International Journal of Technology

http://ijtech.eng.ui.ac.id

Research Article

Fine Particulate Matter ($PM_{2.5}$) Concentration Prediction in an Urban Area of Thailand Using Deep Learning Models

Jakkaphan Whasphuttisit[1*], **Watchareewan Jitsakul**[1]

[1]Department of Information Technology Management, King Mongkut's University of Technology North Bangkok, 1518 Pracharat 1 Road, Wongsawang, Bangsue, Bangkok, 10800, Thailand

*Corresponding author: s6407021910016@e-mail.kmutnb.ac.th; Tel.: +66-804423615; Fax: +66-25874350

Abstract: Fine particulate matter with an aerodynamic diameter of 2.5 micrometres or less ($PM_{2.5}$) poses major health risks. This study compares twelve deep-learning architectures — nine recurrent variants plus CNN-LSTM, PatchTST, and a Transformer — to machine-learning, statistical, and lag-1 persistence baselines for one-hour-ahead $PM_{2.5}$ forecasting at a Bangkok station (PCD-Air4Thai 35T, Lat Krabang), using 24,840 hourly observations (November 2022–August 2025). Every deep model used one identical, leakage-free pipeline; therefore, differences reflect architecture, not preprocessing. Accuracy was assessed using RMSE, MAE, sMAPE, R^2, and MASE over five runs; all 171 model pairs were tested with the Diebold–Mariano test and the Benjamini–Hochberg correction ($\alpha = 0.05$). The Bidirectional group (Bi-RNN, Bi-LSTM, Bi-GRU) formed the statistically best tier (hold-out RMSE = 5.712–5.743 µg/m^3, R^2 = 0.652–0.656), significantly outperforming all other architectures ($p_{BH} < 0.05$), with no within-group difference. The best model depended on the evaluation protocol: Bi-RNN led on the hold-out (5.712 µg/m^3) and Bi-GRU led on the pooled RMSE (5.39 µg/m^3). The same two-tier separation was reproduced by an expanding-window walk-forward validation (Spearman ρ = 0.573). Attention did not improve accuracy: all three attention variants exhibited severe cross-validation instability, and PatchTST and the Transformer performed worst. Tree-based baselines were competitive (Random Forest, 5.825 µg/m^3), and one-step ARIMA/SARIMA were reasonable (SARIMA $R^2 = 0.622$). All models outperformed lag-1 persistence (6.258 µg/m^3) by only ∼5–9%; SHAP and LIME identified $PM_{2.5}$_lag1 as the dominant predictor. Bidirectional recurrence — not attention or transformer complexity — offers the best short-term urban $PM_{2.5}$ forecasting at this data scale.

Keywords: Attention mechanism; Deep learning; $PM_{2.5}$ forecasting; Recurrent neural network; Time series analysis

1. Introduction

Fine particulate matter ($PM_{2.5}$), defined as particles with an aerodynamic diameter of no more than 2.5 m, is a significant risk factor for human health. Pope and Dockery (2006) summarized long-term evidence demonstrating a causal relationship between $PM_{2.5}$ exposure and increased cardiovascular and respiratory disease mortality. Cohen et al. (2017) used data from the 2015 Global Burden of Disease Study to estimate that ambient $PM_{2.5}$ pollution causes 4.2 million premature deaths annually worldwide, representing 7.6% of all deaths. Owing to their small size, these particles can penetrate the upper respiratory tract and embed themselves deep within the alveoli, causing permanent respiratory and cardiovascular damage.

In 2021, the World Health Organization lowered its annual $PM_{2.5}$ guideline from 10 µg/m^3 to 5 µg/m^3 and its 24-hour guideline from 25 µg/m^3 to 15 µg/m^3 (WHO, 2021), and only 9% of monitored countries met these guidelines in 2023, with South and Southeast Asia consistently recording the highest pollution levels (IQAir, 2024). For Thailand, Son et al. (2023) analyzed

https://doi.org/10.14716/ijtech.v17i4.8040
Received January 2026; Revised April 2026; Accepted July 2026

TROPOMI satellite data and estimated a 2020 annual average $PM_{2.5}$ of 21.4 µg/m^3 — 4.3 times the WHO guideline — ranking Thailand 34th among the 118 most polluted countries. The problem is worsened during the dry season (December–April) by pollutants from open burning, traffic, and industry, coupled with unfavorable dispersion conditions.

Bangkok, a metropolis with over 10 million inhabitants and the economic hub of the country, regularly experiences acute $PM_{2.5}$ pollution. In a study of hourly $PM_{2.5}$ levels in Bangkok, Bhatta et al. (2026) found a clear correlation between $PM_{2.5}$ concentration and meteorological factors and observed distinct diurnal and seasonal patterns. Accurate $PM_{2.5}$ forecasting is therefore crucial for early warning systems that allow vulnerable populations to adjust their behavior to reduce exposure and help regulatory agencies plan control measures during pollution episodes.

Research on air-quality forecasting has evolved progressively from traditional statistical models to more sophisticated deep learning through machine learning (Goodfellow et al., 2016; LeCun et al., 2015; Schmidhuber, 2015). This review groups studies using model type and research approach to clearly highlight developments and gaps.

Traditional statistical models such as the Autoregressive Integrated Moving Average (ARIMA) and its seasonal extension, SARIMA, have long been standard tools for time-series forecasting (Box et al., 2015), but they are limited in capturing the nonlinearity and long-range dependencies of hourly $PM_{2.5}$ series (Hyndman & Athanasopoulos, 2021). In a large-scale comparison, Makridakis et al. (2018) found that statistical models remain competitive for short-term forecasting, whereas machine learning and deep learning hold a clear advantage for data with complex patterns, multiple seasonalities, or many explanatory variables. In Thailand, Parntrasri and Puangthongthub (2026) combined Extreme Value Theory with ARIMA to forecast weekly $PM_{2.5}$ peaks in the northeast and found ARIMA limited in capturing extreme transboundary-haze events, reflecting the need for a more robust approach.

Traditional machine learning for air-quality forecasting includes Random Forest (Breiman, 2001) and Gradient Boosting (Friedman, 2001), popular ensemble approaches that capture nonlinear relationships and resist outliers without distributional assumptions. Son et al. (2023) found that deep learning learned the relationship between TROPOMI satellite signals and provincial $PM_{2.5}$ in Thailand better than Random Forest and XGBoost. Bhatta et al. (2026) compared random forest, gradient boosting, and LSTM for hourly $PM_{2.5}$ forecasting in Bangkok and found clear limitations of the machine-learning models during sudden surges, while Patel et al. (2025) found LSTM significantly more effective than traditional machine-learning models in Maharashtra, India. However, most of these studies tested only two to four architectures and did not compare multiple deep-learning architectures under tightly controlled conditions.

Recurrent neural networks (RNNs), originally proposed by Elman (1990) and still widely applied in recent air-quality modelling Wang et al. (2026), process sequences by forwarding hidden states. Hochreiter and Schmidhuber (1997) solved this with the LSTM architecture, whose three-gate mechanism (input, forget, and output gates) enables efficient learning of long-term dependencies; LSTM subsequently became a standard architecture for time-series forecasting (Goodfellow et al., 2016) and remains the backbone of recent urban air-quality models (Eren et al., 2025; Zhao et al., 2026). Applications confirm this advantage: Park et al. (2023) found LSTM best during rapid concentration changes when forecasting $PM_{2.5}$ in South Korean container-port areas; Keerthana and Ushasree (2024) reported LSTM significantly more accurate than RNN owing to superior long-term pattern recognition; and Ahmad et al. (2024) confirmed that LSTM consistently outperforms standalone statistical baselines in learning long-range dependencies in non-stationary sequences.

Gated recurrent unit and hybrid models. The GRU, proposed by Cho et al. (2014), simplifies the LSTM by combining the input and forget gates into a single update gate, resulting in fewer parameters and faster training. Its performance is comparable to that of the LSTM on medium-sized datasets. Drewil and Al-Bahadili (2022) investigated the use of GAs to fine-tune LSTM hyperparameters and found that the selection of appropriate hyperparameters signifi-

cantly affected performance. Eren et al. (2025) compared four hyperparameter optimization methods for LSTMs and found that Bayesian optimization yielded the best results with the fewest trials. This finding informed the model training design adopted in the present study.

Hybrid models that combine CNNs and RNNs leverage the strengths of both architectures. Sharma et al. (2020) found a CNN–LSTM significantly more accurate than standalone CNN and LSTM models for hourly total-suspended-particle forecasting, and Sreenivasulu and Mokesh Rayalu (2025) combined CNN, LSTM, multi-head attention, and GRU for AQI forecasting, achieving an R^2 of up to 0.98 — although such models are complex, prone to overfitting when data are limited, and rarely compared against simpler alternatives. Wang et al. (2026) estimated air quality directly from surveillance images using a CNN–RNN framework, and Duan et al. (2026) applied CNN–LSTM to accelerometer signals for fall detection (99.3% accuracy), confirming that convolutional feature extraction combined with recurrent temporal modeling is a robust hybrid design for sequential spatiotemporal data.

Bidirectional RNNs and the attention mechanism extend these designs. The bidirectional RNN (Bi-RNN) of (Schuster & Paliwal, 1997) processes the sequence in both the forward and backward directions; bidirectional variants continue to deliver state-of-the-art PM2.5 forecasts (Gupta et al., 2026). The attention mechanism (Bahdanau et al., 2015) dynamically weights the importance of different time steps, and Vaswani et al. (2017) advanced it into the Transformer architecture, which relies solely on self-attention and has begun to be applied to environmental time-series forecasting. Zhao et al. (2026) found that integrating WaveNet with LSTM captured multiscale temporal patterns better than a standalone LSTM at a substantially higher computational cost. Zhang and Zhou (2024) showed that deep-model performance is directly affected by the quality of missing-data handling, motivating the outlier-resistant moving-median imputation used here. Al-Selwi et al. (2023) demonstrated that attention-based models can outperform non-attention baselines and classical methods such as ARIMA and SVM by selectively weighting the most informative time steps.

A major limitation of deep learning is its "black-box" nature, which makes interpreting the decision-making process difficult. Lundberg and Lee (2017) proposed SHAP (SHapley Additive exPlanations), grounded in game theory, to provide consistent explanations of the predictions of any machine-learning model, identifying which variables have the greatest influence and in which direction. Integrating attention-weight heatmaps with SHAP enables this study to provide a two-level explainability analysis, an aspect lacking in most prior work.

A common problem in forecasting research is the reporting of only point estimates of performance metrics (RMSE, MAE, R^2), without testing whether the differences between models are statistically significant or merely the result of random variation in the data. (Diebold & Mariano, 1995) proposed a test (the DM test); recent work extends this framework of predictive-ability inference to machine-learning forecasts (Escanciano & Parra, 2026). In the context of $PM_{2.5}$ forecasting for Bangkok, the present study applies the DM test to all 171 unique model pairs among the 19 models evaluated, with the Benjamini–Hochberg correction for multiple comparisons. The literature reviewed above illustrates the development of the field and highlights the gaps that the present study addresses.

The abovementioned systematic literature review revealed five key gaps that this study aims to address. First, systematic benchmarks across all three levels of RNN architecture are lacking: previous studies (Keerthana & Ushasree, 2024; Park et al., 2023; Patel et al., 2025) tested only two to four architectures, and no controlled comparison exists of the Vanilla, Bidirectional, and Attention-based variants across all three families (RNN, LSTM, and GRU). Second, statistical significance testing between models is rare: most studies report only point estimates without the Diebold–Mariano test (Diebold & Mariano, 1995; Escanciano & Parra, 2026) or other formal tests. Third, dual explainability (SHAP + attention weights) is lacking, although explainability is crucial to the reliability of artificial-intelligence models in public-health and environmental applications (Lundberg & Lee, 2017). Fourth, hourly, single-station studies for Bangkok are scarce: most research in Thailand uses annual, provincial-level data (Son et al., 2023) or weekly

data (Parntrasri & Puangthongthub, 2026), and although Bhatta et al. (2026) studied hourly $PM_{2.5}$ in Bangkok, they tested only three models without significance testing. Fifth, many studies fit scalers on the entire dataset, leaking test-set information and biasing evaluation; this study fits the scaler only on the training set so that the evaluation reflects real-world performance.

The main objective of this study is to develop and compare deep-learning models for forecasting hourly $PM_{2.5}$ concentrations at Lat Krabang Station, Bangkok. The objectives are as follows:

1. To compare the performance of twelve deep-learning model architectures — nine RNN-based (Vanilla: RNN, LSTM, GRU; Bidirectional: Bi-RNN, Bi-LSTM, Bi-GRU; Attention-augmented: Attn-RNN, Attn-LSTM, Attn-GRU) and three additional models (CNN-LSTM, PatchTST, and a Vanilla Transformer) — under a single, identical, leakage-free pipeline, using RMSE, MAE, sMAPE, MASE, and R^2 with 95% confidence intervals from 5 independent runs.

2. To compare the deep-learning models with eight baseline models (Naïve Lag-24h, lag-1 persistence, Random Forest, Gradient Boosting, XGBoost, ARIMA, SARIMA, and Prophet) and to test the statistical significance of the differences using the Benjamini–Hochberg correction and the Diebold–Mariano test (Diebold & Mariano, 1995; Escanciano & Parra, 2026).

3. To develop a leakage-free data-preparation pipeline that includes moving-median imputation (24-hour window) and cyclic feature engineering for six variables ($PM_{2.5}$ and five meteorological variables).

4. To explain the decision-making mechanisms of the models using SHAP analysis and attention-weight heatmaps (for attention-based models) to identify the most influential variables and time steps.

This study presents four distinct contributions to the literature. First, a systematic benchmark of all nine RNN architectures (the Vanilla, Bidirectional, and Attention variants of RNN, LSTM, and GRU) plus CNN-LSTM, PatchTST, and a Vanilla Transformer, together with machine-learning, statistical, and persistence baselines, under a single, identical, leakage-free pipeline — so that the comparison reflects architecture rather than preprocessing. Second, a three-dimensional evaluation — accuracy (RMSE, MAE, sMAPE, MASE, R^2), stability (95% CI from five independent runs and blocked cross-validation), and statistical significance, with all 171 unique model pairs tested using the Diebold–Mariano test and the Benjamini–Hochberg correction. Third, a single-station hourly case study for Bangkok spanning 24,840 time steps (34 months) — longer than previous studies in the same context (Bhatta et al., 2026) — and among the first to evaluate bidirectional and attention-based architectures for $PM_{2.5}$ in Bangkok. Fourth, dual explainability via SHAP and attention-weight heatmaps, complemented by LIME applied to a deep model.

2. Methodology

This study developed the framework shown in Figure 1, which divides the process into six main steps: data collection, data preparation, data splitting, model building and training, model testing and evaluation, and model interpretation. Each step is described in detail below.

2.1 Data Gathering

The primary data used in this study are hourly $PM_{2.5}$ concentrations from PCD-Air4Thai air-quality monitoring Station No. 35T in Lat Krabang District, Bangkok, Thailand. The data

cover the period from November 2022 to August 2025 — a total of two years and ten months — and comprise 24,840 data points.

The study used six variables: (1) $PM_{2.5}$ (the target variable), (2) wind speed, (3) wind direction, (4) temperature, (5) relative humidity, and (6) air pressure. The five meteorological variables were included alongside $PM_{2.5}$ to enable the models to learn the multivariate relationships between meteorological conditions and $PM_{2.5}$ concentrations more comprehensively.

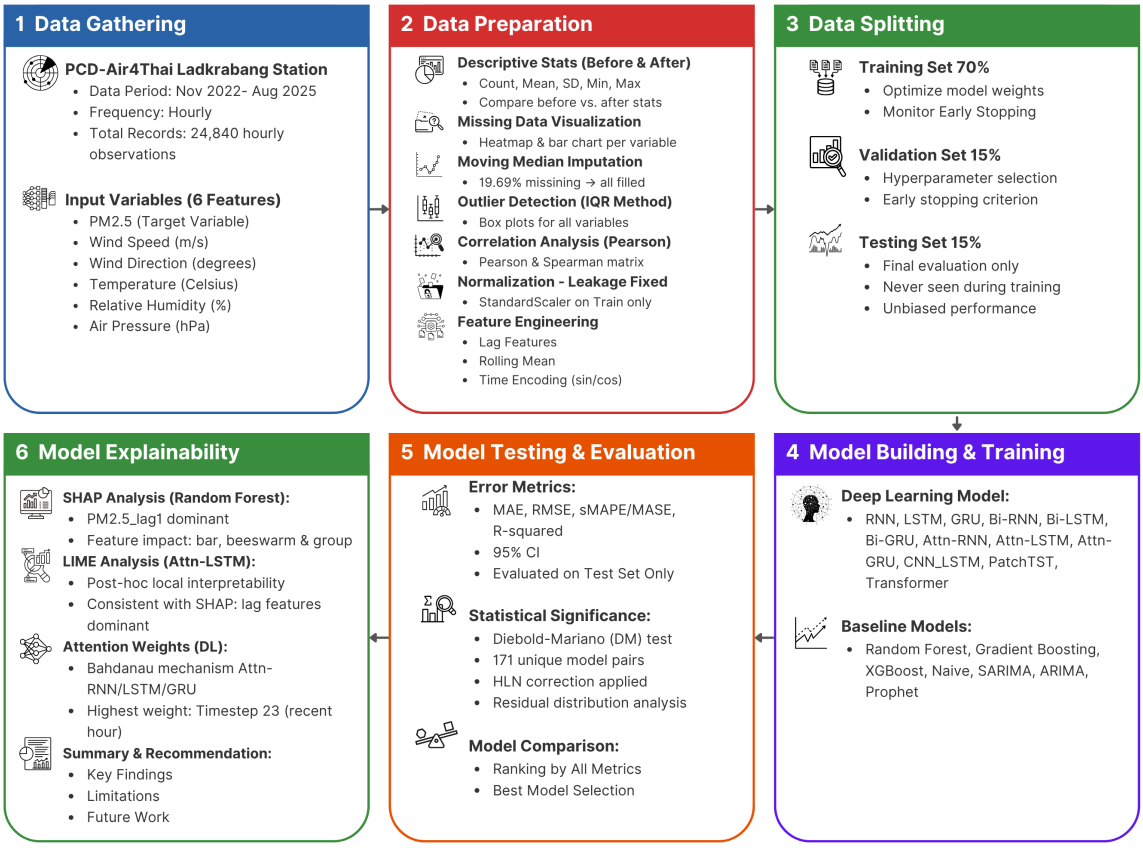

Figure 1 Research Framework — Forecasting $PM_{2.5}$ Concentration using deep learning

2.2 Data Preparation

Before the data can be used to build and train the models, they must be prepared to ensure completeness and readiness for analysis. This study involves eight sub-steps of data preparation, as described below.

1. Descriptive Statistics Before the Imputation. The first step is to perform a descriptive statistical analysis of the raw data before imputing missing values to characterize the data and quantify the extent of the missing values. The analysis covers the number of data points (Count), mean, standard deviation (SD), minimum (Min), maximum (Max), skewness, kurtosis, and percentage of missing values (Missing %). A missing data visualization is then produced to examine the distribution patterns of missing values across variables and over time. A heatmap of the daily missing-value rate for each variable and a bar chart of the percentage of missing values per variable were generated. These visualizations help identify the periods during which the measurement equipment malfunctioned and missing values accumulated.

2. Moving median imputation. Missing values were imputed using a moving-median method with a window size of 24 hours, corresponding to the daily $PM_{2.5}$ cycle. For each

missing value, the method computes the median of the observations within a ±12-hour window centered on the gap, rather than the mean, which is sensitive to outliers. The moving median is robust to extreme values and better preserves the data distribution relative to linear interpolation.

3. Descriptive Statistics after the Imputation. Descriptive statistics were computed after imputation to compare the data distributions before and after imputation and to verify that the method did not substantially alter the principal statistics.

4. Outlier Detection. Outlier detection was performed using the interquartile range (IQR) method after imputation, with the lower bound set at Q1 − 1.5×IQR and the upper bound at Q3 + 1.5×IQR. The IQR method was chosen because it is robust to skewed data, which is characteristic of $PM_{2.5}$ concentrations. The detection results for all six variables are presented as box plots. In this study, all outliers were retained rather than removed because extreme values in air quality data often reflect real-world events, such as unusually high $PM_{2.5}$ levels in winter or biomass-burning episodes in nearby areas. Removing them might cause the models to lose crucial information for learning extreme patterns in the data.

5. Correlation Analysis. The relationships between all pairs of variables were analyzed using two correlation coefficients. The first parameter is the Pearson correlation coefficient, which measures the linear relationship between variables. The second parameter is the Spearman correlation coefficient, which measures the monotonic (rank-order) relationship and is robust to non-normally distributed data. The use of both coefficients enables the examination of both linear and monotonic relationships, which aids in understanding the associations between the meteorological variables and $PM_{2.5}$ before model building.

6. Normalization. To prevent data leakage that could lead to overly optimistic evaluation results, feature scaling was performed using a StandardScaler fitted only on the training set. The fitted parameters were then applied to transform the training, validation, and test sets without refitting. This procedure reflects a real-world scenario in which future data are unknown.

7. Feature Engineering. Three additional feature types were created to enrich the feature set and help the models learn temporal patterns. Lag features comprise $PM_{2.5}$ lags of 1–24 hours ($PM_{2.5}$_lag1–$PM_{2.5}$_lag24), for 24 lag features. The rolling statistics provide the rolling mean and standard deviation over 3-, 6-, 12-, and 24-hour windows (eight rolling features). Cyclical encoding represents the hour, day-of-week, and month using sine and cosine functions to preserve continuity (e.g., hour 23 is close to hour 0), yielding six cyclical features. For the deep-learning models, the data were arranged as sequences of 24 time steps × 6 features to predict the next-hour $PM_{2.5}$ value (one step ahead); the machine-learning models used a total of 43 engineered features (lag, rolling-statistic, cyclical, and meteorological features).

2.3 Data Splitting

The data were split chronologically into three subsets in a 70:15:15 ratio as follows:

Splitting the dataset chronologically, rather than randomly, prevents look-ahead bias, which would occur if a model learned future information during training. Keeping the validation set separate from the test set ensures that the final evaluation of the test set is unbiased. The dataset comprises 24,840 raw hourly records, of which 24,768 were used as labeled samples across the three partitions (Table S1). The remaining 72 records correspond to the initial warm-up period required for the longest lag feature ($PM_{2.5}$_lag24): the first 24 records cannot serve as prediction targets because their lag-24 values fall outside the dataset. To prevent any leakage between the

training, validation, and test windows, an additional 48 records at the partition boundaries are excluded.

The chronological hold-out was used as the primary evaluation, complemented by blocked temporal cross-validation and an expanding-window (rolling-origin) walk-forward validation to verify robustness; the appropriateness of a blocked hold-out for short-horizon forecasting with a large, representative test period is supported by Bergmeir and Benítez (2012) and Hyndman and Athanasopoulos (2021). This condition is satisfied here: the 15% test partition spans March–August 2025 (3,702 hours), covering a full seasonal transition. The three-fold blocked temporal cross-validation, conducted under the same training budget as the hold-out (up to 60 epochs, early-stopping patience of 10), was broadly consistent with the hold-out: the bidirectional and standard recurrent group was the most stable (CV RMSE = 5.27–5.64 $\mu g/m^3$) and the attention/transformer group unstable. The CV winner (Bi-GRU, 5.27 $\mu g/m^3$) and the hold-out winner (Bi-RNN, 5.71 $\mu g/m^3$) differ by less than 0.5 $\mu g/m^3$ — within run-to-run variability — so both evaluations support the same architectural conclusion: bidirectional recurrence is the most accurate and robust design for this dataset.

2.4 Three-Fold Blocked Temporal Cross-Validation Method

To verify that the hold-out rankings are not specific to the single test period, a three-fold blocked temporal cross-validation (CV) was applied within the 70% training partition. The training data were divided into three consecutive, chronologically ordered folds of equal length. In each fold, the earlier time steps served as the sub-training set and the immediately following segment as the fold's validation set, strictly preserving temporal ordering and preventing any look-ahead bias. Each of the twelve deep-learning models was retrained from scratch on every fold under the same fixed hyperparameter configuration used for the hold-out experiment, and the fold root mean square error (RMSE) on each validation segment was recorded.

2.5 Model building and training

2.5.1 Deep learning model, 9 architectures

This study developed and compared nine deep-learning models based on recurrent neural networks (RNNs), grouped as follows. The first is the Vanilla group: RNN, LSTM, and GRU, which are basic architectures that process data in a single (unidirectional) direction. The second is the Bidirectional group: Bi-RNN, Bi-LSTM, and Bi-GRU, which process data in both the forward and backward directions. The third is the attention group: Attn-RNN, Attn-LSTM, and Attn-GRU, which incorporate the Bahdanau attention mechanism to allow the models to weight important time steps.

2.5.2 Fixed Configuration

Table 1 presents the fixed hyperparameter configuration applied to all deep-learning models.

2.5.3 Additional Deep Learning Architectures (CNN-LSTM, PatchTST, and Vanilla Transformer)

Three additional deep-learning architectures (CNN-LSTM, PatchTST, and a Vanilla Transformer) were implemented and evaluated under the same unified protocol as the nine RNN models: the same 24-time-step × 6-feature input window, Adam optimizer (lr = 1×10^{-3}), batch size 512, up to 60 epochs with early stopping (patience 10), MSE loss, and five independent runs (Table 1). This ensures that the comparison reflects architecture rather than preprocessing or tuning. The three architectures are summarized below.

CNN-LSTM: a 1D convolutional layer (64 filters, kernel size 3, same padding) extracts local temporal patterns from the 24×6 input window, followed by a MaxPooling1D layer (stride 2) and an LSTM layer (64 units); a dense head (Dense(32, ReLU) → Dropout 0.1 → Dense(1))

produces the one-hour-ahead forecast.

Table 1 Fixed parameter configurations

Parameter	Values	Reason
Units (Hidden Size)	64	Through the Grid Search test, a balance between efficiency and training time was found.
Layers	1	Sufficient for hourly time series data.
Dropout Rate	0.10	Prevent overfitting without significantly reducing accuracy.
L2 Regularization	1×10^{-3}	Control the weight/size.
Batch Size	512	Suitable for the data size and GPU memory.
Epochs	≤ 60	Combined with Early Stopping (patience=10)
Optimizer	Adam (lr=0.001)	Stable Adaptive Learning Rate
Loss Function	MSE	Suitable for regression problems.
Number of test rounds	5 rounds/model	Calculate Mean ± 95% CI.

PatchTST: The 24-step input window is divided into non-overlapping patches (patch length 12, stride 12), which are projected to d_model = 64; a Transformer encoder (4 attention heads, feed-forward dimension 128, dropout 0.1, with residual connections and layer normalization) processes the patch tokens, and global average pooling followed by Dense(1) produces the forecast.

Vanilla Transformer: the 24-step input window is projected to d_model = 32; a single transformer encoder block (2 attention heads, key dimension 16, feed-forward dimension 64, dropout 0.1, with residual connections and layer normalization) processes the tokens, followed by global average pooling and Dense(1).

2.5.4 Baseline (Benchmark) models

Eight reference models were implemented to serve as benchmarks. Naïve Lag-24h uses the value from 24 h earlier as the forecast, and a lag-1 persistence baseline uses the previous-hour value ($\hat{y}$_t = y_{t−1}), the natural reference for one-step-ahead forecasting. The same 43 engineered features were used for the Random Forest (100 trees, max_depth 12), Gradient Boosting (200 estimators, learning_rate 0.05, max_depth 5), and XGBoost (300 estimators, max_depth 6, learning_rate 0.05). ARIMA(2,0,2) and SARIMA$(1,0,1)(1,0,1)_{24}$ are fitted to the $PM_{2.5}$ training series and evaluated as one-step walk-forward forecasts (fixed parameters, Kalman state updated at every step), matching the one-step setup of the deep learning models. The Facebook Prophet is evaluated as an additive trend–seasonality decomposition baseline (weekly and daily seasonality, changepoint_prior_scale = 0.05).

2.6 Model testing and evaluation

Model evaluation was performed on a held-out test set that was never seen during training using the following metrics:

2.6.1 Evaluation metrics

Model performance was evaluated using RMSE, MAE, MAPE, sMAPE, MASE, and R^2. RMSE (the square root of the mean squared error) penalizes large errors more heavily than small ones; MAE gives the average error magnitude in the original units; MAPE expresses the error as a percentage, and the symmetric sMAPE is additionally reported because MAPE is unstable when $PM_{2.5}$ approaches zero; MASE (the MAE divided by the in-sample one-step naïve MAE) below 1 indicates greater skill than the naïve one-step forecast; and R^2 measures the proportion of explained variance. In addition, a pooled RMSE — the unweighted mean of the hold-out RMSE and the three blocked-CV fold RMSEs — provides a single ranking that integrates all

four evaluation scenarios and guards against selecting a model that overfits the distribution of a single test partition (Table 3).

In addition, the Diebold–Mariano (DM) test — with the Harvey–Leybourne–Newbold (HLN) small-sample correction — was performed for all 171 unique model pairs, with Benjamini–Hochberg control of the false discovery rate (FDR) at the significance level $\alpha = 0.05$.

2.7 Model Explainability

To understand the decision-making mechanisms of the models, two analytical approaches were employed. The first step is the SHAP analysis (SHapley Additive exPlanations). This approach assessed feature importance using TreeExplainer on a random forest, computing SHAP values for each feature based on game theory to identify which features have the greatest impact on the forecast. The second method is attention-weight analysis. This approach examined the attention weights from the Attn-RNN, Attn-LSTM, and Attn-GRU models to determine which time steps (0 = oldest, 23 = newest) receive the most weight during forecasting.

The third approach applied local interpretable model-agnostic explanations (LIME) to provide post-hoc local interpretability for the best-performing attention-based model (Attn-LSTM), serving as a model-agnostic complement to SHAP. Unlike SHAP, which is applied via TreeExplainer and is specific to tree-based models, LIME perturbs individual input instances and fits a local linear surrogate to approximate the behavior of the deep model in the neighborhood of each prediction. LIME was applied with 200 perturbations per instance for each of the 50 randomly selected test samples, and the mean absolute LIME coefficient across all 50 samples was used as an aggregate feature-importance measure. The degree of convergence between the LIME and SHAP feature rankings provides cross-method validation of the identified feature-importance structure, lending additional confidence to the interpretability findings.

3. Results and Discussion

This section presents the results in the procedural order of the research framework: data preparation, exploratory data analysis (EDA), feature engineering, baseline models, deep-learning models, statistical comparisons, and model explainability.

3.1 Data preparation and exploratory analysis results

Table S2 presents the descriptive statistics of the raw data before imputation. The raw dataset comprised 24,840 hourly records (November 2022–August 2025); $PM_{2.5}$ had the highest proportion of missing values (19.69%), and Figure S1 shows that data were missing for several consecutive days in some months, motivating an imputation method that is robust to long gaps and outliers.

After moving-median imputation (24-hour window), the missing rate was reduced to 0% while the principal statistical characteristics of the data were preserved (Table S3), and Figure S2 confirms that the imputed series closely tracks the original series for all variables.

Figures S3 and S4 present the $PM_{2.5}$ time series over the study period. $PM_{2.5}$ concentrations were high in winter (November–February) and low during the rainy season (June–October), consistent with the Bangkok climate. A diurnal pattern was also observed, with elevated concentrations in the morning (06:00–09:00) and evening (19:00–22:00), corresponding to peak traffic hours.

Figure S5 presents the correlation matrix for all pairs of variables. Air pressure had the strongest positive correlation with $PM_{2.5}$ (Pearson r = +0.439; Spearman $\rho = +0.497$), whereas temperature showed a weak negative correlation (r = −0.157). The correlation between wind speed and wind direction was negligible (r < 0.12).

Figure S6 presents box plots of all variables used to detect outliers via the IQR method. All outliers were retained in this study because they reflect genuine environmental events.

Stationarity was assessed using both the augmented Dickey–Fuller (ADF) and KPSS tests.

The ADF test yielded a statistic of −6.906 ($p < 0.001$), indicating that the $PM_{2.5}$ series is stationary, whereas the KPSS test yielded a statistic of 1.110 ($p = 0.010$), indicating local non-stationarity — a pattern typical of series with a strong seasonal component. Because the ADF test rejected the unit-root null hypothesis, a non-differenced specification ($d = 0$) was used for ARIMA/SARIMA, with the seasonal terms and engineered features rather than differencing captured the seasonal and cyclical structure. The ACF and PACF results confirmed a strong autocorrelation at lags of 1, 24, and 48 h.

Figure S7 presents the dataset split and the distribution of $PM_{2.5}$ in each partition: the training set contains 17,364 data points (November 2022–October 2024), the validation set contains 3,702 data points (October 2024–March 2025), and the test set contains 3,702 data points (March 2025–August 2025).

3.2 Results of the baseline model experiment

Before testing the deep-learning models, eight reference models (including a lag-1 persistence baseline and one-step walk-forward ARIMA/SARIMA) were implemented to serve as minimum benchmarks.

The sMAPE values are reported for all models to ensure comparability across Table S5 and Table 2. sMAPE is preferred over MAPE because MAPE is undefined or unstable when $PM_{2.5}$ values approach zero (as can occur during low-pollution periods), whereas sMAPE bounds the error symmetrically.

Table S4 shows that Random Forest performs best among the baselines (RMSE = 5.825 μg/m^3, $R^2 = 0.642$). When implemented correctly as one-step walk-forward forecasts, SARIMA (RMSE = 5.982 μg/m^3, $R^2 = 0.622$) and ARIMA(2,0,2) (RMSE = 6.176 μg/m^3, $R^2 = 0.597$) are competitive with the lag-1 persistence reference (RMSE = 6.258 μg/m^3); only Prophet (RMSE = 49.742 μg/m^3) and Naïve-24h ($R^2 = -0.045$) perform poorly. This corrects the earlier impression — based on a static multi-step ARIMA/SARIMA implementation — that statistical models are unsuitable for hourly urban $PM_{2.5}$ forecasting.

3.3 Results of the Deep Learning Model Experiments

Table S5 summarizes the evaluation results of the twelve deep-learning models on the test set, averaged across five independent runs per model. All models yield MASE values greater than 1 (1.369–1.849) in Table S5 because the in-sample denominator (the mean absolute one-step change on the training set) understates the difficulty of the more volatile test window under a temporal distribution shift. A MASE above 1 therefore does not imply underperformance against a contemporaneous baseline: every bidirectional model and most deep-learning models surpass the test-set lag-1 persistence benchmark (RMSE = 6.258 μg/m^3). This highlights the importance of reporting MASE alongside a test-matched persistence baseline whenever distribution shifts may be present.

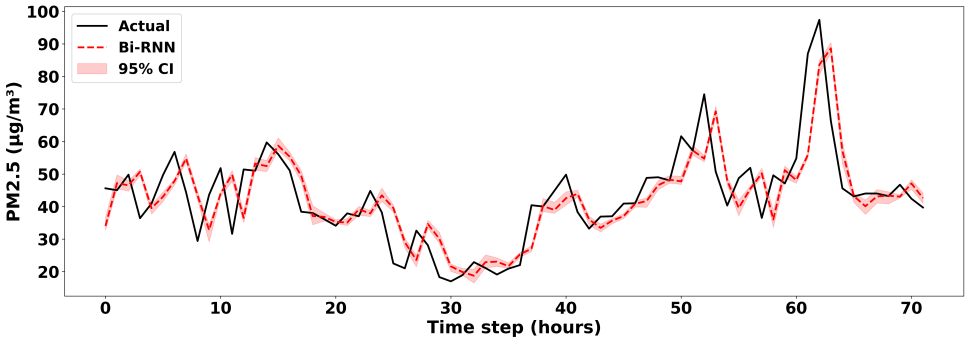

Figure 2 Forecast results of Bi-RNN (the hold-out best model within the statistically-tied bidirectional group) versus the observed $PM_{2.5}$ values over a representative 72-hour window of the test set, with the 95% confidence interval from five runs

Figure 2 presents the one-hour-ahead Bi-RNN forecasts over a 72-hour window alongside the observed values (95% CI from five runs). As shown in Table S5, the Bidirectional group (Bi-RNN, Bi-LSTM, Bi-GRU) formed a statistically indistinguishable best tier under the unified protocol (RMSE = 5.712–5.743 µg/m^3, R^2 = 0.652–0.656), led on the hold-out by Bi-RNN (RMSE = 5.712 µg/m^3, MAE = 4.292 µg/m^3, sMAPE = 21.52%, R^2 = 0.656). Under the pooled RMSE (Table 3), Bi-GRU ranked first (5.39 µg/m^3) — the winner within the bidirectional tier depends on the evaluation protocol. Attn-RNN appeared competitive on the hold-out (5.763) but was severely unstable under cross-validation; the attention-augmented LSTM/GRU variants (Attn-LSTM 5.951, Attn-GRU 5.962) and the patch- and transformer-based models (PatchTST 7.129, Transformer 7.637) were the weakest. Notably, CNN-LSTM (5.876) fell to the middle of the ranking once evaluated under the same feature window and training budget, indicating that its previously reported advantage was a pipeline artifact rather than an architectural one.

3.4 Analysis by the Architecture Group

Vanilla group (RNN, LSTM, and GRU): The three Vanilla models clustered in the middle of the ranking (RNN 5.834, LSTM 5.873, GRU 5.893 µg/m^3) and were statistically indistinguishable from the tree-based baselines, confirming that a plain recurrent cell already captures most of the available short-term signal.

Bidirectional group (Bi-RNN, Bi-LSTM, Bi-GRU): this group occupied the top three positions (5.712–5.743 µg/m^3) and was significantly more accurate than the Vanilla, Attention, and ML baselines (DM, $p_{BH} < 0.05$). Therefore, processing the 24-hour window in both directions yields a consistent and statistically significant gain — the clearest architectural effect observed in this study.

Attention group (Attn-RNN, Attn-LSTM, Attn-GRU): adding Bahdanau attention did not reliably improve accuracy. On the hold-out test, Attn-RNN appeared competitive (RMSE = 5.763), but the pooled evaluation (Table 3) ranked it ninth (pooled RMSE = 7.50 µg/m^3), with CV Fold 3 alone reaching 9.660 µg/m^3 — exposing severe instability. All three attention variants showed high cross-validation variance: Attn-RNN (CV mean 8.074 ± 1.155, pooled rank 9), Attn-LSTM (8.158 ± 1.768, pooled rank 10), and Attn-GRU (6.070 ± 0.507, pooled rank 8). Attn-LSTM (5.951) and Attn-GRU (5.962) also ranked lowest among the recurrent models in the hold-out test. Thus, parameters and instability without reliable accuracy gains at this data scale are added.

Notably, the 60-epoch equal-budget training constraint, imposed to ensure a fair cross-architecture comparison, may be insufficient to fully converge attention-based models. This interpretation is supported by the high cross-validation variance observed for Attn-LSTM (CV RMSE = 8.158 ± 1.768) and Attn-GRU (6.070 ± 0.507) compared with Bi-RNN (5.712 ± 0.026). The multi-horizon analysis further shows that PatchTST and the Transformer improve substantially at h = 6–12, suggesting that their underperformance at h = 1 is partly attributable to the training budget and dataset scale rather than a fundamental architectural limitation. Therefore, the conclusion that "attention does not consistently help" should therefore be read as conditional on the equal-budget, single-station protocol used in this study.

3.5 Comparison of all the models

Table 2 and Figure S8 compare the 19 models (12 deep-learning and 7 reference models, including the lag-1 persistence baseline and the Naïve-24h baseline) on the test set, sorted by RMSE; Prophet is an extreme outlier and is reported only in Table S4. The Naïve-24h (lag-24) model is included in Table 2 and the Diebold–Mariano significance test in Section 3.10 as a lag-24 persistence baseline.

Random Forest (RMSE = 5.825 µg/m^3, R^2 = 0.642) ranked fifth overall — nominally outperforming every Vanilla recurrent model, the Attn-LSTM and Attn-GRU variants, and CNN-LSTM, yet remaining statistically indistinguishable from CNN-LSTM, XGBoost, and Attn-RNN

(Diebold–Mariano test, $p_{BH} > 0.05$). Only the three bidirectional models ranked above it: Bi-RNN significantly ($p_{BH} = 0.009$), and Bi-LSTM and Bi-GRU marginally ($p_{BH} \approx 0.05$). These findings demonstrate that well-engineered tree-based models remain a strong, low-cost alternative for one-hour-ahead urban $PM_{2.5}$ forecasting.

Table 2 Comparison of all 19 models on the test set (sorted by RMSE)

Rank	Model	Type	RMSE	MAE	sMAPE (%)	R^2
1	Bi-RNN	DL-Bidirectional	5.712	4.292	21.52	0.656
2	Bi-LSTM	DL-Bidirectional	5.740	4.362	21.93	0.652
3	Bi-GRU	DL-Bidirectional	5.743	4.348	21.78	0.652
4	Attn-RNN	DL-Attention	5.763	4.405	22.15	0.649
5	Random Forest	ML-Baseline	5.825	4.350	25.75	0.642
6	RNN	DL-Vanilla	5.834	4.404	21.96	0.641
7	XGBoost	ML-Baseline	5.848	4.424	26.66	0.639
8	LSTM	DL-Vanilla	5.873	4.469	22.35	0.636
9	CNN-LSTM	DL-Hybrid	5.876	4.494	22.51	0.636
10	Gradient Boosting	ML-Baseline	5.892	4.403	26.34	0.634
11	GRU	DL-Vanilla	5.893	4.461	22.20	0.633
12	Attn-LSTM	DL-Attention	5.951	4.574	22.83	0.626
13	Attn-GRU	DL-Attention	5.962	4.582	22.86	0.625
14	SARIMA	Statistical	5.982	4.480	25.20	0.622
15	ARIMA(2,0,2)	Statistical	6.176	4.624	26.55	0.597
16	Persistence (lag-1)	Naïve	6.258	4.671	25.96	0.587
17	PatchTST	DL-PatchTST	7.129	5.311	25.67	0.463
18	Transformer	DL-Transformer	7.637	5.796	27.77	0.384
19	Naïve-24h	Naïve	9.950	7.362	41.73	−0.045

3.6 Statistical significance test results (Diebold–Mariano test)

This step determines whether the differences between models are statistically significant. The Diebold–Mariano (DM) test (HLN-corrected) was applied to all 171 unique pairs among the 19 models (Table 2) (including the Naïve-24h lag-24 persistence baseline), with Benjamini–Hochberg control of the false discovery rate at $\alpha = 0.05$.

The DM test results revealed the following key findings:

Best tier (hold-out): Bi-RNN, Bi-LSTM, Bi-GRU, and Attn-RNN are mutually indistinguishable on the hold-out test ($p_{BH} > 0.05$). In the pooled evaluation (Table 3), Attn-RNN falls to ninth (pooled RMSE = 7.50 $\mu g/m^3$), confirming that the three bidirectional models are the only consistently reliable best-tier group.

Bidirectional vs. the rest: each Bidirectional model is significantly more accurate than every Vanilla model, Attn-LSTM, Attn-GRU, CNN-LSTM, PatchTST, the Transformer, and XGBoost ($p_{BH} < 0.05$); Bi-RNN also significantly outperforms Random Forest ($p_{BH} = 0.009$), whereas Bi-LSTM and Bi-GRU are only marginally better than Random Forest ($p_{BH} \approx 0.05$).

CNN-LSTM vs. ML baselines: CNN-LSTM is not significantly better than random forest or XGBoost ($p_{BH} > 0.05$); Attn-LSTM is significantly worse than random forest.

Worst tier: PatchTST and the Transformer are significantly outperformed by almost every other model ($p_{BH} < 0.05$), and all credible models significantly outperform lag-1 persistence and Naïve-24h.

Non-significant pairs include Bi-RNN vs. Bi-LSTM vs. Bi-GRU vs. Attn-RNN, and CNN-LSTM vs. Random Forest vs. XGBoost — that is, within-tier differences lie within run-to-run noise.

3.7 Cross-Validation and RMSE Robustness Analysis

Table 3 presents a unified comparison of all 12 deep-learning models across four evaluation scenarios: the hold-out test and each of the three blocked CV folds, together with the pooled RMSE (the unweighted mean of all four RMSE values) to verify that the hold-out rankings in Table S5 are not an artifact of the specific test period. The models are sorted by the pooled RMSE.

Table 3 Comparison of the unified protocol — hold-out RMSE and three-fold blocked CV RMSE (µg/m^3) for the 12 deep-learning models; sorted by pooled RMSE (the mean of four evaluation scenarios: hold-out + 3 CV folds). Bi-GRU leads in the pooled evaluation (5.39 µg/m^3); Bi-RNN leads in the hold-out test alone (5.712 µg/m^3)

Rank (Pooled)	Model	Hold-out RMSE	CV Fold 1	CV Fold 2	CV Fold 3	CV Mean ± Std (µg/m^3)	Pooled RMSE
1	Bi-GRU	5.743	5.773	5.175	4.851	5.266 ± 0.382	5.39
2	GRU	5.893	5.773	5.250	4.972	5.332 ± 0.332	5.47
3	Bi-LSTM	5.740	5.841	5.230	5.082	5.385 ± 0.329	5.47
4	LSTM	5.873	5.823	5.274	5.095	5.398 ± 0.310	5.52
5	CNN-LSTM	5.876	5.834	5.247	5.187	5.423 ± 0.292	5.54
6	RNN	5.834	5.870	5.514	5.201	5.528 ± 0.273	5.60
7	Bi-RNN	5.712	5.982	5.579	5.364	5.642 ± 0.256	5.66
8	Attn-GRU	5.962	6.779	5.809	5.623	6.070 ± 0.507	6.04
9	Attn-RNN	5.763	7.617	6.945	9.660	8.074 ± 1.155	7.50
10	Attn-LSTM	5.951	8.096	6.025	10.354	8.158 ± 1.768	7.61
11	PatchTST	7.129	8.487	8.210	9.322	8.673 ± 0.472	8.29
12	Transformer	7.637	10.099	9.543	9.849	9.831 ± 0.227	9.28

Table 3 shows how the evaluation protocol affects the individual model rankings. On the hold-out test alone, Bi-RNN ranks first (RMSE = 5.712 µg/m^3); however, under the pooled criterion, Bi-GRU leads (Pooled RMSE = 5.39 µg/m^3), whereas Bi-RNN falls to seventh place. Despite this intra-group reshuffling, the Bidirectional family (Bi-GRU, Bi-LSTM, Bi-RNN) consistently occupied three of the top seven pooled positions, confirming that the group's superiority, as identified by the DM test, was robust across temporal folds and not specific to the hold-out distribution. In contrast, the attention-based models ranked eighth to tenth in the pooled RMSE (Attn-GRU: 6.04; Attn-RNN: 7.50; Attn-LSTM: 7.61 µg/m^3), indicating that their competitive hold-out performance does not reliably generalize across CV folds.

3.8 Multi-Horizon Forecast Evaluation

To address the concern that the study evaluated only a single forecast horizon (h = 1), a supplementary direct multi-horizon experiment was conducted for h ∈ {1, 6, 12, 24} hours. For each horizon, the model target was shifted by h−1 positions so that the same 24-step, 6-feature input window predicts $PM_{2.5}$ at time t + h directly, without recursive error compounding. All 12 DL architectures and three ML baselines were retrained from scratch with 5 independent random seeds; the MASE denominator was fixed at the training-set value (3.135 µg/m^3) for cross-horizon comparability. Table 4 summarizes the mean RMSE across models and horizons. The skill score (SS = 1 − RMSE_model / RMSE_lag-h persistence) quantifies the improvement over the lag-h benchmark.

At h = 1, the Bi-directional group retained its lead (Bi-RNN RMSE = 5.712 µg/m^3, SS = +0.087), consistent with the main experiment. However, at h = 6 and h = 12, the tree-based ML models outperformed all the DL architectures: Random Forest achieved RMSE = 7.639 µg/m^3 (SS = +0.177) at h = 6 and RMSE = 8.107 µg/m^3 (SS = +0.220) at h = 12, compared with the best DL result (Bi-RNN: 7.960 and 9.150 µg/m^3, respectively). This reversal can be attributed to the greater robustness of ensemble tree methods against the increased label noise inherent in medium-range forecasting and to the fact that direct-horizon feature engineering (e.g., explicitly including lag-h features) benefits ensemble learners more than sequence-to-scalar RNNs with a fixed 24-step window.

At h = 24, no learned model outperformed the lag-24 persistence baseline (Naïve-24h), which achieved RMSE = 6.232 µg/m^3 (SS = 0.000 by definition). The best DL model (Bi-GRU: 8.316 µg/m^3) and the best ML model (Random Forest: 8.159 µg/m^3) both yield negative skill at this horizon. This outcome reflects the strong diurnal periodicity of $PM_{2.5}$ in Bangkok: pollutant concentrations at a given hour closely track those at the same hour on the previous day, so the lag-24 persistence captures the dominant diurnal pattern that purely data-driven models trained on a 24-hour input window fail to exploit explicitly. These results reinforce the utility of incorporating explicit Fourier or seasonal-decomposition features for day-ahead environmental forecasting.

Notably, PatchTST and the Vanilla Transformer, which perform worst at h = 1 (RMSE = 7.129 and 7.637 µg/m^3, with skill scores of −0.139 and −0.220, respectively), recover substantially at longer horizons (at h = 6: RMSE = 8.23 and 8.22 µg/m^3, SS = +0.113 and +0.114, respectively). This convergence pattern is consistent with the known advantage of attention-based architectures for capturing long-range temporal dependencies and suggests that their underperformance at h = 1 is a structural disadvantage rather than a consequence of the limited training budget (60 epochs).

Table 4 Direct multi-horizon forecast results (mean RMSE, µg/m^3, five seeds). An asterisk (*) indicates the best model per horizon. Skill score = 1 − $\text{RMSE}_{\text{model}}$ / $\text{RMSE}_{\text{lag-h persistence}}$; positive values indicate that the model outperforms the lag-h persistence baseline. At h = 24, the Naïve-24h (lag-24) model is used as the persistence reference

Model	h = 1	h = 6	h = 12	h = 24
Bi-RNN	**5.712***	7.960	9.150	8.405
Bi-GRU	5.743	8.436	9.473	8.316
RandomForest	5.825	**7.639***	**8.107***	8.159
PatchTST	7.129	8.227	8.621	8.817
Transformer	7.637	8.216	8.618	8.678
Naïve-24h	9.950	10.489	10.593	**6.232***
Persistence (lag-h)	6.259	9.277	10.396	—
Skill score (best)	+0.087	+0.177	+0.220	0.000

An expanding-window (rolling-origin) walk-forward validation across five successive origins (training window growing from 3,511 to 17,555 samples; five seeds; the same equal-budget protocol as the hold-out; hyperparameters not re-tuned per fold) reproduces the two-tier structure observed on the hold-out set (Table S6): the recurrent and bidirectional models, together with CNN-LSTM, form a tightly clustered top group (walk-forward RMSE = 5.04–5.65 µg/m^3), separated by roughly 2 µg/m^3 from the attention- and transformer-based models (7.6–9.8 µg/m^3). Within the near-tied top group, the precise ordering is not stable — Bi-GRU leads under walk-forward, whereas Bi-RNN leads on the hold-out — and the rank correlation with the hold-out ordering is moderate (Spearman $\rho = 0.573$; Kendall $\tau = 0.424$), driven mainly by reshuffling within the top tier and the instability of the attention models. The tier-level conclusion — that bidirectional and recurrent architectures are the most accurate and robust for this task, and that attention/transformer complexity is not warranted at this data scale — therefore holds across

the hold-out, blocked cross-validation, and walk-forward protocols.

3.9 Explainability analysis results of the proposed model

3.9.1 SHAP Analysis (Random Forest)

Figure S9 and Figure 3 present the SHAP analysis for the random forest, which computes SHAP values for a 500-point data sample using TreeExplainer. The analysis showed that $PM_{2.5}$_lag1 was clearly the most important feature (mean —SHAP— = 8.722) — 21 times higher than that of the second most important feature ($PM_{2.5}$_lag22, mean —SHAP— = 0.406) — indicating that the previous-hour $PM_{2.5}$ value is the strongest predictor.

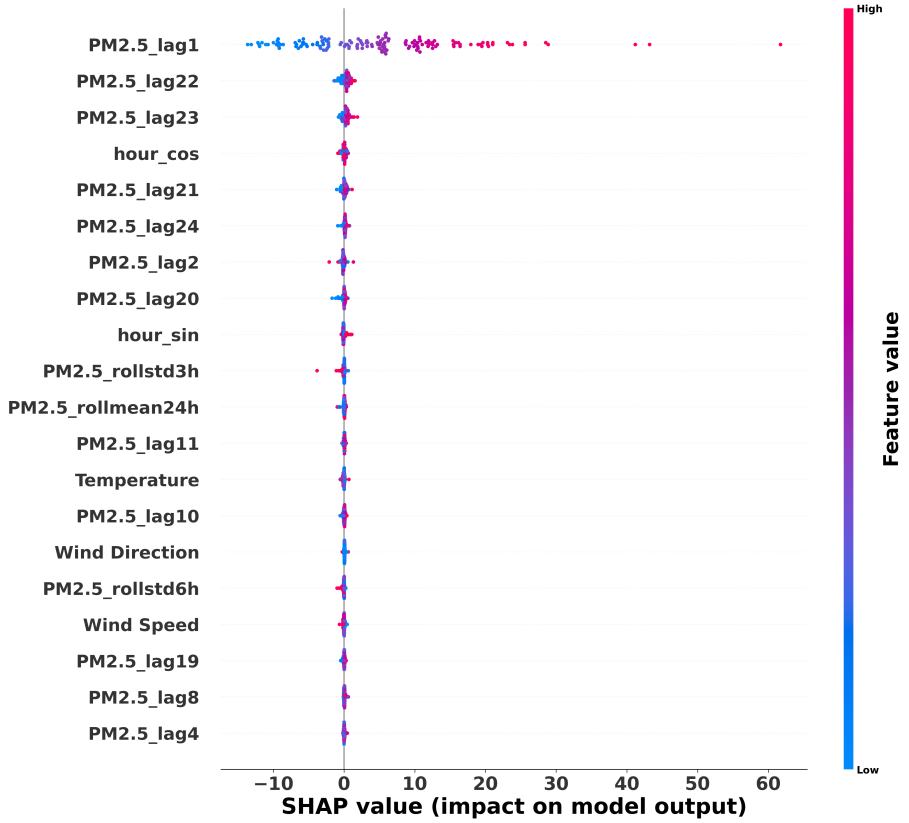

Figure 3 SHAP beeswarm plot for the random forest model, showing the impact of each feature's distribution and direction on the predicted $PM_{2.5}$, with $PM_{2.5}$_lag1 the dominant driver

The feature-group analysis showed that the $PM_{2.5}$ lag features had the highest cumulative importance (summed mean —SHAP— ≈ 10.88), followed by cyclical features such as hour_sin/cos (total importance ≈ 0.48). The low importance of the other features relative to the lag features indicates that historical $PM_{2.5}$ values alone contain sufficient information for short-term (one-hour-ahead) forecasting.

3.9.2 Attention weight analysis

Figure 4 presents the Attn-RNN model's attention-weight heatmap over the first 50 examples from the test set. The attention-weight analysis showed that all three attention models assigned the highest weight to the most recent time step (time step 23, the previous hour), with smaller weights on the immediately preceding hours (≈ time steps 20–22) and almost none on the oldest time steps (0–1). This is consistent with the SHAP finding that $PM_{2.5}$_lag1 is the dominant predictor.

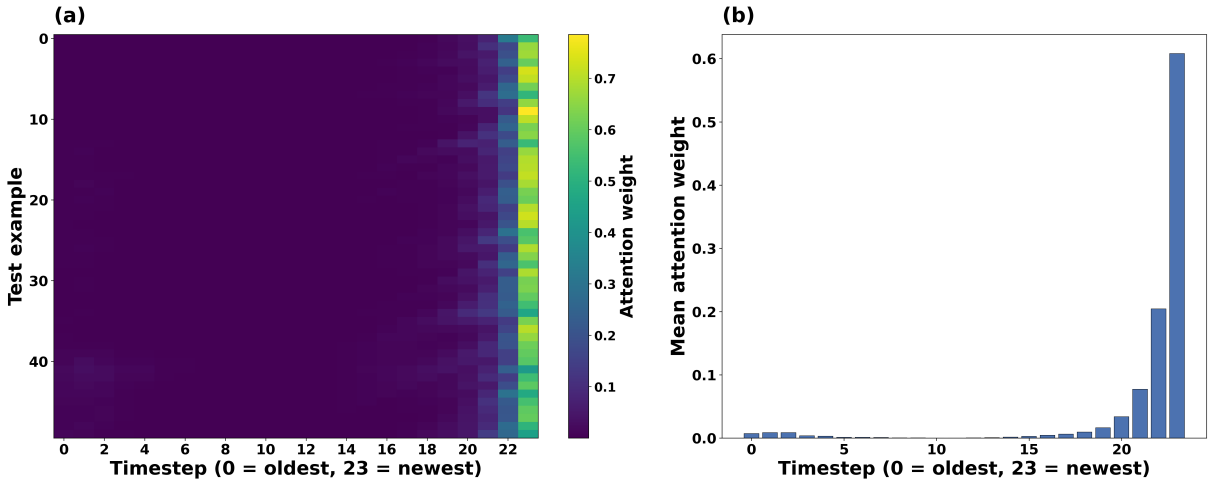

Figure 4 Attention weights of the Attn-RNN model: (a) per-timestep attention weights for 50 representative test examples, and (b) mean attention weight by timestep, indicating that the most recent hour (timestep 23) receives the highest weight

3.9.3 LIME analysis (Attn-LSTM)

Local interpretable model-agnostic explanations (LIME) (Ribeiro et al., 2016) was applied to a representative attention-based deep model (Attn-LSTM) to provide post-hoc local interpretability for the deep learning predictions. Unlike SHAP, which was applied to the random forest using TreeExplainer (a white-box method specific to tree-based models), LIME is fully model-agnostic: it perturbs each input instance and fits a weighted local linear surrogate to approximate the behavior of the complex model in the neighborhood of that instance. LIME was executed with 200 perturbations per instance for each of the 50 randomly selected test samples, and the mean absolute LIME coefficient across all 50 samples was used as an aggregate feature-importance measure.

Figure S10 presents the aggregated LIME feature importance for the Attn-LSTM model. Consistent with the SHAP analysis of the random forest, $PM_{2.5}$_lag1 emerges as the dominant predictor, followed by the recent lags $PM_{2.5}$_lag2 and $PM_{2.5}$_lag3, reflecting the strong short-term temporal persistence of $PM_{2.5}$; the remaining lag features, 24-hour rolling statistics, and meteorological variables contribute small and broadly comparable marginal effects. The convergence of the LIME and SHAP rankings provides cross-method validation of the identified feature-importance structure.

Figure S11 shows a representative local LIME explanation for a high-$PM_{2.5}$ event: an elevated $PM_{2.5}$_lag1 (the previous-hour observation) is the strongest positive driver, corroborated by adjacent lag features, while lower wind speed further increases and off-peak hourly cyclical features moderate the forecast. This local explanation demonstrates the Attn-LSTM model's ability to capture both persistence effects and environmental context, providing the per-prediction transparency essential for operational air-quality forecasting and early-warning applications.

Overall, the LIME analysis corroborates the SHAP findings, demonstrating that $PM_{2.5}$ autoregressive lag features dominate the predictions of both the interpretable random forest and the deep Attn-LSTM, thereby enhancing confidence in the feature-importance structure identified across different XAI methods.

4. Conclusion

This study aimed to evaluate and compare the performance of 19 forecasting models — comprising twelve deep-learning architectures (nine RNN-based: RNN, LSTM, GRU, Bi-RNN, Bi-LSTM, Bi-GRU, Attn-RNN, Attn-LSTM, Attn-GRU; plus CNN-LSTM, PatchTST, and a

Vanilla Transformer) and seven reference models (Random Forest, Gradient Boosting, XGBoost, Naïve Lag-24h, lag-1 persistence, ARIMA(2,0,2), and SARIMA) — for forecasting hourly $PM_{2.5}$ concentrations one hour ahead at PCD-Air4Thai monitoring Station No. 35T, Lat Krabang District, Bangkok, using data from November 2022 to August 2025, totaling 24,840 data points. All deep-learning models were trained under a single, identical, leakage-free pipeline so that the architecture rather than preprocessing reflects the comparison.

4.1 Key findings of the experiment

Finding 1 — Bidirectional recurrence, not attention, drives performance. Under a single, identical pipeline, the Bidirectional group (Bi-RNN, Bi-LSTM, Bi-GRU) constituted the most accurate tier (hold-out RMSE = 5.712–5.743 µg/m^3; pooled Bi-GRU = 5.39 µg/m^3), significantly more accurate (DM, $p_{BH} < 0.05$) than the Vanilla models, Attn-LSTM and Attn-GRU variants, CNN-LSTM, and ML baselines. Adding Bahdanau attention did not reliably improve accuracy: on the hold-out test, Attn-LSTM (5.951) and Attn-GRU (5.962) ranked among the weakest recurrent models, and Attn-RNN appeared tied with the bidirectional tier (RMSE = 5.763) but fell to ninth in the pooled evaluation (7.50 µg/m^3; Table 3), indicating that its hold-out result reflects instability rather than consistent capability.

The Diebold–Mariano test (BH-corrected) confirmed that each bidirectional model significantly outperformed the Vanilla RNN, LSTM, and GRU, as well as the Attn-LSTM, Attn-GRU, CNN-LSTM, PatchTST, Transformer, and XGBoost ($p_{BH} < 0.05$). Bi-RNN significantly outperforms random forest ($p_{BH} = 0.009$), whereas Bi-LSTM and Bi-GRU are only marginally better than random forest ($p_{BH} \approx 0.05$). The four best models — Bi-RNN, Bi-LSTM, Bi-GRU, and Attn-RNN — are statistically indistinguishable from one another.

Finding 2 — Architecture performance hierarchy. The hold-out ordering is as follows: Bidirectional (best tier) > Vanilla RNN/LSTM/GRU ≈ tree-based ML ≈ CNN-LSTM > Attn-LSTM/Attn-GRU > correctly specified SARIMA/ARIMA ≈ persistence > PatchTST/Transformer > Naïve-24h. Attn-RNN was statistically tied with the bidirectional tier on the hold-out test (RMSE = 5.763) but dropped to ninth in the pooled evaluation (7.50 µg/m^3). The bidirectional tier winner also depends on the evaluation protocol: Bi-RNN leads on the hold-out test alone, whereas Bi-GRU leads when the hold-out and CV results are combined (Table 3).

The differences are not statistically significant on the hold-out test within the bidirectional best tier (Bi-RNN 5.712, Bi-LSTM 5.740, Bi-GRU 5.743; all pairwise $p_{BH} > 0.05$). The pooled evaluation (Table 3) confirms that there is no single overall winner: Bi-GRU leads in the combined evaluation (pooled rank 1, 5.39 µg/m^3), whereas Bi-RNN leads in the hold-out test alone (hold-out rank 1, 5.712 µg/m^3). Users should select a model based on their deployment criterion (single-test peak accuracy vs. cross-evaluation stability).

Finding 3 — Tree-based ML is highly competitive. Random Forest (RMSE = 5.825 µg/m^3, $R^2 = 0.642$) ranked fifth overall, ahead of every Vanilla recurrent model, the Attn-LSTM and Attn-GRU variants, and CNN-LSTM, yet was statistically indistinguishable from CNN-LSTM, XGBoost, and Attn-RNN (DM, $p_{BH} > 0.05$); only the three bidirectional models ranked above it. Thus, quality feature engineering enables low-cost, GPU-free models to match most deep networks for one-hour-ahead forecasting.

Finding 4 — Correctly implemented statistical models are reasonable baselines (earlier claim withdrawn). When run as one-step walk-forward forecasts, SARIMA (RMSE = 5.982 µg/m^3, $R^2 = 0.622$) and ARIMA(2,0,2) (RMSE = 6.176 µg/m^3, $R^2 = 0.597$) are competitive and close to the persistence baseline ($R^2 = 0.587$). The earlier conclusion that statistical models are unsuitable for this task arose from a static multi-step implementation, not from any inherent property of the models; only Prophet (additive decomposition) remains a poor fit for the highly variable hourly data.

All credible models (deep-learning, ML, and corrected statistical) significantly outperform the Naïve-24h baseline; however, the gain of the best model over lag-1 persistence was modest (a

∼9% reduction in RMSE), so the practical skill ceiling at this station and horizon was limited.

Finding 5 — $PM_{2.5}$_lag1 is the dominant predictor, and air pressure is the most correlated meteorological variable. The random forest's SHAP analysis shows that $PM_{2.5}$_lag1 ranks far above all other features (mean —SHAP— = 8.722, 21× that of the second feature), corroborated by LIME applied to the deep model; the strong persistence baseline (R^2 = 0.587) is a direct consequence. Among the meteorological variables, air pressure has the strongest correlation with $PM_{2.5}$ (Pearson r = +0.439; Spearman ρ = +0.497).

Finding 6 — Forecast accuracy is horizon-dependent; model rankings shift across horizons. A supplementary multi-horizon experiment (h = 1, 6, 12, and 24 hours) revealed that the Bi-RNNs led at h = 1 (Bi-RNN RMSE = 5.712 µg/m^3), whereas tree-based ML models (Random Forest) outperformed all DL architectures at h = 6 (RMSE = 7.639 µg/m^3) and h = 12 (RMSE = 8.107 µg/m^3). At h = 24, no model outperformed the Naïve-24h (lag-24) persistence baseline (RMSE = 6.232 µg/m^3), reflecting Bangkok's strong diurnal $PM_{2.5}$ cycle. Therefore, model selection should be guided by the operational forecast horizon and not by single-horizon benchmarks.

Finding 7 — Moving-median imputation preserves the statistical characteristics of the data well. Moving-median imputation (24-hour window), used instead of linear interpolation, fully imputed 19.69% of the missing $PM_{2.5}$ values (4,891 data points) while closely preserving the original data's statistical characteristics. The mean changed little (from 30.71 to 30.88 µg/m^3) and the skewness decreased from 1.668 to 1.409, reflecting a more symmetric distribution. This method is more robust to outliers than linear interpolation because it uses the median rather than the mean.

4.2 Limitations of the study

This study has several limitations that should be considered when interpreting the results.

- The experiment was conducted at a single station (Station 35T, Lat Krabang). Therefore, the conclusions are station-specific and may not be generalizable to other stations or cities.
- This study focuses primarily on one-step-ahead (h = 1) forecasting; a supplementary direct multi-horizon analysis (h = 1, 6, 12, and 24) is provided; however, the main model selection, hyperparameter tuning, and statistical tests were optimized for h = 1 only. Model performance at medium-range (h = 6–12) and day-ahead (h = 24) horizons can differ substantially from the h = 1 ranking.
- Despite the use of five meteorological variables, important predictors, such as traffic volume, a biomass-burning index, and boundary-layer height — which could improve accuracy — were not available.
- The SARIMA model in this study does not incorporate exogenous variables; thus, its full potential may not have been realized.

4.3 Future research approaches

Based on this study's findings and limitations, five main directions are proposed for future research: (1) incorporating additional predictors such as traffic volume, biomass-burning indices, and boundary-layer height; (2) extending the experiment to multiple monitoring stations across Bangkok and to additional forecast horizons, for example with graph-based spatiotemporal models; (3) testing data-efficient long-sequence models, given that PatchTST and the Vanilla Transformer performed worst at this data scale; (4) operational deployment of a bidirectional model (Bi-RNN, the hold-out leader; Bi-GRU under the pooled and walk-forward protocols) as the main engine of an area-specific early-warning system; and (5) multi-horizon and multi-output

forecasting, where tree-based models outperform the deep-learning models at longer horizons (h $\geq$ 6) and direct multi-output architectures merit systematic study.

4.4 Final conclusion

This study presents a comprehensive comparison of deep-learning models for $PM_{2.5}$ forecasting under a rigorous testing framework (a leakage-free protocol, a chronological 70/15/15 split, five test cycles per model, and the DM statistical test). Moving-median imputation was used to handle up to 19.69% of the missing data, and five meteorological variables were combined with $PM_{2.5}$ in a multivariate time-series forecasting model.

The results confirm that, under a fair, unified pipeline, the bidirectional recurrent models are the best for one-hour-ahead $PM_{2.5}$ forecasting at Lat Krabang Station, with the Bidirectional group forming the best tier (DM, $p_{BH} < 0.05$; no significant within-group difference). Bi-RNN leads the hold-out test (RMSE = 5.712 µg/m^3, MAE = 4.292 µg/m^3, sMAPE = 21.52%, R^2 = 0.656), and Bi-GRU leads the pooled evaluation across the hold-out and three cross-validation folds (Pooled RMSE = 5.39 µg/m^3, Table 3). Attention-, transformer-, and patch-based architectures did not reliably improve accuracy, tree-based machine learning was competitive, and correctly specified one-step ARIMA/SARIMA models were reasonable baselines. The advantage of the bidirectional tier over lag-1 persistence was modest (a ∼9% reduction in RMSE on the hold-out test).

Acknowledgements

The authors thank King Mongkut's University of Technology North Bangkok for its support of this work. This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors

Conflict of Interest

The authors declare no conflicts of interest.

Supplementary Material

Supplementary figures (Figures S1–S12) and tables (Tables S1–S6) associated with this article are available in a separate supplementary file.

References

Ahmad, F. A., Liu, J., Hashim, F., & Samsudin, K. (2024). Short-term load forecasting utilizing a combination model: A brief review. *International Journal of Technology*, *15*(1), 121–129. https://doi.org/10.14716/ijtech.v15i1.5543

Al-Selwi, H. F., Abd. Aziz, A., Bin Abas, F., Kayani, A., & Noor, N. M. (2023). Attention based spatial-temporal GCN with Kalman filter for traffic flow prediction. *International Journal of Technology*, *14*(6), 1299–1308. https://doi.org/10.14716/ijtech.v14i6.6646

Bahdanau, D., Cho, K., & Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. *Proceedings of the 3rd International Conference on Learning Representations (ICLR 2015)*. https://arxiv.org/abs/1409.0473

Bergmeir, C., & Benítez, J. M. (2012). On the use of cross-validation for time series predictor evaluation. *Information Sciences*, *191*, 192–213. https://doi.org/10.1016/j.ins.2011.12.028

Bhatta, J., Acharya, S. R., & Yang, K. M. (2026). Machine learning-enhanced air quality forecasting and trend analysis: A five-year comprehensive assessment of PM2.5 concentrations in Bangkok, Thailand. *Environmental Challenges*, *22*, 101442. https://doi.org/10.1016/j.envc.2026.101442

Box, G. E. P., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. (2015). *Time series analysis: Forecasting and control* (5th). John Wiley & Sons.

Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5–32. https://doi.org/10.1023/A:1010933404324

Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder–decoder for statistical machine translation. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1724–1734. https://doi.org/10.3115/v1/D14-1179

Cohen, A. J., Brauer, M., Burnett, R., Anderson, H. R., Frostad, J., Estep, K., & Forouzanfar, M. H. (2017). Estimates and 25-year trends of the global burden of disease attributable to ambient air pollution: An analysis of data from the Global Burden of Diseases Study 2015. *The Lancet*, *389*(10082), 1907–1918. https://doi.org/10.1016/S0140-6736(17)30505-6

Diebold, F. X., & Mariano, R. S. (1995). Comparing predictive accuracy. *Journal of Business & Economic Statistics*, *13*(3), 253–263. https://doi.org/10.1080/07350015.1995.10524599

Drewil, G. I., & Al-Bahadili, R. J. (2022). Air pollution prediction using LSTM deep learning and metaheuristics algorithms. *Measurement: Sensors*, *24*, 100546. https://doi.org/10.1016/j.measen.2022.100546

Duan, L.-C., Minh, N.-N., Dung, T.-C., & Linh, T.-T.-T. (2026). Energy-efficient TinyML approach for wearable fall detection on edge devices using spatial-temporal deep learning. *International Journal of Technology*, *17*(3), 919–935. https://doi.org/10.14716/ijtech.v17i3.8445

Elman, J. L. (1990). Finding structure in time. *Cognitive Science*, *14*(2), 179–211. https://doi.org/10.1207/s15516709cog1402_1

Eren, B., Erden, C., Atalı, A., & Ozdemir, S. (2025). A comparative analysis of hyperparameter optimization using LSTM-based deep learning models for urban air quality predictions. *Ain Shams Engineering Journal*, *16*, 103786. https://doi.org/10.1016/j.asej.2025.103786

Escanciano, J. C., & Parra, R. (2026). Extending the scope of inference about predictive ability to machine learning methods [Published online: 14 January 2026]. *Journal of Business & Economic Statistics*, 1–12. https://doi.org/10.1080/07350015.2025.2562964

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, *29*(5), 1189–1232. https://doi.org/10.1214/aos/1013203451

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press. https://www.deeplearningbook.org

Gupta, V., Dash, Y., Goyal, A., Lodh, A., Singh, V., Routray, A., & Kumar, P. (2026). A BiLSTM-driven framework for operational PM2.5 forecasting: Integrating meteorological kinematics for urban air quality management. *Frontiers in Climate*, *8*, 1855755. https://doi.org/10.3389/fclim.2026.1855755

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Hyndman, R. J., & Athanasopoulos, G. (2021). *Forecasting: Principles and practice* (3rd). OTexts. https://otexts.com/fpp3

IQAir. (2024). *World air quality report 2023* (tech. rep.). IQAir. https://www.iqair.com/world-air-quality-report

Keerthana, G., & Ushasree, R. (2024). Air quality prediction using RNN and LSTM. *Journal of Innovation and Technology*, *2024*(48).

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444. https://doi.org/10.1038/nature14539

Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems (NeurIPS)*, *30*, 4765–4774.

Makridakis, S., Spiliotis, E., & Assimakopoulos, V. (2018). Statistical and machine learning forecasting methods: Concerns and ways forward. *PLOS ONE*, *13*(3), e0194889. https://doi.org/10.1371/journal.pone.0194889

Park, S.-Y., Woo, S.-H., & Lim, C. (2023). Predicting PM10 and PM2.5 concentration in container ports: A deep learning approach. *Transportation Research Part D*, *115*, 103601. https://doi.org/10.1016/j.trd.2022.103601

Parntrasri, W., & Puangthongthub, S. (2026). Weekly extremes of PM2.5-attributable cardiovascular mortality in Northeastern Thailand: Extreme-value return levels and 26-week forecasts. *City and Environment Interactions*, *29*, 100313. https://doi.org/10.1016/j.cacint.2026.100313

Patel, P., Patel, S., Shah, K., Desai, K., Patel, S., Shah, M., & Patel, S. (2025). A systematic study on PM2.5 and PM10 concentration prediction in air pollution using machine learning and deep learning model. *Environmental Chemistry and Ecotoxicology*, *7*, 1401–1415. https://doi.org/10.1016/j.enceco.2025.07.001

Pope, C. A., & Dockery, D. W. (2006). Health effects of fine particulate air pollution: Lines that connect. *Journal of the Air & Waste Management Association*, *56*(6), 709–742. https://doi.org/10.1080/10473289.2006.10464485

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "why should I trust you?": Explaining the predictions of any classifier. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1135–1144. https://doi.org/10.1145/2939672.2939778

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural Networks*, *61*, 85–117. https://doi.org/10.1016/j.neunet.2014.09.003

Schuster, M., & Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, *45*(11), 2673–2681. https://doi.org/10.1109/78.650093

Sharma, E., Deo, R. C., Prasad, R., Parisi, A. V., & Raj, N. (2020). Deep air quality forecasts: Suspended particulate matter modeling with convolutional neural and long short-term memory networks. *IEEE Access*, *8*, 244965. https://doi.org/10.1109/ACCESS.2020.3039002

Son, R., Stratoulias, D., Kim, H. C., & Yoon, J.-H. (2023). Estimation of surface PM2.5 concentrations from atmospheric gas species retrieved from TROPOMI using deep learning: Impacts of fire on air pollution over Thailand. *Atmospheric Pollution Research*, *14*, 101875. https://doi.org/10.1016/j.atmosres.2023.101875

Sreenivasulu, T., & Mokesh Rayalu, G. (2025). Accurate hourly AQI prediction using temporal CNN-LSTM-MHA+GRU: A case study of seasonal variations and pollution extremes in Visakhapatnam, India. *Results in Engineering*, *27*, 106303. https://doi.org/10.1016/j.rineng.2025.106303

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems (NeurIPS)*, *30*.

Wang, X., Liu, X., & Mao, W. (2026). Air quality estimation from sequential surveillance images using a unified CNN-RNN framework. *iScience*, *29*, 114919. https://doi.org/10.1016/j.isci.2026.114919

WHO. (2021). *WHO global air quality guidelines: Particulate matter (PM2.5 and PM10), ozone, nitrogen dioxide, sulfur dioxide and carbon monoxide* (tech. rep.). World Health Organization. https://www.who.int/publications/i/item/9789240034228

Zhang, X., & Zhou, P. (2024). A transferred spatio-temporal deep model based on multi-LSTM auto-encoder for air pollution time series missing value imputation. *Future Generation Computer Systems*, *156*, 325–338. https://doi.org/10.1016/j.future.2024.03.015

Zhao, Z., Qin, J., Hu, A., Zhang, N., Xie, J., & Sun, Y. (2026). WaveNet-LSTM: A deep learning air quality prediction model based on both air pollutants and meteorological factors. *Information Sciences*, *730*, 122894. https://doi.org/10.1016/j.ins.2025.122894