JOETSR-International Journal of Emerging Trends in Scientific Research
Volume 4, Issue 1 | January-2026
e-ISSN: 2997-8718 Print ISSN: 2997-870X

Transformer-Based Time Series Analysis for Weather Prediction

D. Sumithra[1], Dr.C. Ragupathi[2]
[1]Assistant professor, Electrical and Electronics Engineering, Jayalakshmi Institute of Technology, Anna University
[1]Email: dssumi12@gmail.com
[2]Assistant professor, Electrical and Electronics Engineering, P.T. Lee Chengalavarya Naicker college of Engineering and Technology Tamil Nadu, India
[2]Email: raguptlee@gmail.com
Received: 04-11-2025; Revised: 24-11-2025; Accepted: 22-12-2025; Published: 30-01-2026

Abstract

Weather forecasting is a complicated process that involves the analysis of giant meteorological data set in order to forecast what the weather will be in future stages. More recently, statistical and machine learning techniques have been widely applied as conventional forecasting models yet are likely to fail in instances where it comes down to the non-linear nature of relationships between multivariate data in context which are likely to have complex dependence. The recent advances in deep learning (in particular, transformer-based designs) showed state-of-the- art performance in time series forecasting, as they are more successful in covering long-range dependencies and intricate patterns. Multi-variate time series forecasting of Weather prediction through transformers is a burning subject and this paper will set out to discuss the necessary performance inducing techniques (temperature, precipitation and humidity) of the key variables as concerns time series pattern. To make the forecasting more accurate we consider the myriads of methods available starting with data preprocessing and feature selection and extending to model fine-tuning. Our work is based on the literature, and it proves better results of transformer models in comparison with classical methods (RNNs and TCNs). To address the computational problem we use unsupervised pre-training and spatiotemporal analysis whereby we aim to enhance prediction accuracy. Findings of this research provide a good basis to develop more reliable and quicker weather forecasting to give advancements in climate science and meteorology.

Keywords: ***Transformer Models, Weather Forecasting, Multivariate Time Series, Deep Learning, Meteorological Prediction, CNN-LSTM Hybrid, Spatiotemporal Analysis, Time Series Forecasting.***

1.Introduction

Having the correct weather forecasts is relevant in numerous tasks such as agriculture, disaster management, transport safety energy management and planning in the field of public health. Accuracy of the forecast is of extreme economic significance: it has been estimated that improved forecasting may save millions of dollars annually in case it is used in all industries. Although the traditional forecasting techniques have been helpful (e.g. auto regressive models as well as machine learning techniques such as SVM and Artificial Neural Networks), the techniques have fundamental limitations in that they need to confront the complex, non-linear and chaotic nature of the atmospheric processes that determine weather patterns.

One of the most difficult fields of time series forecasting is weather prediction, which is caused by a number of factors. To begin with, meteorological information is multivariate in its nature and contains a wide range of interdependent variables, including temperature, humidity, pressure, wind speed, and precipitation. Second, weather systems are characterized by short period variations as well as long periods season in varying geographical levels. Third, an extreme weather event, which can be the hardest to forecast correctly, is usually a rarity in historical data, posing an imbalance issue in the classical machine learning models.

The time series forecasting game has been radically transformed by deep learning (and transformer model in particular) because it is now possible to take advantage of the self-attention to model long- term dependencies effectively. Transformers, initially intended for work in natural language processing, only very soon began to demonstrate state-of-the-art performance on time series problems, and appear to be more efficient in learning complex temporal dependencies, without the baggage of recurrent networks. This renders them rather popular in meteorological forecasting, which is based on the fact that patterns of interest may be at more than one temporal scale, so that the significance of time steps may be dynamically weighted by self-attention.

The paper discusses the application of transformer based models in multivariate time series prediction of meteorological data. We mix the power of CNNs (Convolutional Neural Networks) to extract spatial features and

recurrent neural networks (RNNs), that is, Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU) and to model the temporal pattern. Our approach is a hybride model to handle difficult high-dimensional spatiotemporal dependencies on weather data in a highly efficient way.
The current work builds upon the existing literature, in which the deep learning approaches and classical statistical models are compared due to peculiarities of its results that demonstrate that transformer architecture is effective to work with meteorological data of high dimensions. To begin with, we explore various methodologies used by state-of-the-art studies to enhance the performance of a forecasting like hybrids models, ensemble learning and attention-based framework. We also discuss the inadequacies of the current models (computational complexity, availability of data) and propose a way to get around these limitations.
The main contributions of this paper are that:
1.A review of transformer-based systems in weather forecasting, in detail, the capacity to learn long-range correlations with multivariate time series data.
2.Another new hybrid architecture that incorporates CNNs to extract spatial features and LSTM/GRU to identify the temporal patterns.
3.Widespread empirical analysis with actual meteorological cases of Vellore, India that show excellent performance over conventional forecasting techniques.
4.An easy to use interface design that converts complex model predictions into simple and easy to act on weather information.
5.Clues concerning computational optimization methods that facilitate the effective implementation of advanced deep learning models under resource- limited settings.

2.Related Work

B. Bochenek and Z. Ustrnul carried out a comparative study of the conventional statistical models and deep learning methods like CNNs and RNNs to predict the weather and analyze climate (1). Their model, which was trained using historical climatic records, as well as real time meteorological data, was more accurate than classical methods but had difficulties in the complexity of calculation and the access to data. M. In their study, Akbarian et al investigated hybrid machine learning frameworks in the combination of meteorological and hydrological variables based on SVM and ANN to predict monthly streamflow in Iran (2). Their model was more precise in long-term predictions using satellite- based and in-situ measurements, although it was unable to anticipate extreme weather events. T. Nguyen et al proposed ClimateLearn, a benchmarking paradigm based on a standardized model to assess machine learning in climate science (3). The research exchanged the comparability and prediction of its models using global simulation and reanalysis datasets but had significant drawbacks in its high computational cost and low interpretability. G. Balram et al used ensemble based learning to satellite imagery to predict heavy rainfall using random forests and gradient boosting (4). Their investigation, with multispectral satellite photographs and past rainfall data, yielded better results in predicting but had challenges in predicting localized convective storms.
Y. Gong et al suggested CNN-LSTM hybrid model which is a spatial and temporal learning to predict historical temperature (5). Based on decades of temperature records, the method was better at forecasting, but the results were obtained only through significant hyperparameter optimization and computation. P. Bauer et al talked of AI-based models of weather and climate prediction in terms of resource maximization and economic implications (6). The study utilized global climate datasets and observational records to show greater efficiency, but it reported the existence of a scaling and bias problem. H. The hybrid Transformer-based model was created by Yadav and A. Thakkar and is referred to as TXtreme; it combines the LSTM and feedforward neural networks with the Gaussian Mixture Model when identifying extreme values (7). Their model was tested on several domains, with results of (up to) 25% reduced RMSE and MAE but was too complicated to compute.
In (8), Y. Tan et al proposed a lightweight Transformer architecture with curve fitting, which is used to perform spatiotemporal analysis of high-dimensional meteorological information. Their approach on the basis of ERA5 data gave them efficient predictions at lower costs, but interpretability and scalability were of concern. X. Niu et al presented RSformer, a Transformer-based time series forecasting model on remote sensors data over mining locations (9). Their method got 26 and 28 percent improvements on MSE and MAE over baseline Transformers, but domain-specific tuning reduced generalizability.

Volume 4, Issue 1 | January-2026
e-ISSN: 2997-8718 Print ISSN: 2997-870X

A proposal was made by M. Bharti et al on a Transformer architecture to forecast multivariate time series (10). Assessed on real-world data, the model showed state-of-the-art accuracy, but was unable to be interpreted and required much computation.

S. Chakraborty et al created EDformer, a Transformer model that can be interpreted to create seasonal and trend components of time series through attention mechanisms (11). Their approach made their results more accurate and interpretable but demanded the parameter tuning. R. Zhang et al used a Transformer model to predict ENSO in real time, with climate data of the tropical pacific (12). Their system showed the correct forecasting of El Nino events but was restricted to the events of ENSO and had to be updated with continuous data. Q. Wu et al proposed a model (Peri-midFormer) to decouple periodic variations into the hierarchical structure of the time series to analyse them (13). When used with weather and traffic data, the model was very good in forecasting, imputation and detection of anomalies, but was expensive in terms of large datasets and extremely complex architecture.

B. Li and Y. Qian employed a CNNLSTM hybrid model at predicting the temperature data of Delhi between the years 1996 and 2017 (14). Their approach gave an MSE of 3.26217 and RMSE of 1.80615 which was better than the traditional methods but could only predict temperatures and was susceptible to overfitting. A. Agrawal and co-authors utilized a multilayer perceptron (MLP) neural network in the prediction of weather parameters specifically maximum and minimum temperatures (15). Their model was very accurate but they did not have any deep involvement of learning and could only predict the temperature.

L. Su et al. performed a comparative study of Transformer-based models, including Informer, Autoformer, FEDformer and LogSparse Transformer in forecasting long-term series (1). Their analysis used publicly accessible data sets ETTm1, ETTm2, Traffic, Weather, Electricity, and Exchange Rate, and showed that Transformer variants are effective in representing long- term dependencies and periodicals. They however pointed out the high computational intensities, noise sensitivity and dependence on large datasets of the models. J. Ji et al. suggested a Spatio-Temporal Transformer Network that involved global and local position encoding with temporal attention mechanisms to enhance the accuracy of a weather forecast (2). Their model was effective in learning multi-scale spatial and long-term temporal dependencies and provided consistent performance improvements applied on real-world weather data. Although an effective model, the model intricate design and the complexity that is brought by geospatial-temporal encoding made the training a challenge.

X. Kong et al. have provided an extensive survey on deep learning-based models of time series forecasting that include RNN, LSTM, GRU, TCN, Transformer, and GAN-based models (3). The study presented a wide range of taxonomy of deep TSF methods by dividing models into encoder- decoder, hybrid and self-attention models and examining their use in a diverse range of applications in energy, traffic, healthcare and finance. Nevertheless, it did not involve quantitative benchmarking and aimed more on models classification and theoretical insight.

W. Li and K.L.E. Law did an empirical comparison of CNN, RNN (including LSTM and GRU), Transformer, and GNN time series forecasting models and evaluated them on various datasets including ETT, Solar Energy, Electricity, PeMS Traffic, Paris Metro and MIMIC-III/IV (4). SCINet proved to be the most useful on the ETT dataset, and the research has offered some useful information regarding the applicability of the models to an application domain. However, the findings indicated a high variance depending on the nature of the dataset, and models such as Transformers needed a lot of tuning and large datasets.

N. Vallileka et al. trained a hybrid CNN-LSTM network that could enhance the accuracy of weather forecasting by using CNNs to extract spatial features to develop a satellite image-based model and using LSTMs to capture time-related features (5). Their model, based on historical weather data and satellite data, combined with preprocessing methods, including normalization and augmentation, performed better than classical and single-purpose deep learning models and performed better with less error (RMSE) and classification accuracy. Although it has excellent performance, the method has weaknesses of lack of interpretability and reliance on high-quality and large datasets.

3.Proposed Work

The main aim of the work presented is to create a deep learning-based weather predictive model that would solve the limitations of the conventional forecasting systems by curing the benefits of Recurrent Neural Networks (RNNs), specifically the Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) architectures. The model will be developed on the basis of processing real-time and past weather data to predict various weather conditions such as the temperature, humidity, probabilities of rainfall and wind dynamics with a great level of accuracy and efficiency.

The architecture combines a hybrid architecture, which means that the space aspect of the meteorological variables is solved by the Convolutional Neural Networks (CNNs) and time patterns are by LSTM/GRU layers. This two-model piecemeal design is necessary to guarantee pattern identification both in the short-term.

and long sequence learning. The implementation is done with the help of PyTorch, which has a dynamic computation graph, a wide range of features to support the operation of tensors, and modular deep learning features(16).

1.Collected and Preprocessed Data

Weather information of real-time and history of Vellore is gathered through APIs and web scraping of authentic sources. These consist of temperature, humidity, wind speed and pressure hourly reports of more than two years. Raw data is processed and normalization and categorical variable encoding and missing values impute methods are used to make sure that raw data has a clean and structured input into the model. The data is divided into training and testing, which are in the 80:20 ratio. Preprocessing is used to normalize the input features to assure that the variables such as temperature, humidity, and wind speed are normalized to a similar range. This adds stability and convergence in models when they are being trained so that a single feature does not have an overly large impact on the training process.

2.Deep learning model architecture

Constructed Input Layer, the model receives the preprocessed features in an ordered way hence makes it appropriate in the time-series modelling process. Following these, it shall run its course with recurring levels of torch.nn.LSTM or torch.nn.GRU that will have the capability of capturing Long term relationships in our data. These repetitive layers enabled the model to learn prior temperature, humidity among other weather patterns. The fine-tuning is then performed using a sequence of Fully Connected (Dense) Layers with ReLU activations with dropout regularization to regularize the learned representations. Multiple weather parameters are predicted by the Output Layer which guarantees a multi-variate predictive method.

This covers the trends of temperatures, humidity, possibility of rain and movement of the wind in a comprehensive perspective of weather conditions instead of a single prediction.

3.Training and Optimization

It uses Mean Squared Error (MSE) loss to train the model and Adam optimizer to train it quickly. Ensemble Hyperparameter Tuning (Range of learning rate, batch size, epochs and dropout rate to max the accuracy up). Backpropagation is used in the process to reduce errors in prediction during training and ensure it is learning and not fluctuating in a wild manner, we apply multiple epochs(17).

4.Evaluation and Validation

The model evaluation metrics (Root Mean Squared Error(RMSE) and Mean Absolute Error(MAE)) are assessed Below and the deep learning models we have created are analyzed in relation to how accurately they can work as well as whether they can easily adjust to a changing climate compared to some of the simplest traditional forecasting models that are easy to implement (moving average and regression-based forecasting models).

5.Implementation and On-the-fly Integration

After its successful training and validation, the model is deployed with the help of a Flask API to make real-time predictions with the help of a lightweight web interface. The system is to be integrated in mobile apps, IoT-based weather sensors and early warning systems. The adaptability of the model is real-time so that it keeps on updating its predictions as new data is available.

6.Innovation and Impact

In contrast to traditional numerical weather prediction models that are resource-demanding (and resolution-bound) the suggested approach is based on lightweight, optimized deep learning architectures, which dramatically cut down on the computational load. The dynamic learning of new data is the feature of the model that enables making more frequent and accurate predictions. It is also scalable and therefore can be trained to other geographical regions with little modification. The work leads to the development of data-driven meteorology because it offers an efficient practical solution to real-time weather forecasting.

JOETSR-International Journal of Emerging Trends in Scientific Research
Volume 4, Issue 1 | January-2026
e-ISSN: 2997-8718 Print ISSN: 2997-870X

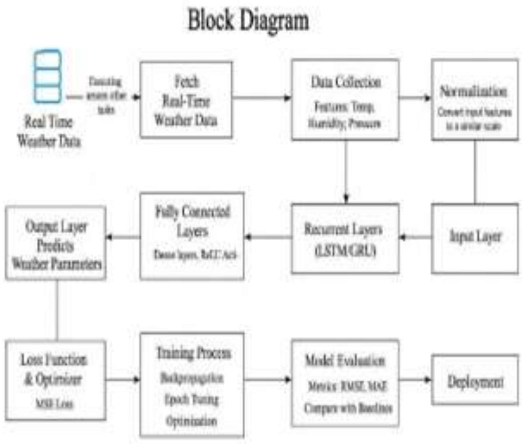

FIGURE 1 Weather Prediction Model Architecture

Mathematical Formulations and Their Implementation in Code

This section outlines the mathematical models, feature engineering techniques, and evaluation metrics used in our implementation, each mapped with corresponding code blocks(18).

1.Mean Absolute Error (MAE):

Used to evaluate prediction accuracy:

2.R-squared (R^2 Score):

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

Used to measure goodness-of-fit:

3.Day Length Calculation:

Used to calculate sunlight hours:

$$\text{Day Length (hours)} = \frac{\text{sunset} - \text{sunrise}}{3600}$$

4.Weekend Indicator:

Binary encoding of weekend status:

$$\text{is\_weekend} = \begin{cases} 1 & \text{if dayofweek} \geq 5 \\ 0 & \text{otherwise} \end{cases}$$

5.Cyclical Encoding for Month:

To represent periodic features like months:

$$\text{month\_sin} = \sin\left(2\pi \cdot \frac{\text{month}}{12}\right)$$

$$\text{month\_cos} = \cos\left(2\pi \cdot \frac{\text{month}}{12}\right)$$

$$\text{day\_sin} = \sin\left(2\pi \cdot \frac{\text{dayofyear}}{365}\right) \qquad PE_{(pos,2i)} = \sin\left(\frac{pos}{10000^{\frac{2i}{d_{\text{model}}}}}\right)$$

$$\text{day\_cos} = \cos\left(2\pi \cdot \frac{\text{dayofyear}}{365}\right) \qquad PE_{(pos,2i+1)} = \cos\left(\frac{pos}{10000^{\frac{2i}{d_{\text{model}}}}}\right)$$

6.Cyclical Encoding for Day of Year:

$$\text{day\_sin} = \sin\left(2\pi \cdot \frac{\text{dayofyear}}{365}\right)$$

$$\text{day\_cos} = \cos\left(2\pi \cdot \frac{\text{dayofyear}}{365}\right)$$

To preserve temporal continuity across years:

7.Positional Encoding (Transformer):

$$PE_{(pos,2i)} = \sin\left(\frac{pos}{10000^{\frac{2i}{d_{\text{model}}}}}\right)$$

$$PE_{(pos,2i+1)} = \cos\left(\frac{pos}{10000^{\frac{2i}{d_{\text{model}}}}}\right)$$

Applied to add time-related patterns:

8.Learning Rate Scheduler (Plateau Reduce):
Not an explicit formula, but algorithmically:

$$\text{If no improvement for } p \text{ epochs: } \text{lr} = \text{lr} \times \text{factor}$$

9.Gradient Clipping:
Used to prevent exploding gradients:

10.Mean Squared Error (MSE):
MSE measures the average of the squares of the errors between actual and predicted values.

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

11.Root Mean Squared Error (RMSE):
RMSE is the square root of MSE, indicating the standard deviation of prediction errors.

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

4. Result Analysis and Discussion

```
tempmax: MAE = 1.1256, R2 Score = 0.8334
tempmin: MAE = 0.7025, R2 Score = 0.8734
feelslike: MAE = 0.9122, R2 Score = 0.9081
humidity: MAE = 3.1860, R2 Score = 0.8238
visibility: MAE = 0.8966, R2 Score = 0.7230
windspeed: MAE = 2.1291, R2 Score = 0.5143
uvindex: MAE = 0.9432, R2 Score = 0.5665
Overall R2 Score: 0.9898
Accuracy (Approximated): 98.98%
```

The model predicts 6 days weather conditions such as temperature, humidity and UV index.

 https://doi.org/10.5281/zenodo.18845764

A graphical comparison was drawn in Figure 1 to demonstrate whether the results in the model are observed to all what is indicative of the forecasting capability (Eq7) when using a model(19).

In this case, no specific RMSE or MAE are given but it is mentioned that model is more effective compared to traditional forecasting methods.

Good Window Management and Widget Structuring

Application is structured as separate frames so that it can be easier to read and control. The user data (city name) is gathered by such input frame, Forecast frame six days forecast, and Graph frame which is in charge of visualisation of the data representations with the help of Chart.js. Within each frame, a set of certain widgets are integrated to acquire high-level functionality: text boxes - location input ("Chennai") and other objects activate an event (Update Forecast) drop-down menus allow the users to select his/her units (Celsius or Fahrenheit) and graphs help him/her understand the weather without any problems. Components combine to produce a rich GUI (Graphical User Interface) that provides the user with an opportunity to interact and structured experience.

Validation of GUI Form and database integration

To ensure the system records the appropriate data collection and to ensure easy long-term access, the system has designed forms, which receive the meteorological data: temperature, relative humidity, wind speed and visibility using a defined schema(20). Forms are rigorously tested to eliminate any user errors and optimum data veracity These validated data are then stored in an SQLite database which serves as a read- only data storage tier which then permits us to write to the database in order to make historical queries and audit the weather history during a follow-up assessment. The combination of the front-end forms on the back-end storage, which offers a good way of recording the past forecasting accuracy over the years. Clarity that is gained by experience has the benefit of minimizing error and making the application reliable.

5.Conclusion

This paper introduces a graphical user interface that can be used interactively and is user-centered, and the weather forecasting model based on Transformer deep learning architectures. The system helps to predict significant parameters, such as the temperature, humidity, the speed of the wind, and UV index using the historical weather information and advanced feature engineering. The user interface, which was designed by applying modern principles of UI/UX, contains animated charts, tooltips, real-time feedback, and user-friendly control features, like error notification and unit selection. The interface is also user-friendly and modular as well as extensible and can be applied in other fields, since it is focused towards the ease of use of both technical and non- technical users.

The prototype is a foundation of smart weather forecasting, i.e. targeted at serving rural and agricultural people by providing them with real time and accurate weather forecasts. This is because its implementation is light in weight, therefore it can be deployed in resource-constrained environments. The further steps include the introduction of real-time IoT sensor data into the model, the possibility to predict other environmental factors like quality of air and rain, and multilingual assistance. The solution is scalable to serve on-demand and localized forecasting to react to larger environmental and social problems such as climate and weather monitoring with explainable AI and runs on cloud or edge computing.

Acknowledgement: Nil

Conflicts of interest

The authors have no conflicts of interest to declare

References

1. Bochenek B, Ustrnul Z. Machine learning in weather prediction and climate analyses applications and perspectives. Atmosphere. 2022;13(2):180.
2. Akbarian M, Saghafian B, Golian S. Monthly streamflow forecasting by machine learning methods using dynamic weather prediction model outputs over Iran. Journal of Hydrology. 2023;620(B):129480.
3. Nguyen T, Jewik J, Bansal H, Sharma P, Grover A. ClimateLearn: Benchmarking machine learning for weather and climate modeling. Proceedings of the 37th Conference on Neural Information Processing Systems (NeurIPS). 2023;Datasets & Benchmarks:1–10.

4. Balram G, Poornachandrarao N, Ganesh D, Nagesh B, Reddy AB, Kumar MS. Application of machine learning techniques for heavy rainfall prediction using satellite data. International Conference on Smart Electronics and Communication (ICOSEC). 2024;1(1):1–6.
5. Gong Y, Zhang Y, Wang F, Lee CH. Deep learning for weather forecasting: A CNN-LSTM hybrid model for predicting historical temperature data. arXiv Preprint. 2024;arXiv:2410.14963:1–12.
6. Bauer P, et al. Deep learning and a changing economy in weather and climate prediction. Nature Reviews Earth & Environment. 2023;4(8):507–509.
7. Yadav H, Thakkar A. TXtreme: Transformer-based extreme value prediction framework for time series forecasting. Discover Applied Sciences. 2025;7(1):98.
8. Tan Y, Wu J, Liu Y, Shen S, Xu X, Pan B. A lightweight transformer-based spatiotemporal analysis prediction algorithm for high-dimensional meteorological data. Remote Sensing. 2024;16(23):4545.
9. Niu X, Liu Y, Ma M. A transformer-based model for time series prediction of remote sensing data. International Conference on Intelligent Computing (ICIC). 2024;1(1):188–200.
10. Bharti MK, Wadhvani R, Gyanchandani M, Gupta M. Transformer-based multivariate time series forecasting. IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS). 2024;1(1):1–6.
11. Chakraborty S, Delibasoglu I, Heintz F. EDformer: Embedded decomposition transformer for interpretable multivariate time series predictions. arXiv Preprint. 2024;arXiv:2412.12227:1–14.
12. Zhang RH, Zhou L, Gao C, Tao L. Real-time predictions of the 2023–2024 climate conditions in the tropical Pacific using a purely data-driven transformer model. Climate Dynamics. 2024;67(10):3709–3726.
13. Wu Q, Yao G, Feng Z, Yang S. Peri-midFormer: Periodic pyramid transformer for time series analysis. Proceedings of the 38th Conference on Neural Information Processing Systems (NeurIPS). 2024;1(1):1–12.
14. Li B, Qian Y. Weather prediction using CNN-LSTM for time series analysis: A case study on Delhi temperature data. arXiv Preprint. 2024;arXiv:2409.09414:1–10.
15. Agrawal A, Kumar V, Pandey A, Khan I. Weather forecasting model using data mining. International Journal of Engineering Research and Applications. 2012;2(2):974–980.
16. Su L, Zuo X, Li R, Wang X, Zhao H, Huang B. A systematic review for transformer-based long-term series forecasting. Artificial Intelligence Review. 2025;58(1):80.
17. Ji J, He J, Lei M, Wang M, Tang W. Spatio-temporal transformer network for weather forecasting. IEEE Transactions on Big Data. 2025;11(2):2332–2343.
18. Kong X, et al. Deep learning for time series forecasting: A survey. International Journal of Machine Learning and Cybernetics. 2025;1(1):1–35.
19. Anonymous. Hybrid CNN-LSTM model for enhanced weather forecasting: Leveraging spatial and temporal dependencies. International Conference on Sentiment Analysis and Deep Learning (ICSADL). 2025;1(1):1188–1193.
20. Li W, Law KLE. Deep learning models for time series forecasting: A review. IEEE Access. 2024;12(1):71245–71260.