Arduino UNO as an analog-to-digital converter (ADC), (4) MQ2 gas sensor, (5) soil moisture sensor, (6) buzzer for acoustic alerts, and (7) RGB LED for visual status indication. Data flows from sensors to the Arduino, then via UART to the NanoPi, where the deep learning model runs locally.

Environmental data collected by the sensors are transmitted via UART from the Arduino UNO to the NanoPi. Sensor data can be streamed via standard ADC interfaces or UART protocols to the NanoPi, which performs on-device normalization and forecasting. The system supports configurable sampling rates and buffering logic to ensure low-latency, continuous inference. The NanoPi runs a pre-trained deep learning model in Python, executing inference in real time to assess environmental conditions. Sensor readings (e.g., temperature and moisture) are shown on the LCD, and alerts are triggered using the buzzer and RGB LED. These alerts are currently based on simple threshold logic rather than AI output, ensuring minimal latency during actuation. The experimental hardware setup is illustrated in Figure 8.

B. Deployment on the NanoPi NEO Plus2

Due to the absence of an HDMI port, the NanoPi board was accessed via SSH. The board was connected to a local network using a USB WiFi dongle and assigned a static IP address during the initial setup. Debian 12 (64-bit) was selected as the operating system to support critical libraries like NumPy and Tensorflow-lite.

Access to GPIO ports from the virtual environment required elevated permissions, as reading digital pin data typically requires `sudo`. To solve this, a dedicated Linux group was created and assigned permissions to access the necessary device files. This approach enabled GPIO interaction without `sudo`, which is typically restricted in virtual environments.

C. Analog-to-Digital Conversion Strategy

The NanoPi board does not include native analog input pins. To read analog sensors such as the MQ2 gas sensor and the soil moisture sensor, we integrated an Arduino UNO to serve solely as an analog-to-digital converter (ADC). The Arduino continuously reads analog voltages, digitizes the values, and transmits them to the NanoPi via UART. No computation occurs on the Arduino side.

This architectural choice keeps the system simple while enabling analog interfacing. However, the Arduino dependency could be eliminated in future iterations by replacing it with a dedicated ADC IC (e.g., `ADS1115`), allowing the NanoPi to operate all sensors and actuators independently.

D. Inference on Edge

The pre-trained model was exported in a lightweight format and executed on the NanoPi using optimized Python scripts. This validates the practicality of deploying multivariate deep learning models on low-power, low-cost hardware for environmental monitoring.

VI. Conclusion and Future Work

This paper introduced an efficient and accurate attentional bidirectional LSTM model for multivariate time series forecasting, specifically designed for edge devices. It outperformed existing methods on the Jena Climate dataset, with analyses confirming that attention and bidirectionality are key to its success. The model is lightweight, fast, and suitable for real-world edge deployment on both GPUs and low-power devices like NanoPi. Future work includes optimizing the hardware setup, enabling multi-step forecasting, integrating predictions into on-device decision-making, and testing across diverse datasets. The study offers a practical road map for bringing deep learning to edge environments.

An immediate extension of this work is to implement an efficient data communication pipeline using the MQTT protocol. This would enable real-time transmission of sensor data and forecast results from the NanoPi edge device to a web-based weather prediction dashboard. Leveraging MQTT's lightweight publish-subscribe mechanism ensures minimal latency and bandwidth usage, making it ideal for constrained edge environments. Integrating this communication layer will facilitate remote monitoring, visualization, and user interaction with the forecasting system, bridging the gap between on-device processing and user-facing applications.

References

[1] A. Utku and U. Can, "An efficient hybrid weather prediction model based on deep learning," *International Journal of Environmental Science and Technology*, vol. 20, no. 10, pp. 11 107–11 120, 2023.

[2] Ö. A. Dombaycı and M. Gölcü, "Daily means ambient temperature prediction using artificial neural network method: a case study of turkey," *Renewable Energy*, vol. 34, no. 4, pp. 1158–1161, 2009.

[3] X. Liu, S. Yuan, and L. Li, "Prediction of temperature time series based on wavelet transform and support vector machine," *Journal of Computers*, vol. 7, no. 8, pp. 1911–1918, 2012.

[4] S. Salcedo-Sanz, R. C. Deo, L. Carro-Calvo, and B. Saavedra-Moreno, "Monthly prediction of air temperature in australia and new zealand with machine learning algorithms," *Theoretical and Applied Climatology*, vol. 125, pp. 13–25, 2016.

[5] J. Jasmine *et al.*, "Advanced weather prediction based on hybrid deep gated tobler's hiking neural network and robust feature selection for tackling environmental challenges," *Global NEST Journal*, vol. 27, no. 4, pp. 1–13, 2025.

[6] O. Kisi and J. Shiri, "Prediction of long-term monthly air temperature using geographical inputs," *International Journal of Climatology*, vol. 34, no. 1, pp. 179–186, 2014.

[7] D. Fister *et al.*, "Accurate long-term air temperature prediction with machine learning models and data reduction techniques," *Applied Soft Computing*, vol. 136, no. 4, p. 110118, 2023.

[8] R. M. Schmidt, "Recurrent neural networks (rnns): A gentle introduction and overview," 2019. [Online]. Available: https://arxiv.org/abs/1912.05911

[9] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, pp. 1735–1780, 11 1997.

[10] C. Li, Y. Zhang, and G. Zhao, "Deep learning with long short-term memory networks for air temperature predictions," in *2019 International Conference on Artificial Intelligence and Advanced Manufacturing (AIAM)*, Dublin, Ireland, 2019, pp. 243–249.

[11] C. Li *et al.*, "Air temperature forecasting using traditional and deep learning algorithms," in *2020 7th International Conference on Information Science and Control Engineering (ICISCE)*, Changsha, China, 2020, pp. 189–194.

[12] F. Xiao, "Time series forecasting with stacked long short-term memory networks," arXiv preprint, 2020, arXiv:2011.00697. [Online]. Available: https://arxiv.org/abs/2011.00697

[13] Z. Huang, W. Xu, and K. Yu, "Bidirectional lstm-crf models for sequence tagging," 2015. [Online]. Available: https://arxiv.org/abs/1508.01991

[14] J. Hou *et al.*, "Prediction of hourly air temperature based on cnn–lstm," *Geomatics, Natural Hazards & Risk*, vol. 13, no. 1, pp. 1962–1986, 2022.

[15] O. Pooladzandi and Y. Zhou, "Improving levenberg–marquardt algorithm for neural networks," arXiv preprint, 2022, arXiv:2212.08769. [Online]. Available: https://arxiv.org/abs/2212.08769

[16] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," 2016. [Online]. Available: https://arxiv.org/abs/1409.0473

[17] M. Dixon, "Dataset: Jena climate dataset," Dataset; DOI: 10.57702/mf-brilnw, 2024, retrieved via DOI on December 16, 2024.

[18] FriendlyElec, "Nanopi neo plus2 single-board computer," Hardware specification; Allwinner H5 (quad-core Cortex-A53 ARMv8), 1 GB DDR3, 8 GB eMMC, Gbps Ethernet, Wi-Fi, Bluetooth 4.0, USB, microSD; Size 40×52 mm, 2017, accessed 2025-08-10; see FriendlyElec wiki and official spec pages.

[19] M. Hossain *et al.*, "Forecasting the weather of nevada: a deep learning approach," in *International Joint Conference on Neural Networks (IJCNN)*, Killarney, Ireland, 2015, pp. 1–6.