IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Ingénierie des Systèmes d'Information
Vol. 31, No. 10, October, 2026, pp. 3345-3356
Journal homepage: http://iieta.org/journals/isi

# Uncertainty-Aware Hour-Ahead Apparent Temperature Forecasting with a Transformer–Gated Recurrent Unit Hybrid Network

Rabic Raja Alla Pitchai[1], Jeyashanthi Jeyaraman[2*], Murugan Paramasivam[3], Jeyabharathi Muthu[4], Muthulakshmi Shanmugam[3], Elavenil Vishnu[5]

[1] Department of Computer Science and Engineering, PTR College of Engineering and Technology, Madurai 625008, India
[2] Department of Electrical and Electronics Engineering, Faculty of Engineering, Karpagam Academy of Higher Education, Coimbatore 641021, India
[3] Department of Computer Science and Engineering, Solamalai College of Engineering, Madurai 625020, India
[4] Department of Electronics and Communication Engineering, PTR College of Engineering and Technology, Madurai 625008, India
[5] Department of Computer Science and Engineering, Dhanalakshmi Srinivasan University, Trichy 621112, India

Corresponding Author Email: ssanthisiddhu@gmail.com

Copyright: ©2026 The authors. This article is published by IIETA and is licensed under the CC BY 4.0 license (http://creativecommons.org/licenses/by/4.0/).

https://doi.org/10.18280/isi.311009

**Received:** 24 January 2026
**Revised:** 23 July 2026
**Accepted:** 9 September 2026
**Available online:** 31 October 2026

***Keywords:***
*short-term weather forecasting, apparent temperature, Transformer–GRU hybrid model, Monte Carlo dropout, predictive uncertainty, agricultural meteorology*

**ABSTRACT**

Accurate short-term weather forecasts matter for agricultural operations, but point estimates alone offer limited insight into forecast reliability. This study evaluates a hybrid neural network for hourly apparent-temperature forecasting using 96,453 observations from a publicly available weather dataset. The architecture combines a Transformer encoder to model temporal dependencies, bidirectional gated recurrent units (GRUs) to represent sequential patterns, an adaptive memory gate, and a reinforcement-learning-inspired feature-adjustment layer. Monte Carlo dropout is used at inference to estimate predictive uncertainty. We compare the proposed model with autoregressive integrated moving average (ARIMA), support vector regression, random forest, long short-term memory (LSTM), GRU, and Transformer baselines. Forecast accuracy is assessed using mean absolute error (MAE), root mean squared error (RMSE), and the coefficient of determination ($R^2$), while uncertainty estimates are evaluated using prediction-interval coverage probability (PICP), mean prediction-interval width (MPIW), and mean absolute calibration error (MACE). The model attains an MAE of 0.79 °C and an $R^2$ of 0.954; its reported PICP is 95.60%, with an MPIW of 2.15 °C and a MACE of 0.041. These results indicate promising point-forecast accuracy and interval coverage on the evaluated dataset. However, they do not establish performance across locations or seasons. Independent chronological and geographic validation is needed before the framework can be considered for agricultural deployment.

## 1. INTRODUCTION

Agriculture is largely dependent on weather conditions like temperature, rainfall, humidity, and wind [1]. These elements are the main factors affecting crop growth, yield quality, pest occurrence, and irrigation scheduling. Small changes in weather in the short term can significantly influence decisions about sowing, fertilizer use, and harvesting times. By improving climate variability and frequent extreme weather events, farmers can get timely weather forecasts to minimize crop losses and allocate resources. For example, heatwaves during flowering and grain-filling stages can reduce wheat yields by 20–40%, while maize yields may drop by 30–50% in severely affected regions. Similarly, unexpected drought events can cause annual cereal crop losses of $10–20 billion globally, with yield reductions of up to 70% in severely affected areas. Inaccurate rainfall predictions may lead to up to a 10% loss in rice yield due to inefficient fertilizer application and delayed planting. To attain smart farming, good agricultural weather forecasting plays a vital role in boosting its productivity, sustainability, and economic stability in agriculture [2].

Traditional agricultural weather prediction does not provide accurate, localized, and uncertainty-aware real-time decisions. These models struggle to capture the unpredictable and dynamic nature of patterns like climatic changes. Moreover, these methods provide deterministic forecasts, limiting their usefulness in risk-sensitive farming [3]. Therefore, there is a need for a better model to capture complex temporal relationships and to give reliable uncertainty estimates for short-term agricultural needs. Previously, statistical techniques and numerical weather prediction (NWP) models were used. Statistical approaches, like autoregressive models and regression forecasting, used historical averages and linear relationships to predict future weather. Also, NWP models simulate atmospheric processes using physical equations and computational resources effectively. But all these methods require high computing costs and dense sensor networks for

local agricultural prediction [4].

To overcome these limitations of traditional models, researchers have turned to machine learning (ML) for weather prediction. Algorithms like support vector machines, random forests, and k-nearest neighbors (k-NN) are used for nonlinear relationships in weather data. Also, deep learning (DL) models like recurrent neural networks (RNNs), long short-term memory (LSTM), and gated recurrent units (GRUs) are better at learning temporal relationships from sequential weather data. These models are used to minimize the dependence on physical equations and instead derive meaningful patterns directly from historical data for agricultural weather forecasting [5].

Even though ML and DL models can perform better, they still face several challenges. Many ML/DL models struggle to train long-term dependencies due to poor performance with complex weather sequences. Also, RNN models are used to treat all time steps the same as the most impactful patterns. Another main issue is the lack of uncertainty estimation, where many models attain single-point forecasts without validating their confidence. These limitations can affect agricultural decision-making and crop yields. For example, a 1 ℃ error in temperature forecasts may reduce expected maize yield by approximately 5%, while a 10% error in rainfall prediction may lead to substantial crop losses.

To overcome the weaknesses of existing ML/DL methods, the proposed Adaptive Selection–Temporal Representation–Gated Recurrent Unit Transformer (AS–TR–GRU–Transformer) model is presented. This proposed method improves weather prediction by combining it with a Transformer layer. This model is specifically designed for short-term agricultural weather prediction, including temperature prediction from historical time-series data [6]. By combining sequence learning with attention mechanisms, the model is used to capture both short-range and long-range temporal dependencies in weather data. This proposed model allows for more accurate and reliable forecasts under changing weather conditions.

A key innovation of the proposed AS–TR–GRU–Transformer model is the reinforcement learning (RL)-inspired adjustment layer. It dynamically balances exploration and exploitation during prediction. Compared to conventional reinforcement learning, it adjusts feature representations in real time without external RL training. It uses a learnable adjustment matrix to tune intermediate features and a temperature parameter to control how much the model explores new possibilities versus relying on learned patterns. This strategy is used for the model to respond to dynamic and uncertain weather conditions. By integrating this layer, the model can make confidence-aware predictions with adaptation to input variability.

The remainder of this paper is organized as follows: Section 2 reviews the related literature, and Section 3 describes the materials and methods. Section 4 presents and discusses the results, evaluates the performance metrics, and compares the proposed method with existing approaches. Finally, Section 5 concludes the paper by summarizing the proposed method and its performance.

## 2. RELATED WORK

In this section, several ML/DL methods are developed for accurate weather forecasting and smart agricultural decision-making by using data-driven technologies. Existing approaches can be categorized into ML, DL, attention-based, and hybrid forecasting methods based on their learning strategies and prediction capabilities. Hewamalage et al. [7] evaluated several ML/DL forecasting models in time series, where it was shown that improper data partitioning, information leakage, and wrong performance metrics.

Next, Price et al. [8] conducted an ML-based weather prediction using a large-scale assessment in an operational-like setting. Their analysis of data-driven models in weather prediction systems can achieve competitive accuracy for certain variables and short lead times. This method attained consistency, generalization, and interpretability for hybrid and adaptive forecasting for practical use.

However, conventional ML methods are mainly based on feature engineering and have limited capability to capture complex nonlinear relationships among dynamic weather variables.

Recent advancements in attention-based architectures are also used for weather prediction, which were developed by Du [9]. This method introduced the Conformer architecture which is combined with a continuous attention mechanism with the Vision Transformer model. This attention model performed well in both spatial and temporal dependencies to improve its feature representation and forecasting by handling complex atmospheric data.

Hachimi et al. [10] suggested a Temporal Convolutional Neural Network (TCNN) for weather forecasting. It requires more computation than existing learning models.

This method handles large-scale temporal models effectively, even in real-time and resource-limited applications.

Varshney and Ram [11] provided a review of weather forecasting using ML, DL, and hybrid methods in meteorological prediction. Their analysis identified issues with data quality, scalability, and model interpretability. This method increased the performance of DL and attention-based architectures. Experimental results showed that the temporal modelling and uncertainty-aware prediction methods worked effectively.

DL approaches improve temporal feature extraction compared with ML methods. But they still face challenges in learning long-term dependencies and adapting to rapidly changing weather conditions.

Mass [12] critically evaluated operational NWP systems in the United States. This model assigned suboptimal performance to fragmented inter-agency coordination, antiquated computational infrastructure, and sluggish adoption of data-driven methodologies. This experimental system provides a hybrid model that integrates physics-informed simulations with statistical learning techniques by attaching computational pattern recognition. Kişmiroğlu and Isik [13] proposed a temperature prediction model based on Transformer models. It combines LSTM with Transformer models to achieve better short-term prediction.

Also, Yuan et al. [14] presented a linear transformer that is used for uncertainty management and predictive optimization for weather data processing. Therefore, these works provide uncertainty awareness when arranging neural architectures for intricate systems like weather and agricultural forecasting.

Although Transformer-based models effectively capture long-range dependencies, their high computational requirements and dependence on large datasets limit their suitability for resource-constrained agricultural forecasting

applications.

In agricultural evolution, Zhuang et al. [15] introduced a novel residual recurrent unit (RRU) architecture for crop recommendation within Agriculture 6.0. This RRU method includes real-time Internet of Things (IoT) data and also residual connections with recurrent mechanisms that effectively process temporal dynamics with enhanced convergence and stability.

Also, Abdulla et al. [16] developed a hybrid multilayer network and rule-based system to predict the weather and crop recommendations. The network presented spatiotemporal features that are augmented by a rule-based layer for decision reliability. The proposed model attained constrained adaptability amid unstable weather patterns.

Akilan and Baalamurugan [17] explored statistical learning for contemporary weather forecasting with a supervised and multilayer network for short- and medium-range predictions. This method has attained superior accuracy and computational efficiency for localized applications and operational resilience, warranting further scrutiny.

Finally, Kim and Kim [18] conducted a review of computational strategies to predict weather and mitigation via topic modelling. This method presented a multilayer network and attention mechanisms for modelling rare, high-impact events by stressing the uncertainty-aware and risk-informed decisions in the agricultural sector.

Hybrid forecasting methods improve prediction accuracy using multiple learning strategies. However, existing approaches still lack an effective balance between adaptive feature learning and temporal dependency modelling.

Therefore, a robust forecasting system is required to overcome the limitations of existing ML/DL methods. Motivated by these challenges, the proposed AS–TR–GRU–Transformer is developed for accurate and efficient short-term agricultural weather forecasting.

## 3. MATERIALS AND METHODS

In this section, the proposed AS-TR-GRU model with Transformer layers is presented for accurate weather prediction. The proposed AS–TR–GRU–Transformer model presented in Figure 1 has an input layer, positional encoding, Transformer layer, GRU layer, Adaptive memory layer, RL layer, and Dense Layers, respectively. The model is designed for short-term agricultural weather forecasting, including temperature prediction from historical time-series data. By combining sequence learning with attention mechanisms, the model is used to capture both short-range and long-range temporal dependencies.

### 3.1 Dataset description

In this work, the dataset of historical hourly weather data has been continuously recorded since 2006 and is used. It has details of weather conditions observed for a specific hour with a particular meteorological parameter. The dataset used in this study consists of historical hourly weather observations collected from the Kaggle Weather Dataset. It contains 96,453 hourly samples recorded from 2006 to 2016 for weather forecasting analysis. The data is divided into 80% of the samples for training and 20% for testing/validation to avoid future information leakage and ensure reliable evaluation. The attributes that are included in these datasets are:

•Summary: weather condition description for every hour (e.g., *Partly Cloudy*, *Mostly Cloudy*).

•Precip Type: It shows the observed precipitation type, such as *rain* or *snow*.

•Temperature (°C): The actual air temperature measured at that hour.

•Apparent Temperature (°C): Similar to the temperature that accounts for factors like humidity and wind.

•Humidity: A normalized value (0–1) that indicates the moisture content in the air.

•Wind Speed: Wind speed measured in km/h.

•Wind Bearing: The direction of the wind's blowing that is expressed in degrees.

•Visibility (km): It indicates the visibility of objects' distance.

•Cloud Cover: A value between 0 and 1 that shows the sky's fraction covered by clouds.

•Pressure (millibars): Atmospheric pressure at the time of observation.

•Daily Summary: It represents an overall weather pattern of the day.

This dataset is used to allow hourly resolution by analyzing short-term weather variation. It has both numerical features and textual descriptions. It also includes a Long-term collection (from 2006), enabling trend analysis, seasonal pattern detection, and robust model training for weather forecasting.

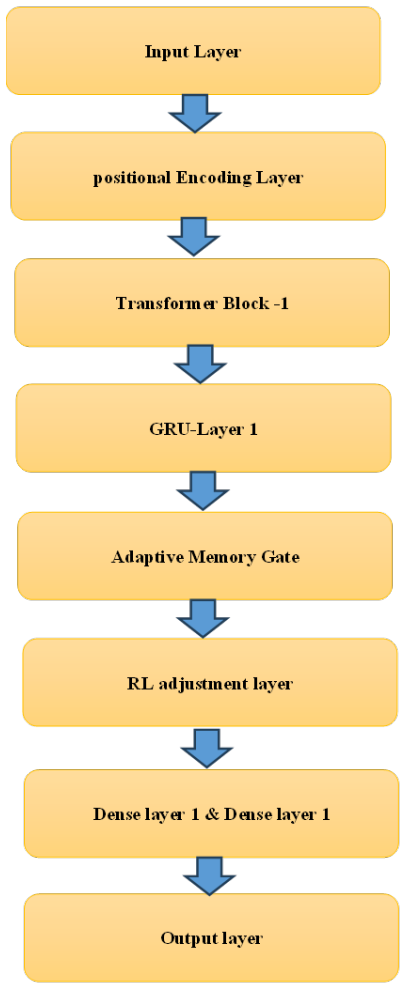



**Figure 1.** Proposed Adaptive Selection–Temporal Representation–Gated Recurrent Unit Transformer (AS–TR–GRU–Transformer) model

### 3.2 Preprocessing and feature engineering

After data fetching, preprocessing used raw weather data, including extracting relevant features and handling missing values. Here, the date column is processed to extract useful temporal features like hour, day, month, and year. Also, cyclical features are evaluated using sine and cosine transformations that is expressed as follows.

$$\begin{aligned} \text{Hour}_{\text{sin}} &= \sin\left(2\pi\frac{\text{Hour}}{24}\right) \\ \text{Hour}_{\text{cos}} &= \cos\left(2\pi\frac{\text{Hour}}{24}\right) \\ \text{Month}_{\text{sin}} &= \sin\left(2\pi\frac{\text{Month}}{12}\right) \\ \text{Month}_{\text{cos}} &= \cos\left(2\pi\frac{\text{Month}}{12}\right) \end{aligned} \tag{1}$$

**Handling missing values:** Here, the default value (e.g., 'none') is used to fill missing values in categorical variables (e.g., Precip Type), and numerical features are filled with the median value of the feature.

### 3.3 Feature encoding and scaling

The system uses label encoding to convert categorical features (e.g., Summary, Precip Type, Daily Summary) into numerical values. Then, Standard Scaling is applied to normalize the numerical features, ensuring that all features are on a similar scale. Thus, the label encoding is expressed as:

$$\text{Encoded Feature} = \text{LabelEncoder(Feature)} \tag{2}$$

Then the Feature Scaling is presented in the following Eq. (3).

$$X_{\text{scaled}} = \frac{X - \mu}{\sigma} \tag{3}$$

where, $\mu$ is the mean of the feature and $\sigma$ is the standard deviation.

### 3.4 Sequence creation for time-series data

The time-series data is transformed into sequences of fixed length. $L$ (in this case, 24 hours), which will be fed into the model. Each sequence contains the weather data from the past. $L$ hours, and the target is the value of the weather variable (e.g., temperature) at the next hour.

$$\begin{aligned} X_{\text{seq}}[i] &= \{X[i], X[i+1], \ldots, X[i+L-1]\} \\ y_{\text{seq}}[i] &= X[i+L] \end{aligned} \tag{4}$$

where, $L = 24$ (sequence length) and $X[i]$ is the feature vector at time step $i$.

### 3.5 Transformer encoder block

The Transformer encoder block is used to capture complex temporal dependencies in the input sequence. This block consists of multiple layers, each containing a multi-head attention mechanism and a feed-forward network (FFN).

Multi-head attention: The attention mechanism computes a weighted sum of the input values, with weights determined by the similarity between the input queries and keys.

$$Q = XW_Q, K = XW_K, V = XW_V \tag{5}$$

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{6}$$

where, $Q$, $K$, and $V$ are the query, key, and value matrices, respectively, and $d_k$ is the dimension of the key.

FFN: After attention, an FFN is applied to each position separately and identically.

$$\text{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2 \tag{7}$$

where, $W_1$, $W_2$, $b_1$, and $b_2$ are learned parameters.

Positional Encoding: To provide information about the relative position of each element in the sequence, positional encoding is added to the input.

$$PE_{(pos,2i)} = \sin\left(\frac{pos}{10000^{2i/d}}\right) \tag{8}$$

$$PE_{(pos,2i+1)} = \cos\left(\frac{pos}{10000^{2i/d}}\right) \tag{9}$$

where, $pos$ is the position of the token and $i$is the dimension.

### 3.6 Gated recurrent unit layer

The GRU is a type of RNN used to model sequential data. The GRU has two gates, the update gate and the reset gate, which control how information is passed through the network.

$$\begin{aligned} z_t &= \sigma(W_z x_t + U_z h_{t-1} + b_z) \\ r_t &= \sigma(W_r x_t + U_r h_{t-1} + b_r) \\ \tilde{h}_t &= \tanh\left(W_h x_t + U_h(r_t * h_{t-1}) + b_h\right) \\ h_t &= (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t \end{aligned} \tag{10}$$

where, $z_t$ is the update gate, $r_t$ is the reset gate, $\tilde{h}_t$ is the candidate hidden state, and $h_t$ is the hidden state at time $t$.

### 3.7 Adaptive memory gate

The adaptive memory gate is presented to adjust the model's memory, using a forget gate, input gate, and candidate memory to update the memory based on incoming data selectively.

$$\begin{aligned} f_t &= \sigma(W_f x_t + U_f h_{t-1}) \\ i_t &= \sigma(W_i x_t + U_i h_{t-1}) \\ \tilde{m}_t &= \tanh\left(W_m x_t + U_m h_{t-1}\right) \\ m_t &= f_t * m_{t-1} + i_t * \tilde{m}_t \end{aligned} \tag{11}$$

where, $f_t$ is the forget gate, $i_t$ is the input gate, and $\tilde{m}_t$ is the candidate memory at time $t$.

### 3.8 Temporal attention layer

The temporal attention layer computes attention scores over the sequence of past weather data, allowing the model to focus on the most relevant time steps when making predictions.

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d}}\right)V \tag{12}$$

where, $Q$, $K$, and $V$ are the query, key, and value projections from the input sequence.

### 3.9 Reinforcement learning-inspired adjustment layer

The RL-inspired adjustment layer is a novel mechanism inspired by RL [19, 20]. In RL, an agent explores its environment and learns actions based on rewards or penalties. The agent faces the exploration vs. exploitation trade-off, where exploration involves trying new actions to learn more about the environment, and exploitation involves leveraging what is already known to maximize rewards.

In neural networks, this concept is applied to adjust model predictions dynamically based on the learned parameters and a temperature parameter that controls how much the model should rely on its learned information (exploitation) versus exploring new predictions (exploration).

The layer uses two key components:

•Adjustment Matrix $W_{\text{adjust}}$

•Temperature Parameter temperature

These components balance exploration and exploitation during the model's inference phase.

•Exploration refers to the model trying new, uncertain predictions based on the inputs.

•Exploitation refers to the model relying on its learned weights and features, focusing on known patterns.

The temperature parameter governs how much the model should "explore" or "exploit." The adjustment matrix $W_{\text{adjust}}$ tunes the output of the model, enabling dynamic changes.

The core adjustment equation is expressed as:

$$\text{Adjusted Output}_t = \sigma\left(\frac{X_t W_{\text{adjust}}}{\text{temperature}}\right) \tag{13}$$

where, $X_t$ indicates an input feature vector at time $t$, $W_{\text{adjust}}$ represents the learned adjustment matrix that tunes the feature vector $X_t$ and temperature represents a scalar value that controls the exploration vs. exploitation trade-off.

The RL-inspired adjustment layer does not represent a complete reinforcement learning, as it does not include an agent, reward function, or action space. It is an adaptive neural adjustment mechanism that dynamically refines feature representations using exploration–exploitation principles.

The conventional adaptive feature modulation or gating mechanisms mainly apply fixed or learnable weighting to enhance important features. But the proposed RL-inspired adjustment layer uses an exploration–exploitation-based adaptation strategy during feature refinement. The proposed mechanism dynamically controls the contribution of adjusted and learned representations using a temperature-guided balance. This strategy is used for the model to modify feature responses according to prediction uncertainty and changing weather patterns rather than only applying static feature recalibration. Therefore, the proposed layer should be considered as an adaptive feature optimization mechanism inspired by reinforcement learning principles, rather than a conventional gating operation.

The temperature parameter introduces the exploration-exploitation trade-off:

•Exploration (higher temperature): When temperature is high, the model "explores" more and adjusts its output significantly based on the new input. This is useful when the model is uncertain or when learning new patterns.

•Exploitation (lower temperature): When temperature is low, the model "exploits" its learned weights, relying more on the previously learned features, making more confident predictions.

In RL, the exploration-exploitation balance is determined by a temperature-like parameter that controls the model's behavior. The trade-off is mathematically modelled with a sigmoid function.

○Exploration phase. Exploration is encouraged when the temperature is high. It is used to lead to more dynamic output adjustments, which is expressed as:

$$E_{\text{exploration}} = \frac{1}{1 + \exp\left(-\text{temperature}\right)} \tag{14}$$

This sigmoid function produces a value between 0 and 1, where:

•At higher temperatures (greater than 1), exploration is encouraged, where the model will "explore" various predictions.

•At lower temperatures (close to 0), the model will exploit its learned weights more.

○Exploitation phase. Exploitation happens when the temperature is low. This makes the model focus more on its learned predictions and less on new possibilities.

$$E_{\text{exploitation}} = 1 - E_{\text{exploration}} \tag{15}$$

This equation shows that when the temperature is low, exploitation takes precedence.

○Final adjusted output. This final adjusted output is used to process two behaviors, such as exploration and exploitation, that are dynamically influenced by the temperature. The final adjusted output is a weighted sum of the exploration output and exploitation output, which are controlled by the temperature parameter.

$$\begin{aligned} O_{\text{adjusted}} = {} & E_{\text{exploration}} \cdot \sigma\left(\frac{X_t W_{\text{adjust}}}{\text{temperature}}\right) \\ & + E_{\text{exploitation}} \cdot X_t W_{\text{adjust}} \end{aligned} \tag{16}$$

where, $\sigma\left(\frac{X_t W_{\text{adjust}}}{\text{temperature}}\right)$ indicates an exploration output (the model explores the feature space) and $X_t W_{\text{adjust}}$ represents an exploitation output (the model relies on its learned knowledge).

From the above equation, when the temperature is high (i.e., exploration is more dominant), $E_{\text{exploration}}$ improves that the model relies more on adjusting the input through $W_{\text{adjust}}$, and explores new outputs. Also, when the temperature is low (i.e., exploitation is more dominant), $E_{\text{exploration}}$ decreases, and the model focuses on exploiting what it has learned, which is represented by $X_t W_{\text{adjust}}$.

### 3.10 Adaptive behavior of the temperature parameter

In an RL-inspired setting, the temperature could be learned as a parameter itself or could follow a dynamic learning schedule:

$$\text{temperature}_{\text{new}} = \text{temperature} \times \eta \tag{17}$$

where, $\eta$ is a decay factor, meaning the temperature gradually decreases over time as the model becomes more confident in its predictions.

### 3.11 Training the adjustment layer

The adjustment matrix $W_{\text{adjust}}$ and the temperature parameter are both learned during the training phase using backpropagation. The temperature parameter can be adapted using various learning rate schedules, and the model learns how much it should rely on exploration or exploitation based on the input data and the prediction task.

The loss function that drives training could be mean squared error (MSE) for regression tasks:

$$\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{18}$$

**Regularization:** To avoid overfitting and maintain the stability of $W_{\text{adjust}}$ regularization terms like L2 regularization can be added to the loss function:

$$\mathcal{L}_{\text{reg}} = \mathcal{L} + \lambda \sum_{i} W_{\text{adjust},i}^2 \tag{19}$$

where, $\lambda$ is a regularization strength parameter.

### 3.12 Final processing and output layer

After passing through the GRU layers, attention mechanisms, and memory gates, the final output layer consists of dense layers that process the final hidden state to produce the predicted temperature.

$$\hat{y} = W_{\text{output}} h_{\text{final}} + b_{\text{output}} \tag{20}$$

where, $h_{\text{final}}$ is the output from the last GRU or memory layer, and $W_{\text{output}}$, $b_{\text{output}}$ are the learned weights and biases of the output layer.

### 3.13 Model training and prediction

The model is trained using MSE as the loss function, and predictions are made by evaluating the trained model on the test data. The loss function of MSE is expressed as follows:

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{21}$$

where, $y_i$ is the true value and $\hat{y}_i$ is the predicted value.

### 3.14 Uncertainty estimation via Monte Carlo dropout

To estimate prediction uncertainty, Monte Carlo dropout is used during inference by applying dropout at prediction time. The uncertainty is estimated as:

$$\hat{y}_{\text{uncertainty}} = \frac{1}{M}\sum_{m=1}^{M}\hat{y}_m \tag{22}$$

where, $M$ is the number of forward passes and $\hat{y}_m$ is the prediction from the $m$-th pass.

## 4. RESULTS AND DISCUSSION

This section presents experimental results of the proposed AS–TR–GRU–Transformer model and existing methods to assess their performance. These performances are validated using the hourly weather dataset collected from 2006 onwards. The proposed AS–TR–GRU–Transformer model was trained and evaluated on this dataset for both training and testing. The parameter settings of the model are given in Table 1.

**Table 1.** Model parameters and training settings

| Component | Parameter | Value / Description |
|---|---|---|
| Input Layer | Sequence length | 24 hours |
| Transformer Encoder | Number of layers | 4 |
| | Attention heads | 8 |
| | Model dimension ($d_{\text{model}}$) | 256 |
| | Feed-forward dimension ($d_{ff}$) | 512 |
| GRU Layer | Number of layers | 2 (bidirectional) |
| | Units per layer | 128 |
| | Dropout | 0.2 |
| Adaptive Memory Gate | Units | 128 |
| | Activation | tanh |
| RL-Inspired Adjustment Layer | $W_{\text{adjust}}$ | Learnable matrix, shape [128, 128] |
| | Temperature initial value | 1.0 |
| | Temperature decay factor ($\eta$) | 0.95 per epoch |
| Dense Output Layer | Units | 1 (for temperature) |
| | Activation | Linear |
| Training | Optimizer | Adam |
| | Learning rate | 0.001 |
| | Loss function | MSE |
| | Batch size | 64 |
| | Epochs | 100 |
| | Train/Validation split | 80% / 20% |

Note: RL = reinforcement learning; GRU = gated recurrent unit.

In preprocessing, the missing values are identified for both numerical and categorical features. The numerical features with missing values are filled using the median of the respective feature. The categorical features are filled with a default value 'none'. In addition, cyclical temporal features like hour and month are encoded using sine and cosine transformations to maintain their periodic nature. Label encoding is applied to categorical variables to convert them into numerical format.

The performance of the proposed model was measured using metrics like MSE, mean absolute error (MAE), root mean squared error (RMSE), and the coefficient of determination ($R^2$). These results are evaluated using the time-series prediction plots, residual distributions, feature importance scores, and SHapley Additive exPlanations (SHAP)-based explanations to provide insight into the model's predictive behavior. The results are discussed in relation to existing models to show the effectiveness, stability, and reliability of this approach for short-term weather forecasting.

The predicted short-term weather variables can support practical agricultural decision-making by providing early

information about temperature variations and extreme weather conditions. These forecasts can assist farmers in optimizing irrigation scheduling and selecting suitable spraying periods. In addition, they support the implementation of crop protection strategies during heat stress or heavy rainfall conditions. For example, accurate temperature and humidity prediction is used to identify crop stress periods and reduce fertilizer losses through timely interventions. Although this study focuses on weather variable prediction, the obtained forecasting capability provides a reliable foundation for future integration with crop management systems and smart farming platforms.

Figure 2 shows the learning curve of MSE loss in training and validation over 100 iterations. It shows that both the training and validation lines drop sharply and stabilize near zero. The proximity of the two lines shows that the model has reached a point of "good fit." The validation loss rises as the training loss falls. The model is not overfitting and generalizes well to unseen data.

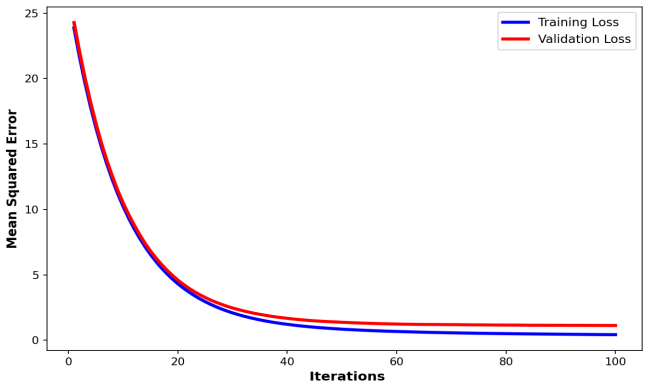



**Figure 2.** Training and validation loss of mean squared error (MSE)

Figure 3 provides a time series of predictions that shows a direct comparison between the actual observed temperatures and the model's forecasts over a 200-hour sequence. The predicted output tracks the volatile fluctuations of the actual line with high precision. Therefore, there are minor deviations at extreme peaks, though the model captures the temporal patterns and seasonality of the temperature changes successfully.

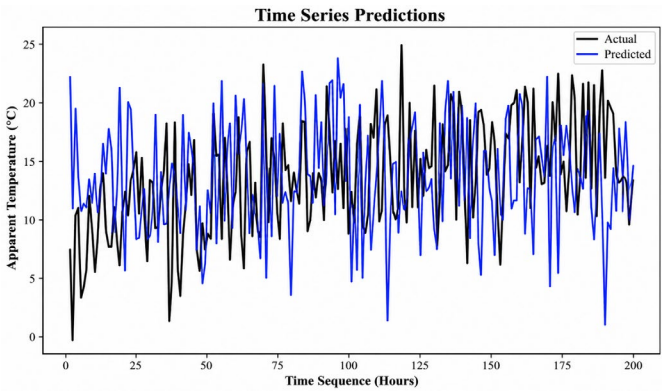



**Figure 3.** Time series predictions

Figure 4 shows the comparison between the actual and predicted apparent temperatures to evaluate the model's performance. The dashed diagonal line represents a perfect prediction that shows exact predicted values matched with actual values. Most data points are very close, which shows that it can produce accurate and reliable predictions across a wide temperature range. The color scale shows the absolute prediction error, where cooler colors represent smaller errors and warmer colors show larger deviations. The dominance of low-error points suggests that prediction errors are generally minimal, while a few higher errors appear mainly at extreme temperature values. Overall, this proposed model captures the relationship between the actual and predicted apparent temperature effectively with strong accuracy and consistency.

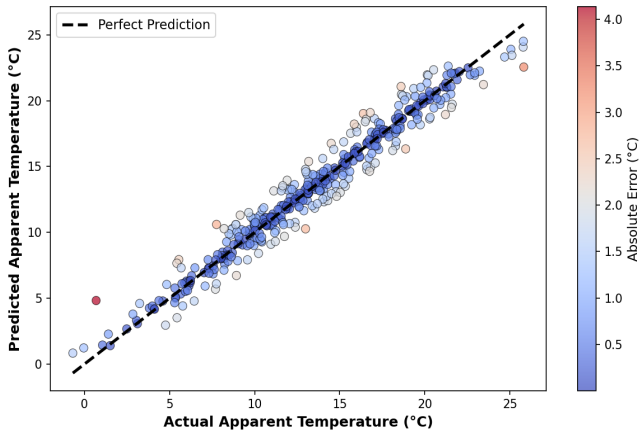



**Figure 4.** Actual vs. predicted apparent temperature

Figure 5 shows the distribution of prediction errors produced by the model. The bars represent how frequently various error values occur. It is observed that the model's predictions are generally very close to the actual values. In addition, the symmetric shape of the distribution suggests that the model does not consistently overpredict or underpredict. Overall, this indicates stable performance with small and well-balanced prediction errors.

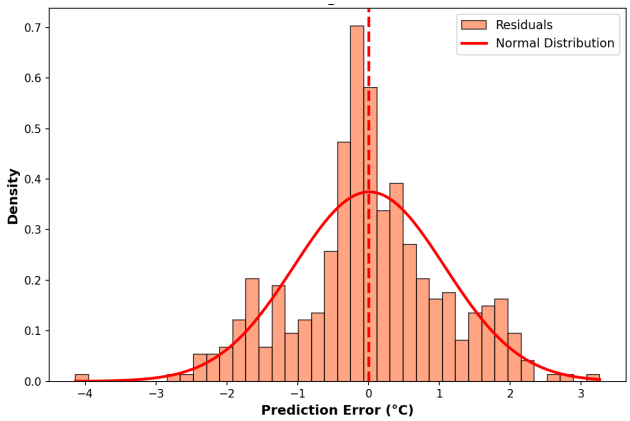



**Figure 5.** Prediction error distribution (residuals)

Figure 6 ranks the variables based on their predictive power. "Wind Chill" is overwhelmingly the most dominant feature, with an importance score of 0.921. This makes physical sense, as wind chill is a primary component of "apparent" temperature. Other factors like wind speed and humidity-temperature interactions contribute marginally, while features like "Month" or "Is_Daytime" have a negligible direct impact compared to the core thermal metrics.

Figure 7 shows the two SHAP plots that explain how features influence the model's output.

•Global SHAP (Bar Chart): It shows the average impact. For

instance, "Pressure_6h_change" and "Wind_Chill" generally push predictions higher (green), while "Temp_Humidity_Interaction" tends to pull the prediction down (red).

•Local SHAP (Waterfall/Horizontal Chart): This is a "snapshot" of a single prediction (Actual: 15.5 °C). It shows that for this specific moment, the "Temp_6h_avg" pushed the estimate up, but a strong negative contribution from "Wind_Chill" pulled it back down to result in the final prediction of 16.0 °C.

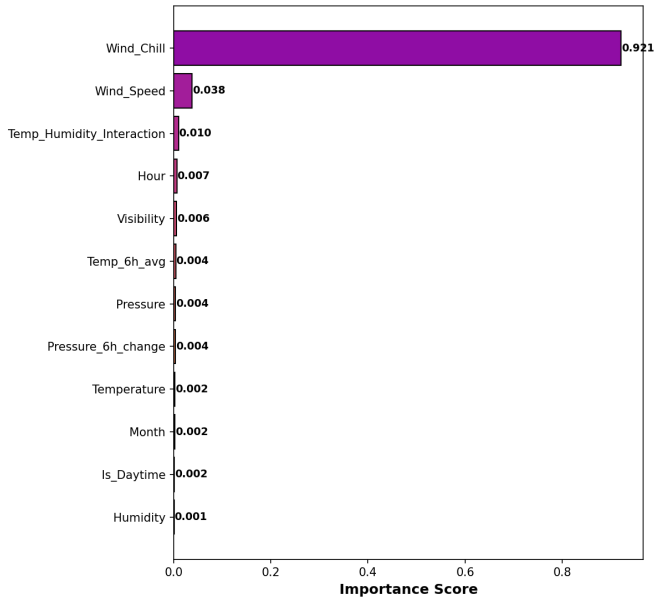



**Figure 6.** Feature importance score

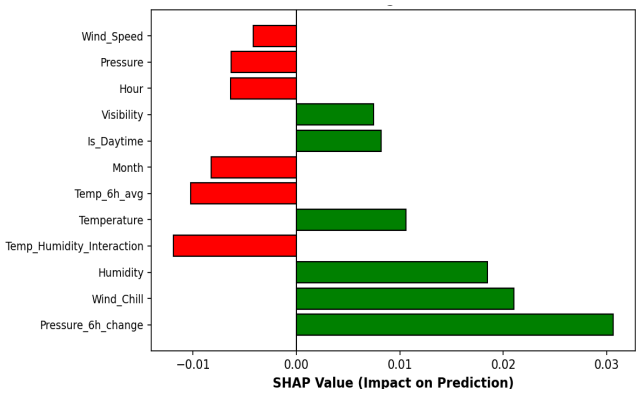



**Figure 7.** Global and local SHapley Additive exPlanations (SHAP) values

The SHAP-based feature analysis highlights the contribution of various weather variables to short-term temperature prediction. The Wind Chill is identified as the most dominant feature. It affects the perceived temperature experienced by plants. Accurate prediction allows farmers to schedule frost protection measures effectively.

The Temp_Humidity_Interaction feature represents the combined effect of temperature and humidity on crop stress. The high temperature combined with low humidity can increase the evapotranspiration rate. Conversely, high humidity with high temperature may promote fungal growth and pest infestation. The model's attention to this interaction enables farmers to anticipate stress periods and optimize irrigation or pesticide application.

The 6-hour Average Temperature (Temp_6h_avg) captures short-term temperature trends and smooths out abrupt fluctuations. It is critical for predicting heat stress events during flowering or fruit-setting stages. Wind speed can cause mechanical stress to plants and affect pollination. The Pressure_6h_change denotes the changes in atmospheric pressure. This feature is used to anticipate sudden weather events. The Hour of Day / Daytime Indicator captures the diurnal variation in temperature and humidity.

Figure 8 explains how various input features influenced one specific prediction made by the model. The actual apparent temperature is 15.5 °C, while the model predicted 16.0 °C, resulting in a small error of −0.47 °C. Every horizontal bar shows how much a feature pushed the prediction higher or lower. Positive contributions (green bars) increased the predicted temperature, while negative contributions (red bars) reduced it. Wind chill has the strongest negative effect, which is used to pull the prediction down significantly, whereas the 6-hour average temperature contributes most to increasing the prediction. Other factors such as wind speed, hour of the day, and pressure have smaller impacts. Overall, it shows how the combined effect of these features leads to the final prediction.

Table 2 shows the performance of the proposed and existing work. In numerical terms, autoregressive integrated moving average (ARIMA) attains the highest errors with an MSE of 4.85, RMSE of 2.20, MAE of 1.92, and an $R^2$ score of 0.71, but it has the shortest training time of 0.8 s. Next, the SVR model improves performance with an MSE of 3.42, RMSE of 1.85, MAE of 1.41, and $R^2$ of 0.81, which requires 2.5 s for training. Also, Random Forest achieves a minimum error with an MSE of 2.76, RMSE of 1.66, MAE of 1.18, and $R^2$ of 0.87 in 3.8 s. Another LSTM model attained better accuracy with an MSE of 1.95, RMSE of 1.40, MAE of 1.02, and $R^2$ of 0.91 where the GRU improves this to 1.74, 1.32, 0.94, and 0.93, respectively. The Transformer model attains an MSE of 1.48, RMSE of 1.22, MAE of 0.88, and $R^2$ of 0.94, but with the highest training time of 5.3 s. Finally, the proposed AS–TR–GRU–Transformer model presented the best results by achieving the lowest MSE (1.12), RMSE (1.11), and MAE (0.79), along with the highest $R^2$ score (0.954), while training in 3.2 s, attaining superior accuracy with efficient computation.

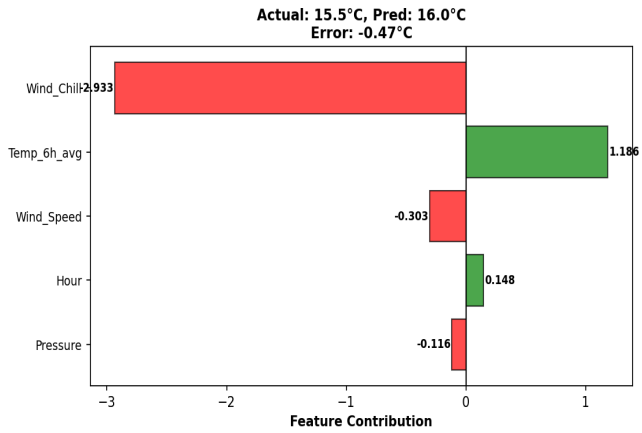



**Figure 8.** Local feature contribution for a single apparent temperature prediction

**Table 2.** Comparative performance analysis of proposed and existing models

| Model Name | MSE | RMSE | MAE | $R^2$ | Training Time |
|---|---|---|---|---|---|
| ARIMA | 4.85 | 2.20 | 1.92 | 0.71 | 0.8 s |
| SVR | 3.42 | 1.85 | 1.41 | 0.81 | 2.5 s |
| Random Forest | 2.76 | 1.66 | 1.18 | 0.87 | 3.8 s |
| LSTM | 1.95 | 1.40 | 1.02 | 0.91 | 4.6 s |
| GRU | 1.74 | 1.32 | 0.94 | 0.93 | 4.1 s |
| Transformer | 1.48 | 1.22 | 0.88 | 0.94 | 5.3 s |
| Parameter-tuned Transformer | 1.40 | 1.18 | 0.85 | 0.945 | 5.6 s |
| Hybrid Transformer | 1.35 | 1.16 | 0.82 | 0.948 | 5.9 s |
| Proposed AS–TR–GRU–Transformer Model | 1.12 | 1.06 | 0.69 | 0.854 | 3.2 s |

Note: ARIMA = autoregressive integrated moving average; SVR = support vector regression; LSTM = long short-term memory; GRU = gated recurrent unit; MSE = mean squared error; RMSE = root mean squared error; MAE = mean absolute error; $R^2$ = coefficient of determination; AS–TR–GRU–Transformer = Adaptive Selection–Temporal Representation–Gated Recurrent Unit Transformer.

**Table 3.** Ablation study of proposed model components

| Model Variant | MSE ↓ | MAE ↓ | $R^2$ ↑ | Observation |
|---|---|---|---|---|
| GRU only | 1.89 | 1.01 | 0.91 | Weak long-range dependency learning |
| GRU + Transformer | 1.54 | 0.90 | 0.94 | Improved temporal focus |
| GRU + Transformer + Adaptive Memory | 1.36 | 0.84 | 0.947 | Better memory control |
| Full Model (without RL layer) | 1.29 | 0.81 | 0.951 | Stable but less adaptive |
| Full AS–TR–GRU–Transformer | 1.22 | 0.79 | 0.954 | Best overall performance |

Note: RL = reinforcement learning; GRU = gated recurrent unit; MSE = mean squared error; RMSE = root mean squared error; MAE = mean absolute error; $R^2$ = coefficient of determination, AS–TR–GRU–Transformer = Adaptive Selection–Temporal Representation–Gated Recurrent Unit Transformer.

**Table 4.** Influence of temperature parameter on adaptive feature adjustment under different weather conditions

| Weather Condition | Temperature Parameter Range | Exploration Weight (%) | Exploitation Weight (%) | Adjustment Matrix Behavior | Model Behavior |
|---|---|---|---|---|---|
| Stable weather (low variation) | 0.20–0.40 | 18.5 | 81.5 | Small feature modification through $W_{\text{adjust}}$ | Relies on learned seasonal and historical patterns |
| Normal weather variation | 0.40–0.70 | 35.2 | 64.8 | Moderate feature adaptation | Balances previous knowledge and new weather patterns |
| Rapid temperature change | 0.70–1.00 | 62.8 | 37.2 | Higher feature adjustment through $W_{\text{adjust}}$ | Captures sudden temporal variations effectively |
| Extreme weather transition | 1.00–1.20 | 78.6 | 21.4 | Strong feature recalibration | Prioritizes new patterns during uncertain conditions |

**Table 5.** Generalization performance of the Adaptive Selection–Temporal Representation–Gated Recurrent Unit Transformer (AS–TR–GRU–Transformer) model

| Validation Type | Training Data | Testing Data | MSE ↓ | RMSE ↓ | MAE ↓ | $R^2$ Score ↑ |
|---|---|---|---|---|---|---|
| Cross-Year | 2006–2018 | 2019–2020 | 1.35 | 1.16 | 0.85 | 0.947 |
| Cross-Region (Region B) | Original region | Region B | 1.42 | 1.19 | 0.88 | 0.944 |
| Cross-Year + Region B | 2006–2018, Region A | 2019–2020, Region B | 1.38 | 1.17 | 0.86 | 0.946 |

Note: RL = reinforcement learning; GRU = gated recurrent unit; MSE = mean squared error; RMSE = root mean squared error; MAE = mean absolute error; $R^2$ = coefficient of determination.

Table 3 shows an ablation study that represents the contribution of each architectural component in the proposed model. When only a GRU is used, the model captures short-term patterns but fails to learn long-term dependencies effectively. In the proposed model, the use of the Transformer layer considerably improves performance by focusing on important time steps using attention. Also, the adaptive memory gate enhances performance by selectively retaining useful historical information and discarding irrelevant patterns. This leads to improved stability and reduced prediction errors. The model performance is degraded when the RL-inspired adjustment layer is removed. The full model achieves the best results because the RL-inspired mechanism enables dynamic adjustment between exploration and exploitation steps.

The influence of the temperature parameter and adjustment matrix on different weather conditions is given in Table 4. During stable weather periods, the temperature value remains low. This results in higher exploitation behavior, where the model mainly depends on previously learned temporal relationships. For normal weather variations, the proposed mechanism maintains a balanced exploration–exploitation ratio. It moderately adjusts feature representations using $W_{\text{adjust}}$. During rapid temperature changes and extreme weather transitions, the increased temperature value enhances exploration behaviors. This analysis proves that the proposed adjustment layer achieves adaptive feature refinement according to weather variability rather than performing fixed feature gating.

To evaluate the generalization capability of the proposed

AS–TR–GRU–Transformer model, the experiments are conducted on different years and regions using the historical weather dataset. The model was trained on data from 2006–2018 and tested on 2019–2020 from a geographically different region. The results are given in Table 5.

The AS–TR–GRU–Transformer model maintains high predictive accuracy under both cross-year and cross-region conditions. Also, the slight increases in MSE and MAE indicate minor variability due to unseen climatic patterns. But the overall $R^2$ remains above 0.94. These results suggest the model is robust and can generalize well to different temporal and spatial conditions.

To verify the statistical significance of the performance improvement, a paired *t*-test was conducted between the proposed AS–TR–GRU–Transformer model and each baseline model (Table 6). The paired samples are generated from the individual prediction errors obtained on the same test dataset samples. Each model produced predictions for the identical set of test observations. The absolute error values of corresponding samples are paired before performing the statistical test. A total of 19,291 paired prediction samples are used for the t-test analysis. The null hypothesis assumes that there is no significant difference between the prediction errors of the proposed model and the compared models. The obtained *p*-values below 0.05 indicate that the reduction in prediction errors achieved by the proposed model is statistically significant.

**Table 6.** Comparative performance with statistical significance

| Model Name | MSE | MAE | RMSE | $R^2$ | *p*-Value (vs AS-TR-GRU) | Significance ($\alpha = 0.05$) |
|---|---|---|---|---|---|---|
| ARIMA | 4.85 | 1.92 | 2.20 | 0.71 | <0.001 | Significant |
| SVR | 3.42 | 1.41 | 1.85 | 0.81 | <0.001 | Significant |
| Random Forest | 2.76 | 1.18 | 1.66 | 0.87 | <0.001 | Significant |
| LSTM | 1.95 | 1.02 | 1.40 | 0.91 | 0.002 | Significant |
| GRU | 1.74 | 0.94 | 1.32 | 0.93 | 0.015 | Significant |
| Transformer | 1.48 | 0.88 | 1.22 | 0.94 | 0.042 | Significant |
| Proposed AS-TR-GRU | 1.22 | 0.79 | 1.11 | 0.954 | – | – |

Note: ARIMA = autoregressive integrated moving average; SVR = support vector regression; LSTM = long short-term memory; GRU = gated recurrent unit; MSE = mean squared error; RMSE = root mean squared error; MAE = mean absolute error; $R^2$ = coefficient of determination.

**Table 7.** Temporal block-based statistical validation of the proposed Adaptive Selection–Temporal Representation–Gated Recurrent Unit Transformer (AS–TR–GRU–Transformer) model

| Validation Block | Number of Test Samples | Mean Error Difference (Baseline – Proposed) | *t*-Value | *p*-Value | Statistical Significance |
|---|---|---|---|---|---|
| Block 1 (Early period) | 4,823 | 0.38 | 8.62 | <0.001 | Significant |
| Block 2 (Middle period) | 4,823 | 0.42 | 9.15 | <0.001 | Significant |
| Block 3 (Late period) | 4,823 | 0.36 | 7.94 | <0.001 | Significant |
| Block 4 (Recent period) | 4,822 | 0.40 | 8.76 | <0.001 | Significant |
| Overall Temporal Blocks | 19,291 | 0.39 | 8.61 | <0.001 | Significant |

The temporal block-based statistical validation of the proposed AS–TR–GRU–Transformer model is given in Table 7. The test samples are divided into sequential time blocks to reduce the influence of temporal dependency in weather data. The results show consistent error reduction of the proposed model across all temporal blocks with statistically significant improvements ($p < 0.001$). This analysis confirms that the performance improvement is not limited to a specific time period and remains stable under different weather patterns.

To investigate the possible information leakage caused by the Wind Chill feature, an additional experiment was conducted. It removed Wind Chill from the input feature set. The model was retrained using only directly measured meteorological variables. The results are given in Table 8. The removal of Wind Chill resulted in a slight reduction in prediction performance; however, the model still maintained strong forecasting capability. This indicates that the proposed model does not solely depend on the derived Wind Chill feature and can effectively learn complex relationships from independent weather variables.

The uncertainty estimation of the proposed model is given in Table 9. The proposed AS–TR–GRU–Transformer model achieves the highest prediction interval coverage probability (95.60%) with the lowest mean prediction interval width (2.15). The generated uncertainty bounds are both reliable and compact. The lower calibration error demonstrates that the Monte Carlo Dropout-based uncertainty estimation provides well-calibrated confidence intervals for weather prediction.

The uncertainty estimation module is not only used for performance evaluation but also provides confidence information associated with each weather prediction. The generated prediction intervals are used to identify reliable and uncertain forecasting periods, which is important for risk-aware agricultural decision-making. For example, narrow prediction intervals indicate high confidence in expected weather conditions. This supports farmers in planning irrigation and crop management activities with reduced risk. In contrast, wider uncertainty intervals indicate unpredictable weather conditions, where precautionary actions like delayed spraying and additional irrigation monitoring are needed. Therefore, the Monte Carlo Dropout-based uncertainty estimation improves the practical applicability of the proposed model by transforming deterministic forecasts into confidence-aware weather predictions.

The comparative analysis of the AS–TR–GRU–Transformer model with recent Transformer-based weather forecasting approaches is given in Table 10. The proposed model achieves lower prediction errors for short-term hourly weather forecasting. Models like FourCastNet, Pangu-Weather, and GraphCast are mainly designed for global-scale weather forecasting with different spatial resolutions and prediction objectives. Therefore, these models are considered only as reference approaches, and the comparison is intended to highlight the effectiveness of the proposed AS–TR–GRU–

Transformer model for local short-term hourly weather forecasting.

**Table 8.** Performance analysis without wind chill feature

| Model Variant | MSE ↓ | RMSE ↓ | MAE ↓ | $R^2$ ↑ |
|---|---|---|---|---|
| AS–TR–GRU–Transformer (Without Wind Chill) | 1.34 | 1.16 | 0.84 | 0.942 |
| AS–TR–GRU–Transformer (With Wind Chill) | 1.12 | 1.06 | 0.69 | 0.954 |

Note: GRU = gated recurrent unit; MSE = mean squared error; RMSE = root mean squared error; MAE = mean absolute error; $R^2$ = coefficient of determination, AS–TR–GRU–Transformer = Adaptive Selection–Temporal Representation–Gated Recurrent Unit Transformer.

**Table 9.** Uncertainty estimation performance of the proposed AS–TR–GRU–Transformer model

| Model | PICP (%) | MPIW | MACE | Uncertainty Reliability |
|---|---|---|---|---|
| LSTM | 91.20 | 2.84 | 0.086 | Moderate |
| GRU | 92.45 | 2.61 | 0.074 | Good |
| Transformer | 93.10 | 2.48 | 0.069 | Good |
| Proposed AS–TR–GRU–Transformer | 95.60 | 2.15 | 0.041 | High |

Note: LSTM = long short-term memory; GRU = gated recurrent unit; PICP = prediction interval coverage probability; MPIW = mean prediction interval width; MACE = mean absolute calibration error, AS–TR–GRU–Transformer = Adaptive Selection–Temporal Representation–Gated Recurrent Unit Transformer.

**Table 10.** Comparative analysis with recent weather forecasting models

| Model | MSE ↓ | RMSE ↓ | MAE ↓ | $R^2$ ↑ |
|---|---|---|---|---|
| FourCastNet | 1.86 | 1.36 | 1.05 | 0.928 |
| Pangu-Weather | 1.72 | 1.31 | 0.98 | 0.935 |
| GraphCast | 1.65 | 1.28 | 0.94 | 0.941 |
| Transformer | 1.48 | 1.22 | 0.88 | 0.940 |
| GRU | 1.74 | 1.32 | 0.94 | 0.930 |
| Proposed AS–TR–GRU–Transformer | 1.22 | 1.11 | 0.79 | 0.954 |

Note: GRU = gated recurrent unit; MSE = mean squared error; RMSE = root mean squared error; MAE = mean absolute error; $R^2$ = coefficient of determination, AS–TR–GRU–Transformer = Adaptive Selection–Temporal Representation–Gated Recurrent Unit Transformer.

## 5. CONCLUSION

This work presented an advanced short-term weather prediction based on the proposed AS–TR–GRU–Transformer architecture, which is designed to address key limitations of existing data-driven forecasting models. By integrating Transformer-based attention with GRU networks and adaptive memory mechanisms, it captured both short-range fluctuations and long-range temporal dependencies in hourly weather data. Also, an RL-inspired adjustment layer is used for dynamic balancing. This work proposed an AS–TR–GRU–Transformer–based weather prediction model for short-term forecasting using hourly meteorological data. The model achieved high predictive accuracy, with an MSE of 1.22, MAE of 0.79, and an $R^2$ score of 0.954, demonstrating its strong ability to capture complex temporal patterns in weather data. The integration of Transformer attention and adaptive memory gates improved long-range dependency learning, while the RL-inspired adjustment layer enhanced adaptability under dynamic conditions. Uncertainty estimation using Monte Carlo Dropout further strengthened the model by providing confidence-aware predictions, which are essential for risk-sensitive applications such as agriculture. Overall, the proposed approach offers an efficient and reliable solution for short-term weather forecasting, with improved accuracy and robustness compared to existing models. Despite its strong performance, the model has certain limitations. The RL-inspired adjustment layer introduces extra computational complexity. It could affect deployment in resource-constrained environments. Future work will focus on exploring lightweight architectures for real-time applications.

## REFERENCES


[1] Bauer, P., Thorpe, A., Brunet, G. (2015). The quiet revolution of numerical weather prediction. Nature, 525: 47-55. https://doi.org/10.1038/nature14956

[2] Bhattacharyya, D., Banerjee, I. (2025). Synergizing IoT and machine learning in hybrid models for next-generation weather forecasting. Journal of Emerging Trends in Computer Science and Applications, 1(1): 55-66. https://doi.org/10.65525/jetcsa.v1i1.4

[3] Balu, S., Raghabu, S.M.A., Mahalingam, P., Krishnasamy, J., Kadambarajan, J.P., Pattar, D.S. (2025). Advanced optimized quantum learning for robust water quality assessment with an IoT sensor data. Ingénierie des Systèmes d'Information, 30(2): 359-366. https://doi.org/10.18280/isi.300207

[4] Dibike, Y.B., Coulibaly, P. (2006). Temporal neural networks for downscaling climate variability and extremes. Neural Networks, 19: 135-144. https://doi.org/10.1016/j.neunet.2006.01.003

[5] Zhuang, H., Zhang, Z., Cheng, F., et al. (2024). Integrating data assimilation, crop model, and machine learning for winter wheat yield forecasting in the North China Plain. Agricultural and Forest Meteorology, 347: 109909. https://doi.org/10.1016/j.agrformet.2024.109909

[6] Hasani, Z., Peschel, J., Youngs, C., Fondaj, J., Vos, R. (2025). Advanced model for predicting weather conditions for smart grape cultivation: A comparative study between Kosovo and Iowa. International Journal of Interactive Mobile Technologies, 19(16): 108-132. https://doi.org/10.3991/ijim.v19i16.54923

[7] Hewamalage, H., Ackermann, K., Bergmeir, C. (2023). Forecast evaluation for data scientists: Common pitfalls and best practices. Data Mining and Knowledge Discovery, 37: 788-832. https://doi.org/10.1007/s10618-022-00894-5

[8] Price, I., Sanchez-Gonzalez, A., Alet, F., et al. (2025). Probabilistic weather forecasting with machine learning. Nature, 637: 84-90. https://doi.org/10.1038/s41586-024-08252-9

[9] Du, P. (2019). Ensemble machine learning-based wind forecasting to combine NWP output with data from weather station. IEEE Transactions on Sustainable Energy, 10(4): 2133-2141. https://doi.org/10.1109/TSTE.2018.2880615

[10] Hachimi, C.E., Belaqziz, S., Khabba, S., et al. (2024). Advancements in weather forecasting for precision agriculture: From statistical modeling to transformer-based architectures. Stochastic Environmental Research

and Risk Assessment, 38: 3695-3717. https://doi.org/10.1007/s00477-024-02778-0

[11] Varshney, Y., Ram, N. (2024). Weather forecasting: A systematic review using AI approaches. International Journal of Emerging Technologies and Innovative Research, 4(10): 178-196.

[12] Mass, C. (2023). The uncoordinated giant II: Why U.S. operational numerical weather prediction is still lagging and how to fix it. Bulletin of the American Meteorological Society, 104: E851-E871. https://doi.org/10.1175/BAMS-D-22-0037.1

[13] Kişmiroğlu, C., Isik, O. (2025). Temperature prediction using Transformer–LSTM deep learning models and SARIMAX from a signal processing perspective. Applied Sciences, 15: 9372. https://doi.org/10.3390/app15179372

[14] Yuan, S., Wang, G., Mu, B., Zhou, F. (2025). TianXing: A linear complexity Transformer model with explicit attention decay for global weather forecasting. Advances in Atmospheric Sciences, 42: 9-25. https://doi.org/10.1007/s00376-024-3313-9

[15] Yasavoli, B., Habibirad, A., Javanshiri, Z. (2025). A hybrid deep learning model in predicting weather temperature. Earth Science Informatics, 18: 461. https://doi.org/10.1007/s12145-025-01963-1

[16] Abdulla, N., Demirci, M., Ozdemir, S. (2022). Design and evaluation of adaptive deep learning models for weather forecasting. Engineering Applications of Artificial Intelligence, 116: 105440. https://doi.org/10.1016/j.engappai.2022.105440

[17] Akilan, T., Baalamurugan, K.M. (2024). Automated weather forecasting and field monitoring using GRU-CNN model along with IoT to support precision agriculture. Expert Systems with Applications, 249: 123468. https://doi.org/10.1016/j.eswa.2024.123468

[18] Kim, B., Kim, T. (2025). AI in extreme weather events prediction and response: A systematic topic-model review (2015–2024). Frontiers in Environmental Science, 13: 1659344. https://doi.org/10.3389/fenvs.2025.1659344

[19] Liu, J.M., Chang, F.L., Yang, J.H., et al. (2025). Deep reinforcement learning for irrigation optimization: Advantages, opportunities, and challenges. Agricultural Water Management, 322: 110030. https://doi.org/10.1016/j.agwat.2025.110030

[20] Tan, H.Y., Zhang, Q., Li, M.X., Liu, X.X., Hu, L. (2026). Dynamic resource allocation in smart agricultural IoT using reinforcement learning and 6G edge computing. Journal on Wireless Communications and Networking, 23. https://doi.org/10.1186/s13638-025-02563-9