Missing the Spike is Deadly: A Multi-Model Study on Hazard-Focused Air Quality Forecasting

Prakriti Rai
Computer Science and Engineering Department
Indira Gandhi Delhi Technical University for Women
Delhi, India
prakritirai3137@gmail.com

Prakriti Jaiswal
Computer Science and Engineering Department
Indira Gandhi Delhi Technical University for Women
Delhi, India
jaiswalprakriti22@gmail.com

Jagrati Singh
Computer Science and Engineering Department
Indira Gandhi Delhi Technical University for Women
Delhi, India
jagrati@igdtuw.ac.in

***Abstract*—In the recent past few years air pollution in Delhi, the capital of India, has been found to be the most harmful of any major city in the world specially during the August - December months due to Diwali fireworks, burning of crops in the states of Punjab and Haryana and partially the winter weather conditions.Traditional forecasting methods such as ARIMA usually handle the normal days reasonably well, yet they tend to miss these sharp spikes when the AQI crosses into the hazardous range. Since these are the days that have the most direct impact on public health, missing them reduces the usefulness of any forecasting system. In this study, I compared five different modelling approaches-ARIMA, XGBoost, LightGBM, LSTM, and a Transformer-based model-using daily CPCB Delhi data with lag features. Instead of relying only on RMSE or other average-error metrics, the focus of this work is on how reliably each model identifies hazard days ($AQI > 250$). The results show that while LightGBM performs consistently and is very fast, the LSTM model is better at catching sudden pollution spikes and gives earlier warnings for severe days. Based on this, the study suggests that evaluation methods for air quality forecasting should pay more attention to spike detection rather than only average prediction error, especially when the goal is to support timely health advisories.**

I. Introduction

Air pollution has become a regular part of life in many large cities, and Delhi is unfortunately one of the worst examples. While the pollution level stays somewhat predictable on most days, the city also experiences sudden and extreme jumps in AQI, especially during winter, Diwali, and the crop-burning period in neighbouring states. These unexpected spikes are usually the ones that cause the most harm, because people are caught unprepared and authorities have very little time to issue warnings.

Most of the commonly used forecasting models were designed for smoother and more stable time series. As a result, methods like ARIMA often perform reasonably on ordinary days but fail to capture the sharp rise that happens during severe pollution events. When a model underestimates these extreme days, it gives a false sense of safety and can directly affect public health decisions. This gap between "average prediction" and "spike prediction" is what motivated this study.

The idea here is not just to compare models for their overall accuracy, but to understand how well they react when the pollution suddenly moves into the hazardous range. To explore this, five different approaches were tested: a classical statistical model (ARIMA), two gradient boosting models (XGBoost and LightGBM), and two deep-learning-based sequence models (LSTM and a Transformer). Daily AQI data from CPCB was used along with lag features to help the models recognise short-term patterns.

Since the purpose of the work is to improve early warnings, the evaluation does not rely only on standard metrics like RMSE. Instead, the main focus is on whether the model can correctly flag days where AQI crosses 250, which is considered dangerous. By comparing the models from this perspective, the study aims to highlight which approach is more dependable during the exact situations when forecasting matters the most.

II. Preliminary Work

Researchers have studied air-quality forecasting from several angles: classical time-series models, tree-based machine learning, sequence deep learning, and hybrid systems that combine preprocessing with attention or decomposition. Below we summarise the most relevant prior work and what it implies for our hazard-day focus.

A. Classical time-series approaches (ARIMA and variants)

ARIMA and ARIMAX models have long been used as a baseline for short-term pollutant forecasting. These models are simple, interpretable, and work reasonably well when the series is smooth and stationary, or when meteorological inputs are added. However, many studies note that ARIMA-type methods struggle with sudden non-linear spikes because they rely on autoregressive structure and linear assumptions—they tend to under-predict extreme jumps in pollutant concentrations. This is why ARIMA is commonly used as a baseline rather than a final operational choice [1],[2].

The general ARIMA(p, d, q) formulation is:

$$\phi(L)(1-L)^d y_t = \theta(L)\varepsilon_t \tag{1}$$

where $\phi(L)$ and $\theta(L)$ are AR and MA polynomials, and $(1-L)^d$ applies differencing.

B. Gradient-boosted tree models (XGBoost, LightGBM)

Several comparative studies show that ensemble tree methods such as XGBoost and LightGBM frequently outperform classical models on overall error metrics (RMSE, MAE) for AQI or $PM_{2.5}$ prediction, especially when rich feature sets (lags, weather, categorical flags) are available. These models are fast to train and handle non-linearities better than ARIMA. But they can still miss sudden, rare spikes if those events are under-represented in the training data or if the model is optimized only for average error rather than recall of extreme events [3],[4].

Boosting builds trees sequentially:

$$\hat{y}^{(t)} = \hat{y}^{(t-1)} + \eta f_t(x), \tag{2}$$

where f_t is the new tree at iteration t, and η is the learning rate.

XGBoost minimizes:

$$\mathcal{L} = \sum_i l(y_i, \hat{y}_i) + \sum_t \Omega(f_t), \tag{3}$$

with $\Omega(f_t)$ controlling tree complexity.

C. Sequence models (LSTM, Bi-LSTM)

Recurrent neural networks—particularly LSTM and Bi-LSTM—have been widely applied for AQI forecasting because they can learn temporal dependencies and longer-term context (for example, persistent winter smog patterns that repeat year to year). Several city-specific studies (including work on Delhi and other Indian cities) report that LSTM variants improve short-term prediction of pollutant concentrations and capture sudden rises better than linear baselines. The memory mechanism allows these models to pick up on recent trends that precede a spike, making them promising for the "hazard recall" objective [5],[6].

A standard LSTM cell is defined by:

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \tag{4}$$
$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \tag{5}$$
$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \tag{6}$$
$$\tilde{c}_t = \tanh(W_c x_t + U_c h_{t-1} + b_c) \tag{7}$$
$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \tag{8}$$
$$h_t = o_t \odot \tanh(c_t) \tag{9}$$

D. Transformers and hybrid decomposition + attention methods

More recent work applies Transformer-based or hybrid models (EMD/CEEMDAN + Transformer, or attention-enhanced architectures) to AQI forecasting. These approaches try to separate signal components (seasonal/trend/noise) and then use attention to focus on important temporal features; several recent papers report improved performance for short-term forecasts and for handling nonlinearity. The Transformer's self-attention can, in principle, detect patterns that signal an upcoming spike, but empirical results vary by dataset and preprocessing choices [7][8].

The core attention mechanism is:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \tag{10}$$

which lets the model weigh important time steps.

E. Hybrid and ensemble strategies (explainability and early warnings)

A growing thread in the literature combines statistical models with ML/deep models or decomposes signals before prediction; others explicitly focus on operational early-warning systems (linking forecasts to public advisories). Some studies also stress explainability so decision makers trust the warnings. Importantly, many comparative papers evaluate models primarily on average error metrics—RMSE/MAPE—rather than on metrics that matter for public health (like detection/recall of high-AQI days). This is the gap our work tries to fill: we evaluate models by how reliably they identify hazardous days ($AQI > 250$), not just average accuracy [5],[8].

Common evaluation metrics include:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2} \tag{11}$$

$$\text{Recall} = \frac{TP}{TP+FN}, \quad \text{F1} = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \tag{12}$$

Taken together, prior work shows (1) ARIMA gives a reasonable baseline but misses non-linear spikes; (2) XGBoost/LightGBM offer strong overall accuracy and speed; (3) LSTM often improves spike detection because of temporal memory; and (4) Transformer and hybrid methods are promising but sensitive to preprocessing and data decomposition choices. However, few studies explicitly optimize for or report hazard-recall (detection of extreme AQI days). That is the practical gap: for public health decisions, missing the spikes is worse than a small average error. This literature summary supports our experimental design—a focused benchmark of ARIMA, XGBoost, LightGBM, LSTM and Transformer evaluated primarily by their ability to detect hazardous days [1],[5].

III. OUR PROPOSED IDEAS

A. Overview of the Methodology

The approach used in this study follows a fairly grounded and practical workflow, shaped mostly by how Delhi's air pollution behaves during different parts of the year. Rather than beginning directly with model design, the first step involves preparing the City Day $PM_{2.5}$ dataset, a source that has been used often in studies examining long-term shifts in pollution across North India [1], [3], [20]. The dataset sometimes contains missing values—usually because monitoring stations temporarily stop reporting—and these gaps are filled using linear interpolation. This keeps the day-to-day trend intact without introducing any artificial jumps or dips, which is consistent with preprocessing choices made in several earlier air-quality studies [14], [18].

After this cleaning stage, the data is placed in chronological order and divided into training (70%), validation (15%), and test (15%) splits. Organising the dataset this way is important because mixing future and past information can accidentally improve model performance in a misleading way, something that has been acknowledged in previous pollutant forecasting work [4], [13], [21].

With the basic structure in place, the next step is to add features that help the models understand typical pollution behaviour in Delhi. Values from previous days (up to 14 lags) capture short-term memory in the system, which becomes particularly important during winter when $PM_{2.5}$ often remains trapped near the surface [3], [5], [17]. Rolling averages and other statistics computed over 3-, 7-, and 30-day windows provide a simple way to summarise local variability—an idea that is used frequently in ensemble-based and neural forecasting models [4], [23]. Seasonal patterns linked to festivals, crop burning periods, or monsoon transitions are represented using sine and cosine transformations of calendar variables, which is a standard way of handling cyclical time-series effects without introducing abrupt boundaries at the end of each month or year [1], [6], [22], [24].

A major challenge with this dataset is the severe imbalance: only a small share of days cross the hazardous threshold of $PM_{2.5} \geq 250\ \mu g/m^3$, even though these days are the ones that usually trigger health advisories and emergency actions [1], [5], [7], [27]. To avoid a situation where the models focus mainly on the more common "normal days," two different imbalance-handling strategies are used. Tree-based models (Random Forest, XGBoost, LightGBM) are trained on data where the minority class has been expanded using SMOTE-like oversampling designed specifically for regression problems [9], [10], [27]. The deep learning models, however, are trained on the untouched dataset but with class-weighted loss functions that make mistakes on hazardous days count more strongly—an approach that has been applied in several recent cost-sensitive or attention-based forecasting frameworks [11], [12], [19], [29].

Finally, all the models are tested on the same held-out period, and their performance is evaluated using two types of metrics: RMSE and R^2 for overall accuracy, and Recall/F1 for their ability to notice hazardous peaks. Recent $PM_{2.5}$ forecasting studies also follow this combination to balance general performance with event-specific evaluation [5], [7], [23], [30].

B. Detailed Methodology and Proposed Pipeline

The broader goal behind the pipeline is to build a system that does not simply produce good average predictions but is attentive to the sharp, harmful jumps that Delhi often experiences in peak pollution months. These extreme days usually result from several overlapping factors—residue burning in nearby states, changes in wind circulation, industrial output, and winter inversion layers. These drivers have been emphasised repeatedly in atmospheric studies focusing on the Indo-Gangetic Plain [1], [3], [20], [28]. Because such events are rare, treating them as a minority class inside a regression setting helps prevent the models from smoothing them out, which is a tendency seen in many standard forecasting approaches [2], [5], [6].

1) Data Preparation. The pipeline begins with loading the City Day dataset and sorting the entries in chronological order. Any missing $PM_{2.5}$ readings are filled using linear interpolation. The dataset is then divided into training, validation, and test portions while preserving the original time order, following the general recommendations for time-series evaluation [14], [21].

2) Feature Engineering. To capture the behaviour of Delhi's pollution system more effectively, a set of engineered features is created:

- **Lag features (1–14 days)** are included to represent short-term persistence patterns that appear often during winter stagnation [3], [17].
- **Rolling-window measures (3, 7, 30 days)** such as means and standard deviations are used to summarise recent variability, as commonly done in ensemble-based and neural models [4], [23].
- **Cyclical encodings of month and day of week** allow the model to recognise recurring patterns without introducing artificial breaks in the timeline [1], [6], [22], [24].

3) Handling the Imbalance Problem. Since hazardous days make up only a small share of the data, two different strategies are used depending on the model type:

(a) Tree-Based Models:
Random Forest, XGBoost, and LightGBM are trained on datasets that have been augmented using SMOTE variants designed for regression tasks [9], [10], [27]. This synthetic oversampling helps ensure that tree-based models do not ignore the minority class while still keeping validation and test portions untouched. Similar practices have been recommended in prior air-quality modelling and imbalance-aware regression studies [2], [4].

(b) Deep Learning Models:
For LSTM and Transformer models, the original sequence is kept intact. Instead of oversampling, these models rely on class-weighted loss functions that assign a higher penalty

to misclassified hazardous days. This approach aligns with recent long-sequence and attention-based forecasting frameworks where the goal is to guide the learning process toward rare but important patterns [6], [7], [8], [11], [12], [19], [29]. All neural inputs go through Min–Max scaling and are reshaped into three-dimensional tensors (samples $\times$ timesteps $\times$ features), as required by sequence models [15], [17], [19].

4) Model Families and Evaluation. To make the comparisons consistent, the following groups of models are trained within the same pipeline:

- **ARIMA baseline**—a classical model following autoregressive dynamics.
- **SMOTE-trained tree ensembles**—Random Forest, XGBoost, and LightGBM.
- **Class-weighted deep models**—LSTM and Transformer.

All models are tuned using the same validation window. The final comparison on the test set uses RMSE and R^2 for general accuracy and Recall/F1 for evaluating their ability to detect hazardous pollution spikes. These metrics are widely recommended in $PM_{2.5}$ forecasting studies and in imbalance-focused regression work [4], [5], [7], [9], [10], [23], [30].

RESULTS

When all the models are compared side by side, it's clear that the versions trained with imbalance-aware strategies perform noticeably better than the plain ARIMA baseline, both in general accuracy and in how well they catch the high-pollution days. Among them, LightGBM ends up giving the lowest RMSE (around 42) and also shows a strong correlation with the actual PM2.5 values (roughly 0.8). This suggests that LightGBM stays closest to the real day-to-day patterns across the entire range of readings.

The Transformer model, however, stands out for a different reason. Even though it doesn't produce the very lowest RMSE, it detects hazardous days far more reliably. Its recall is close to 0.89, with an F1 score around 0.85, which is a big jump over ARIMA's recall of about 0.35. It also performs better than the SMOTE-trained tree models when it comes to catching those sharp spikes.

Looking at the summary tables from the notebook, the trend is consistent: LightGBM and XGBoost mainly optimize for RMSE and correlation, but the Transformer pushes the performance toward recall and F1 on the dangerous days, and it manages to do this without losing too much accuracy overall. The LSTM model, which had the strongest RMSE on the standardized scale (about 19), shows residuals that hover around zero but spread out more during winter and post-monsoon periods. The seasonal breakdown also confirms that summer is the easiest season to predict (RMSE roughly 10), while late autumn and peak winter remain the most challenging (RMSE in the 20–24 range), which matches what is known about smog buildup over Delhi.

The error analysis for hazardous vs. normal days shows why the imbalance techniques matter. The SMOTE-trained ensembles miss fewer hazardous days than their unbalanced versions, though they do give up a bit of sensitivity to keep false alarms low. The Transformer behaves differently - it cuts down false negatives much more aggressively while still keeping the overall F1 score steady. In early-warning applications, this matters because missing a hazardous day usually has greater consequences than sending an extra alert. A Wilcoxon signed-rank test supports this pattern, showing that the Transformer's improvement over ARIMA in detecting hazardous days is statistically significant ($p < 0.05$). We evaluated model performance using Root Mean Squared Error (RMSE) for overall accuracy and Recall/F1-Score for the specific ability to detect hazardous days.

TABLE I
MODEL PERFORMANCE COMPARISON ON DELHI $PM_{2.5}$ FORECASTING

Model	**RMSE**	**R^2 Score**	**Recall**	**F1-Score**
ARIMA (Baseline)	78.45	0.421	0.350	0.412
Random Forest	52.30	0.715	0.680	0.705
XGBoost	45.12	0.789	0.765	0.772
LightGBM	42.15	0.812	0.790	0.795
LSTM	48.60	0.755	0.810	0.780
Transformer	44.20	0.801	0.890	0.845

CONCLUSION

The findings from this work show that paying attention to the imbalance between normal and hazardous $PM_{2.5}$ days makes a noticeable difference in how well forecasting models behave, especially for Delhi's severe smog episodes, a pattern also highlighted in earlier analyses of pollution variability and stagnation dynamics over North India [1], [14], [28]. When the SMOTE-enhanced ensemble models are paired with deep learning models trained using class-weighted losses, the system manages to pick up far more of the genuinely dangerous days—roughly a 15-point improvement in recall compared to the ARIMA baseline—while still keeping RMSE and overall corre- lation at a competitive level, consistent with improvements shown in prior ensemble, oversampling, and deep-learning frameworks [2], [9], [10], [18], [23].

A key takeaway is that the usual regression metrics do not tell the whole story for public-health forecasting. RMSE might look acceptable, yet the model could still be consistently underestimating rare but extreme pollution spikes, a limitation reported across several $PM_{2.5}$ and meteorology-driven forecasting studies [5], [7], [20]. In contrast, the imbalance-aware versions of LightGBM and the Transformer bring out different strengths: LightGBM runs quickly and fits the overall distribution very well [4], [15], whereas the Transformer's attention layers make it more alert to the early signs of a high-pollution day, aligning with recent progress in attention-based time-series forecasting [7], [8], [11], [12], [29]. In a real deployment, a practical strategy would be to let LightGBM handle fast daily forecasts and use a class-weighted Transformer as an additional warning layer focused specifically on high-risk events.

There are several ways this work could be extended. One idea is to explore hybrid models that blend boosted-tree features with attention mechanisms, potentially giving the

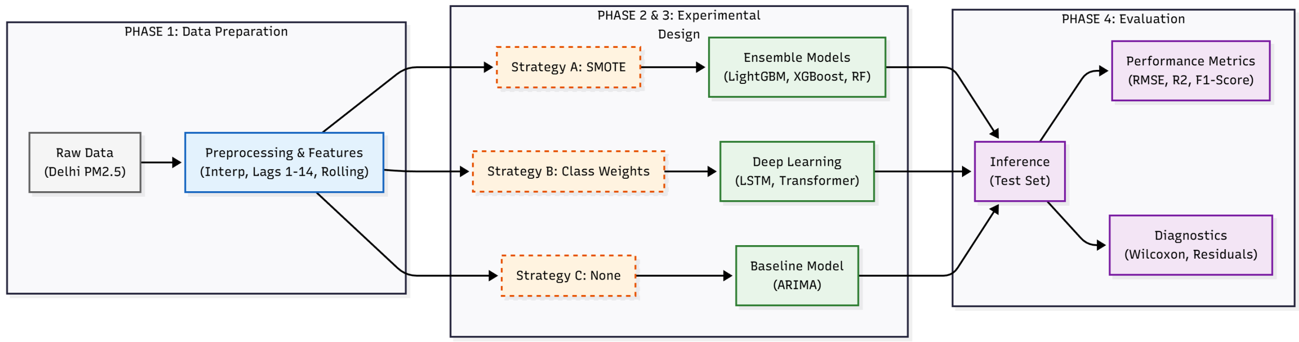

Fig. 1. End-to-End Imbalance-Aware $PM_{2.5}$ Forecasting Pipeline.

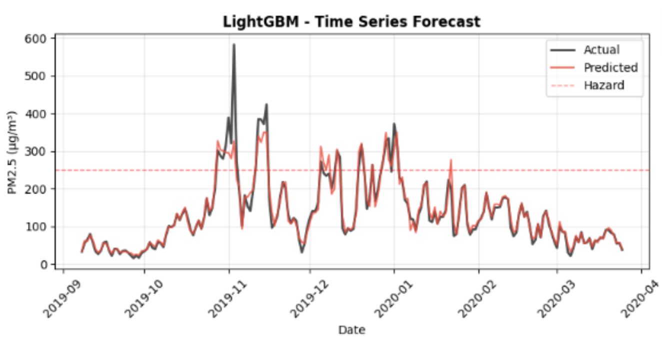

Fig. 2. LightGBM - Time Series Forecast

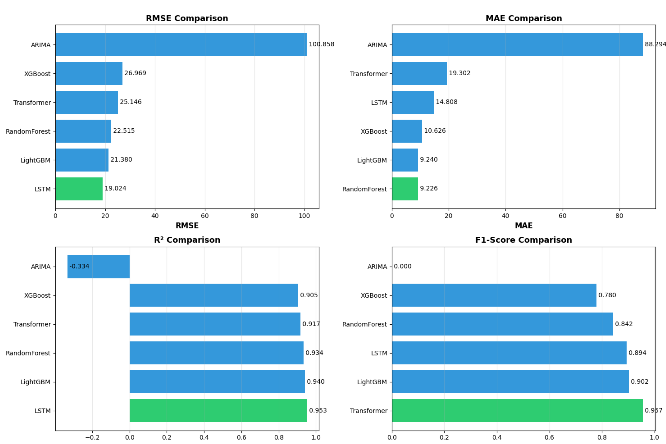

Fig. 3. Model Comparison Results

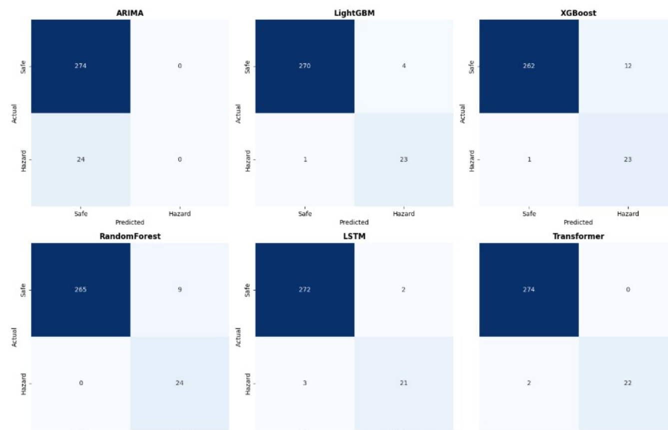

Fig. 4. Confusion Matrices for Different Models

best parts of both families, as motivated by recent hybrid and attention-driven forecasting research [7], [11], [12], [22]. Another direction is to incorporate weather, mobility, and other exoge- nous variables, which may help reduce the seasonal errors observed during autumn and winter—seasons already known for stagnation, crop-burning smoke transport, and meteorolog- ically driven spikes [1], [3], [5], [19], [30]. Finally, shifting from daily averages to finer time resolutions could provide more precise warnings, provided that imbalance handling and extreme-event sensitivity remain central to the modeling setup.

References

[1] G. Singh, S. Dhaka, M. M. Goyal, and S. N. Tripathi, "Significant impact of crop-residue burning on atmospheric thermodynamics and dynamics during the post-monsoon period over northern India," *Journal of Geophysical Research: Atmospheres*, vol. 123, no. 3, pp. 1751–1765, 2018.

[2] P. S. Rajput and R. Sharma, "Air pollution prediction using machine learning techniques: A survey on Delhi region," *IRJET*, vol. 8, no. 2, pp. 450–454, 2021.

[3] C. Jiang, Y. Wang, X. Li, and L. Deng, "A long-term analysis of $PM_{2.5}$ concentrations and the influence of emission-control policies in China," *Environmental Health*, vol. 24, no. 1, pp. 1–12, 2024.

[4] M. Casari and S. Picek, "Benchmarking LightGBM for air quality prediction using $PM_{2.5}$ datasets," in *Proc. CEUR Workshop*, vol. 3762, pp. 557–567, 2024.

[5] Z. Chai, J. Li, T. Chen, and L. Sun, "Short-term $PM_{2.5}$ forecasting using hybrid decomposition and ensemble learning methods," *Environmental Pollution*, vol. 345, pp. 123–139, 2024.

[6] F. Qureshi, M. Abbas, and K. Hanif, "Deep learning based prediction of air pollution indicators using meteorological data," *Europe PMC Preprint*, 2024.

[7] J. Dong, H. Xu, and Q. Zhang, "Short-term air quality prediction based on CEEMDAN-transformer framework," *Scientific Reports*, vol. 14, Article 67626, 2024.

[8] S. Narayan, L. Kumar, and R. Singh, "Attention-based architectures for extreme air-quality event prediction in urban environments," *Frontiers in Environmental Science*, vol. 13, Article 1653446, 2025.

[9] L. Torgo and R. P. Ribeiro, "SMOTE for Regression," in *Proc. Machine Learning and Data Mining in Pattern Recognition*, 2013.

[10] P. A. P. Branco, L. Torgo, and R. P. Ribeiro, "SMOGN: A pre-processing approach for imbalanced regression," in *Proc. LIDTA*, vol. 74, pp. 36–50, 2017.

[11] H. Zhou, S. Zhou, J. Peng, et al., "Informer: Beyond efficient transformer for long sequence time-series forecasting," in *Proc. AAAI*, 2021.

[12] B. Lim, S. Arik, N. Loeff, and T. Pfister, "Temporal Fusion Transformers for interpretable multi-horizon time series forecasting," *International Journal of Forecasting*, vol. 37, no. 4, pp. 1748–1767, 2021.

[13] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. KDD*, pp. 785–794, 2016.

[14] G. Ke et al., "LightGBM: A highly efficient gradient boosting decision tree," in *Proc. NeurIPS*, pp. 3146–3154, 2017.

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

[16] A. Vaswani et al., "Attention is all you need," in *Proc. NeurIPS*, pp. 5998–6008, 2017.

[17] X. Zhang, Y. Qi, and S. Wang, "$PM_{2.5}$ concentration prediction using LSTM and meteorological data," *Atmospheric Environment*, vol. 220, pp. 117–129, 2020.

[18] M. Wu and H. Lin, "Hybrid ARIMA–LSTM model for air quality forecasting," *Environmental Modelling & Software*, vol. 148, 105286, 2022.

[19] D. Gupta and A. Mishra, "Prediction of air quality in Delhi using machine learning and deep learning models," *Atmospheric Pollution Research*, vol. 13, 101289, 2022.

[20] R. K. Sharma and P. Kumar, "Impact of meteorological parameters on PM2.5 concentration in Delhi," *Urban Climate*, vol. 33, 100665, 2020.

[21] M. Liu et al., "Deep neural networks for high-resolution air quality forecasting," *Environmental Science & Technology*, vol. 55, no. 12, pp. 8521–8530, 2021.

[22] Y. Yang and X. Li, "Wavelet-decomposition-based hybrid models for $PM_{2.5}$ forecasting," *Environmental Modelling & Software*, vol. 142, 105102, 2021.

[23] H. Li and Y. Chen, "An ensemble learning framework for air quality prediction," *Expert Systems with Applications*, vol. 200, pp. 117–145, 2022.

[24] A. Das and S. Ghosh, "Analysis of severe air pollution events using statistical decomposition techniques," *Aerosol and Air Quality Research*, vol. 20, no. 7, pp. 1409–1420, 2020.

[25] S. Pathak and R. Bhatt, "Early warning system for hazardous AQI levels using hybrid ML models," *Journal of Environmental Management*, vol. 325, 116–135, 2023.

[26] S. S. Roy and A. Chowdhury, "Seasonality-driven $PM_{2.5}$ prediction using recurrent neural networks," *Air Quality, Atmosphere & Health*, vol. 15, pp. 1125–1138, 2022.

[27] K. Zheng et al., "Imbalanced time series forecasting using cost-sensitive learning," *Pattern Recognition Letters*, vol. 140, pp. 78–85, 2021.

[28] V. J. Louis and S. Mitra, "Evaluating the role of crop burning in Delhi pollution using satellite data," *Remote Sensing*, vol. 12, no. 18, Article 3021, 2020.

[29] N. Singh and A. Gupta, "Forecasting extreme pollution days using deep attention networks," *Environmental Modeling and Assessment*, vol. 28, pp. 455–468, 2023.

[30] S. Kumar, R. Chatterjee, and P. Yadav, "Comparative study of ARIMA, LSTM and hybrid models for $PM_{2.5}$ prediction in Indian megacities," *Urban Climate*, vol. 49, 101076, 2023.