SinkrOn
JURNAL & PENELITIAN TEKNIK INFORMATIKA

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Number 2, April 2025
DOI : https://doi.org/10.33395/sinkron.v9i2.14802

e-ISSN : 2541-2019
p-ISSN : 2541-044X

Comparative Study of Forecasting Models for Smart Campus Air

Dodi Dores Pane[1)], Ardito[2)], Enjeliana Sitompul[3)], Nalla Khairani[4)], Marlince NK Nababan[5)]
[1,2,3,4,5)] Universitas Prima Indonesia, Medan, Indonesia
Dodys0310@gmail.com[1)], arditopane5@gmail.com[2)], enjelsitompul5@gmail.com[3)], nallakhairani77@gmail.com[4)] marlince@unprimdn.ac.id[5)]

Submitted :May 21, 2025 | **Accepted** : Jun 10, 2025 | **Published** : Jun 13, 2025

Abstract: Air quality monitoring has become increasingly critical in urban environments, especially in densely populated smart campuses situated in tropical regions. This study presents a comparative evaluation of three predictive models CNN-GRU, LSTM, and Random Forest, for forecasting air pollution levels, specifically particulate matter concentration (PM), using real-time sensor data. The data were collected from an IoT-based monitoring system built with NodeMCU ESP8266 devices deployed on campus. Each model was trained and evaluated using performance metrics including the coefficient of determination (R^2), Mean Absolute Error (MAE), and Root Mean Square Error (RMSE). The results indicate that the Random Forest model achieved the highest predictive accuracy with $R^2 = 0.9073$, MAE = 123.31, and RMSE = 274.45, outperforming both LSTM ($R^2 = 0.8341$) and CNN-GRU ($R^2 = 0.8714$). The hybrid CNN-GRU model, although capable of capturing both spatial and temporal dependencies, required larger data volumes and longer training times. The LSTM model, while effective in modeling time-series data, demonstrated a tendency to overfit when data was limited. This study highlights the practical advantages of Random Forest in modeling complex environmental data under limited resource constraints, while also demonstrating the potential of hybrid deep learning architectures. These findings contribute to the development of efficient air quality prediction systems that support health-conscious decision-making and environmental management strategies in tropical innovative campus environments.

Keywords: Air Quality, CNN-GRU, LSTM, Random Forest

INTRODUCTION

The Earth's atmosphere is composed of gases, known as air, which are vital in supporting life and natural phenomena. Its composition primarily consists of nitrogen (approximately 78%), oxygen (approximately 21%), and small amounts of other gases such as argon, carbon dioxide, neon, helium, methane, hydrogen, and ozone. In addition to these primary gases, air contains water vapor, dust, and pollutants in varying concentrations, depending on environmental conditions, including air pollution. With the advancement of modern technology, the transportation and manufacturing industries have seen an increase in the emission of particulate matter (PM) at high concentrations, posing various health risks to humans. (Chang et al., 2020).

Air pollution is known to cause health impacts on the general public. (Gariazzo et al., 2020)Air pollutants can be produced by human activities or several natural sources, affecting human health and the environment. Solid particles, gases, and liquid droplets can act as pollutants in the air. (Rubal & Kumar, 2018)Gases, particles, fungi, bacteria, and other contaminants can affect indoor air quality. Several sources of pollution, originating from both indoor and outdoor environments, contribute to poor indoor air quality. Outdoor air pollution sources, such as motor vehicle emissions and industrial activities, can enter buildings through doors, windows, and ventilation systems. (Zhu et al., 2023).

It has become a widely held view that air pollution in urban areas has a direct impact on human health, particularly in developing countries and industrialized nations that have not yet taken adequate measures to improve air quality. (Komputer et al., n.d.). The World Health Organization (WHO) estimates that more than 80% of people living in urban areas where air quality is monitored are exposed to air quality levels exceeding WHO guidelines. WHO also estimates that 4.2 million deaths yearly are linked to exposure to outdoor air pollution (Alléon et al., 2020). Therefore, accurately predicting air quality a few hours in advance has become increasingly challenging in recent years. (Otto et al., 2024).

*Dodi Dores Pane

e-ISSN : 2541-2019
p-ISSN : 2541-044X

The study aimed to determine the most effective method for predicting air quality using various machine learning techniques. (Alzu'bi et al., 2024)One of the deep learning models commonly used in time series prediction is the Long Short-Term Memory (LSTM) model. This model can analyze patterns in temporal data and produce more accurate predictions than traditional methods. (Luo & Gong, 2023). LSTM is a machine learning technique designed to understand long-term dependencies in time series data (Drewil & Al-Bahadili, 2022).

Another practical approach for making this prediction is using machine learning algorithms, such as Random Forest. Random Forest (RF) is a regression method that is highly effective for features with non-linear relationships. This algorithm is recognized for producing predictions with high accuracy, minimizing the risk of overfitting, and maintaining robust performance. (Babu & Thomas, 2023). Among the various methods applied, deep learning methods such as neural networks are often used in time series forecasting tasks due to their ability to recognize complex patterns and nonlinear relationships. (Xie et al., 2025).

Although traditional machine learning models have been widely applied in air quality prediction, few studies have explored the hybrid CNN-GRU approach in the context of smart campuses located in tropical urban areas. This study addresses that gap by developing and comparing three predictive models CNN-GRU, LSTM, and Random Forest—using real-time air quality data collected through an IoT-based system installed on campus. The hybrid CNN-GRU model is designed to capture both spatial and temporal patterns in environmental sensor data. At the same time, Random Forest and LSTM serve as benchmark models representing traditional machine learning and time-series-based deep learning approaches, respectively.

This study aims to compare the performance of three prediction models LSTM, Random Forest, and CNN-GRU in forecasting air quality measured using a NodeMCU ESP8266 device in the Universitas Prima Indonesia rectorate building. The accurate results of each model serve as the basis for determining the best model. Based on the performance evaluation, Random Forest demonstrated the best performance with an R^2 value of 0.9073, MAE of 123.31, and RMSE of 274.45. Meanwhile, LSTM achieved an R^2 of 0.8341, MAE of 175.64, and RMSE of 330.98, while CNN-GRU obtained an R^2 of 0.8714, MAE of 176.07, and RMSE of 296.83.

LITERATURE REVIEW

Previous studies on air quality prediction have employed a wide range of machine learning and deep learning approaches, including Support Vector Regression (SVR), Decision Trees, Random Forests, Long Short-Term Memory (LSTM), Gated Recurrent Units (GRU), and hybrid models such as Convolutional Neural Networks (CNN)- LSTM or CNN-GRU. However, few studies have provided a comprehensive comparison in terms of datasets used, model architectures, evaluation results, and the limitations of each method. For instance, the study by (Alléon & Quennehen, n.d.). Utilized Random Forest and XGBoost with multisource data to predict PM2.5, but did not explicitly address temporal aspects. Meanwhile, a CNN-LSTM model was developed for spatiotemporal data; however, it required a large dataset and incurred high computational costs during training. Models like LSTM and GRU are widely used for time-series data due to their ability to capture long-term temporal patterns; however, they tend to overfit when data is limited and require complex parameter tuning. (Learning et al., 2019)

LSTM (Long Short-Term Memory)

LSTM is a branch of recurrent neural networks (RNN). No RNN structure can perform backpropagation over long intervals, so to address this difficulty, LSTM was proposed. LSTM has two gated cell units that open and close the flow of information within each memory cell, which is the information packet between time intervals (Albeladi et al., 2023)The LSTM architecture provides a brief introduction to the basic concepts of RNN and LSTM, and describes the hyperparameters used in the analysis.

One of the main challenges faced in using the Multilayer Perceptron (MLP) architecture is its limitation in recognizing sequential dependency patterns in sequential data. This issue led to the development of Recurrent Neural Networks (RNN). Unlike MLPs, RNNs can capture temporal relationships in sequential data because information from the previous hidden state is passed to the next hidden state. (Aurelius et al., 2021), as illustrated in Figure 1.

SinkrOn
JURNAL & PENELITIAN TEKNIK INFORMATIKA
Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Number 2, April 2025
DOI : https://doi.org/10.33395/sinkron.v9i2.14802
e-ISSN : 2541-2019
p-ISSN : 2541-044X

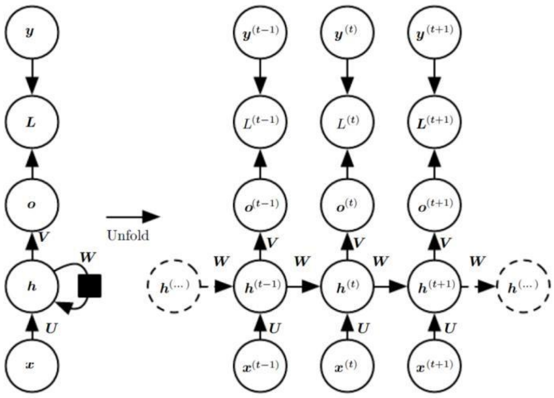

Figure 1. RNN Architecture

Given an input vector sequence $\boldsymbol{x} = (x_1,\ldots, x_T)$, the RNN architecture computes the hidden state vector.

the hidden state sequence $\boldsymbol{h} = (h_1, \ldots , h_T)$ and the output vector sequence $\boldsymbol{y} = (y_1, \ldots , y_T)$ by iterating the following equations from $t = 1$ to T.

$$ht = \mathscr{H}(Wxhxt + Whhht-1 + bh) \quad (1)$$
$$y_t = W_{hy}h_t + b_y \quad (2)$$

Where W represents the weight matrix (e.g. W_{hh} denotes the weight matrix), b represents the bias vector (e.g. b_h denotes the hidden bias vector). $\mathscr{H}$ denotes the activation function used by the RNN layer.

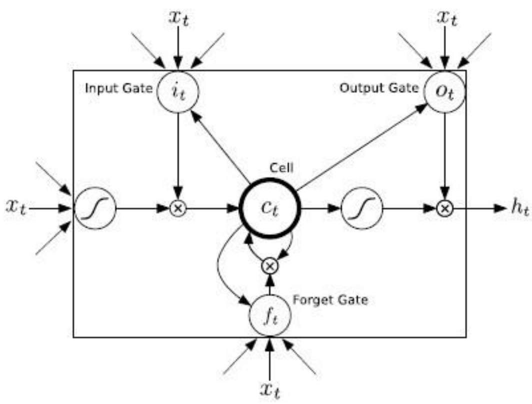

Figure 2. LSTM Memory Cell

The hidden state component is the main difference between LSTM and conventional RNN. While RNN uses a standard hidden state, LSTM utilizes a special memory cell as shown in Figure 3. This memory cell has more parameters and is equipped with a gated system unit that regulates the flow of information. The gating system enables LSTM to overcome the limitations of RNN, such as long-term memory storage issues and the vanishing gradient problem commonly encountered in conventional recurrent neural networks (RNNs). Therefore, LSTM demonstrates superior performance compared to RNN. The equations governing the memory cell in LSTM can be expressed in the form of recursive equations.

$$it = \sigma(Wxixt + Whiht-1 + Wcict-1 + bi) \quad (3)$$
$$ft = \sigma(Wxfxt + Whfht-1 + Wcfct-1 + bf) \quad (4)$$
$$ct = ftct-1 + it\sigma(Wxcxt + Whcht-1 + Wccct-1 + bc) \quad (5)$$
$$ot = \sigma(Wxoxt + Whoht-1 + Wcoct-1 + bo) \quad (6)$$
$$h_t = o_t \tanh(c_t) \quad (7)$$

*Dodi Dores Pane

Where σ denotes the sigmoid function, W represents the weight matrix, and i_t, f_t, c_t, o_t, h_t, respectively, represent the input gate, forget gate, cell state, output gate, and hidden vector at time t.

CNN- GRU (Convolutional Neural Network - Gated Recurrent Unit)

CNN-GRU is a deep learning architecture that combines Convolutional Neural Networks (CNNs) for feature extraction and Gated Recurrent Units (GRUs) for managing sequential data, thereby enabling a more efficient understanding of both spatial and temporal patterns. In this context, CNN is the initial layer that extracts important features from the input data. Convolutional Neural Networks (CNNs) are a critical type of network in deep learning, developed as a variant of multilayer perceptrons and inspired by the concept of artificial neural networks. (Li & Huo, 2023). Meanwhile, GRU (Gated Recurrent Unit) is a type of recurrent neural network with a more compact design than LSTM, as it does not have a separate memory cell component. This model is computationally more efficient while still delivering performance similar to that of LSTM. The GRU structure consists of two main mechanisms: the update gate and the reset gate, where the update gate manages past information that should be retained or discarded during sequential data processing to produce more accurate predictions. (Wang et al., 2023).

Operation CNN

CNN performs a convolution operation on the input data to produce a feature map as follows:

Ft=ReLU(Wc∗Xt+bc)

Ft is the feature map at time t, Wc is the convolutional filter, Xt_t is the input data at time t, bc is the convolutional bias, and * denotes the convolution operation.

Operation GRU

The feature map generated by the CNN is then processed by the GRU layer using the following equations:

Reset Gate	$= \sigma(Wr \cdot Ft + Ur \cdot ht - 1 + br)$
Update Gate	$t = \sigma(Wz \cdot Ft + Uz \cdot ht - 1 + bz)$
Candidate Hidden State	$\sim ht = tanh(Wh \cdot Ft + Uh \cdot (rt \odot ht - 1) + bh)$
Current Hidden State	$ht = (1 - zt) \odot ht - 1 + zt \odot h\sim t$

rt reset gate, zt update gate, $h\sim t$ candidate hidden state, ht_t hidden state at time t, **W** is the input weight matrix, **U** is the hidden state weight matrix, **b** is the bias, and ⊙ denotes the element-wise (Hadamard) product.

Random Forest

This method falls under the ensemble supervised learning approach, utilizing decision trees as its fundamental components. It is highly versatile and can be applied to various tasks, including classification and regression, and is known for its efficiency in training and prediction processes. (Joharestani et al., 2019).
The following is the formula used in the Random Forest algorithm to produce the final prediction, which is based on the aggregation of multiple decision trees.

$$\hat{Y} = \frac{1}{M} \sum_{m=1}^{M} \mathrm{f_m}(\mathrm{X})$$

Y^Is the final predicted PPM value, MIs the number of trees in the forest, and $fm(X)$Is the prediction of the m Decision tree for the input X.

The Random Forest formula works by training each tree on a randomly selected subset of the data with replacement (bootstrapping), ensuring each tree has a unique perspective and reducing overfitting. It uses a random subset of features to form decisions at each node. It combines the final results as the average of all three predictions, producing a more accurate, stable, and robust model that is less susceptible to noise.

METHOD

The selection of the CNN-GRU architecture in this study is based on its advantages in handling spatiotemporal data, which is commonly found in air quality sensor data. Convolutional Neural Networks (CNNs) are effective in extracting spatial features from input data. At the same time, Gated Recurrent Units (GRUs) are capable of capturing short- to mid-term temporal dependencies with greater computational efficiency compared to LSTM. This hybrid architecture is expected to combine the strengths of both models, thereby enhancing prediction accuracy. This study hypothesizes that the CNN-GRU model will achieve predictive performance equal to or better

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Number 2, April 2025
DOI : https://doi.org/10.33395/sinkron.v9i2.14802
e-ISSN : 2541-2019
p-ISSN : 2541-044X

than that of the LSTM and Random Forest models, particularly in processing real-time, sequential sensor data with complex patterns. (Imrana et al., 2024)

IoT Architecture for Air Quality Monitoring

The Internet of Things (IoT) is a network infrastructure enabling various objects with unique identities to connect. These objects generally consist of sensors or actuators equipped with intelligent capabilities. Through this connectivity, data from the devices can be accessed and collected. Their status or operations can be controlled flexibly from various locations, at any time, and using different types of devices (Múnera et al., 2021). The NodeMCU ESP8266 is used in this model, a Wi-Fi-based microcontroller commonly employed in IoT application development. This device supports open programming, allowing users to create, modify, and rebuild programs and project functions flexibly across various programming platforms (Messan et al., 2022). The NodeMCU ESP8266 is equipped with two types of sensors: the MQ-135, which detects gas concentration in parts per million (PPM), and the DHT11, used to measure environmental parameters such as temperature and relative humidity.

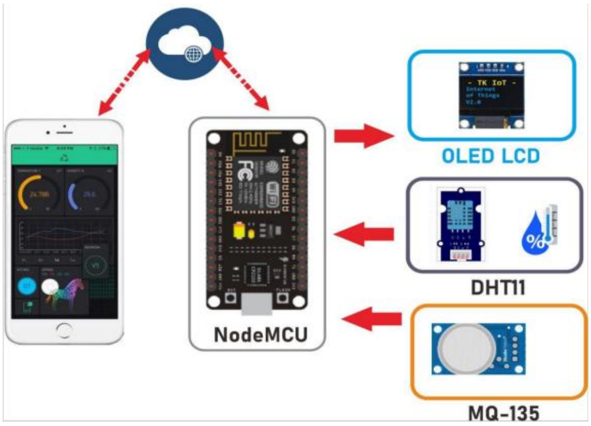

Figure 3. IoT System Design Architecture

Data Source

The data used in this study is primary data, namely air quality data obtained directly using a NodeMCU device in the environment of Universitas Prima Indonesia, specifically on the 10th floor, in the rector's office, and in open areas. 1,669 datasets were collected, consisting of the attributes created_at, entry_id, PPM, Temperature, Humidity, and Status. The attribute used for air quality prediction is PPM. Data was collected by connecting the NodeMCU to ThingSpeak, one of the Internet of Things (IoT) platforms used to monitor air quality in real time.

created_at	entry_id	PPM	TEMPERATUR	HUMIDITY	STATUS
2025-05-02T08:21:25+00:00	402	1574,1886	35,2	53	Poor
2025-05-02T08:21:47+00:00	403	1551,78491	35,2	53	Poor
2025-05-02T08:22:11+00:00	404	1551,78491	35,2	54	Poor
2025-05-02T08:22:33+00:00	405	1574,1886	35,2	54	Poor
2025-05-02T08:22:55+00:00	406	1619,79358	35,2	54	Poor
2025-05-02T08:23:16+00:00	407	1619,79358	35,2	54	Poor
2025-05-02T08:23:37+00:00	408	1574,1886	35,2	54	Poor
2025-05-02T08:23:59+00:00	409	1643,00037	35,2	54	Poor
2025-05-02T08:24:20+00:00	410	1643,00037	35,2	54	Poor
2025-05-02T08:24:42+00:00	411	1643,00037	35,2	54	Poor
2025-05-02T08:25:04+00:00	412	1643,00037	35,2	54	Poor
2025-05-02T08:25:25+00:00	413	1619,79358	35,2	54	Poor
2025-05-02T08:25:47+00:00	414	1507,76306	35,2	54	Poor
2025-05-02T08:26:08+00:00	415	1245,79834	35,2	54	Poor
2025-05-02T08:26:30+00:00	416	1690,23621	35,2	54	Poor
2025-05-02T08:26:52+00:00	417	1574,1886	35,2	54	Poor
2025-05-02T08:27:13+00:00	418	1619,79358	35,2	54	Poor
2025-05-02T08:27:35+00:00	419	1574,1886	35,2	54	Poor
2025-05-02T08:27:56+00:00	420	1619,79358	35,2	54	Poor
2025-05-02T08:28:18+00:00	421	1574,1886	35,2	54	Poor
2025-05-02T08:28:39+00:00	422	1507,76306	35,2	54	Poor
2025-05-02T08:29:00+00:00	423	1529,64441	35,2	54	Poor
2025-05-02T08:29:22+00:00	424	1486,13904	35,2	54	Poor
2025-05-02T08:29:43+00:00	425	1507,76306	35,2	54	Poor
2025-05-02T08:30:04+00:00	426	1529,64441	35,2	54	Poor
2025-05-02T08:30:26+00:00	427	1619,79358	35,2	54	Poor

Figure 4. Air Quality Data

*Dodi Dores Pane

SinkrOn
JURNAL & PENELITIAN TEKNIK INFORMATIKA
Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Number 2, April 2025
DOI : https://doi.org/10.33395/sinkron.v9i2.14802
e-ISSN : 2541-2019
p-ISSN : 2541-044X

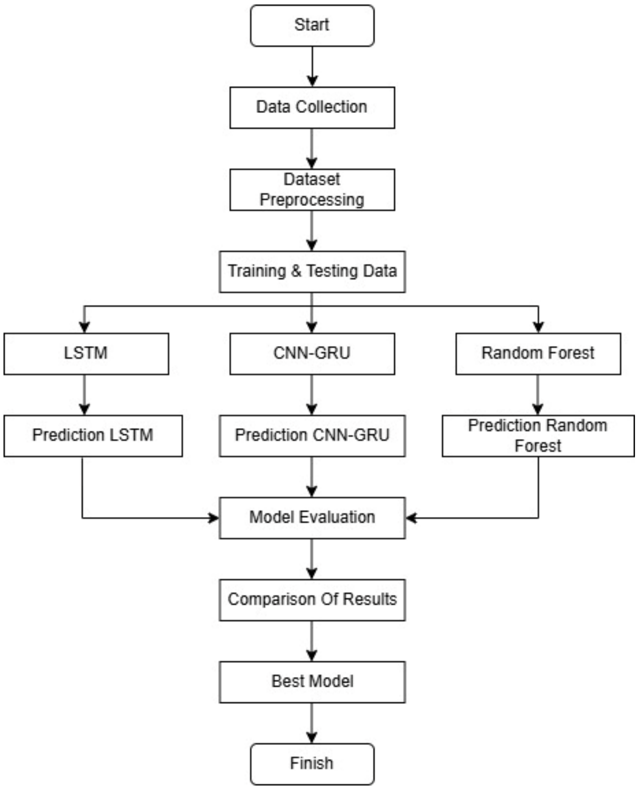

Figure 5. Flowchart Method

RESULT

This study evaluates three machine learning models — LSTM (Long Short-Term Memory), CNN-GRU (Convolutional Neural Network-Gated Recurrent Unit), and Random Forest — to predict PPM (Parts Per Million) values based on sensor data. Each model is evaluated using R^2, MAE, and RMSE metrics to assess how well the model predicts PPM values. The following are the results and performance analysis of each model:

The LSTM model is a recurrent neural network to understand long-term dependencies in time series data. This test achieved an R^2 value of 0.8341, MAE of 175.64, and RMSE of 330.98. Although it can recognize temporal patterns well, this model tends to be sensitive to sudden changes in data patterns, resulting in higher error rates in some cases.

The LSTM model processes data by removing PPM outliers using the IQR method to improve data quality, extracting time features such as hour and minute from the time column to add temporal context, and calculating the rolling average PPM to reduce noise and capture local trends. Then, it creates a time series dataset with a window size of 30 using the create_dataset() function to reflect temporal dependencies between data points. The model is trained with a two-layer LSTM with dropout (the first layer LSTM (128, return_sequences=True) to capture sequential relationships and the second layer LSTM (64) for final decision-making) using the Nadam optimizer, which is suitable for small-scale data, and early stopping to prevent overfitting by halting training if validation loss does not improve within 15 epochs. The model predicts on the test data to measure performance on new data, returns the prediction scale to the original values for more straightforward interpretation, calculates evaluation metrics r2_score to measure model fit, mean_absolute_error to observe average absolute error, and rmse to calculate average error more sensitive to outliers, and visualizes actual vs predicted results for visual evaluation and displays a comparison table of actual and predicted values for further analysis, as shown in Figure 6.

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Number 2, April 2025
DOI : https://doi.org/10.33395/sinkron.v9i2.14802
e-ISSN : 2541-2019
p-ISSN : 2541-044X

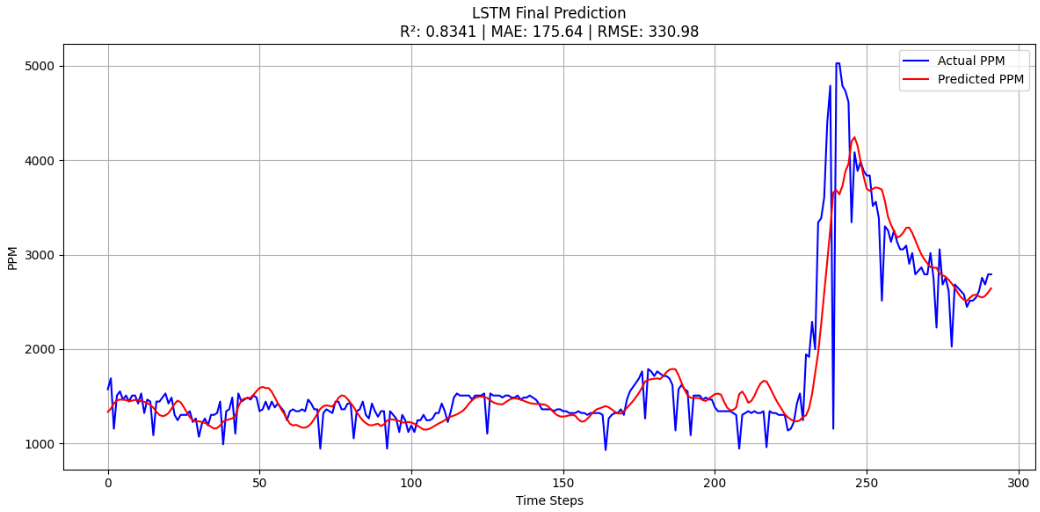

Figure 6. LSTM Prediction

Actual PPM	Predicted PPM
1574.19	1420.34
1690.24	1456.26
1155.48	1505.93
1507.76	1529.20
1551.78	1526.96
1464.77	1516.72
1507.76	1498.64
1443.65	1472.51
1507.76	1456.35
1507.76	1442.95
1422.78	1426.34
1529.64	1405.89
1322.10	1402.79
1464.77	1396.78
1443.65	1388.79

Table 1. Sample comparison LSTM (Actual vs Predicted PPM)

The data above is a comparison between Actual PPM (the gas concentration measured directly and used as the target in model training) and Predicted PPM (the gas concentration predicted by the LSTM model based on input data or features) based on 15 prediction results, where the differences between these two values are used to calculate evaluation metrics such as R^2 Score, MAE (Mean Absolute Error), and RMSE (Root Mean Squared Error) to assess how well the model predicts the PPM values.

In this study, a deep learning model was developed to predict gas concentration in PPM by integrating CNN and GRU. The process began with data collection and cleaning to ensure input quality. Important features were extracted from the data by considering temporal patterns and statistical characteristics. The data was then normalized using MinMaxScaler to bring it to a uniform scale. The data was arranged sequentially using a look-back window technique to capture temporal dependencies. During training, the model had control mechanisms such as EarlyStopping and ReduceLROnPlateau to prevent overfitting and maintain optimal model performance. Model evaluation was conducted using metrics R^2, MAE, and RMSE, resulting in an R^2 value of 0.8774, MAE of 164.62, and RMSE of 289.81. The predictions were also visualized alongside the actual data to facilitate visual assessment of the model's accuracy level, as shown in Figure 7.

*Dodi Dores Pane

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Number 2, April 2025
DOI : https://doi.org/10.33395/sinkron.v9i2.14802
e-ISSN : 2541-2019
p-ISSN : 2541-044X

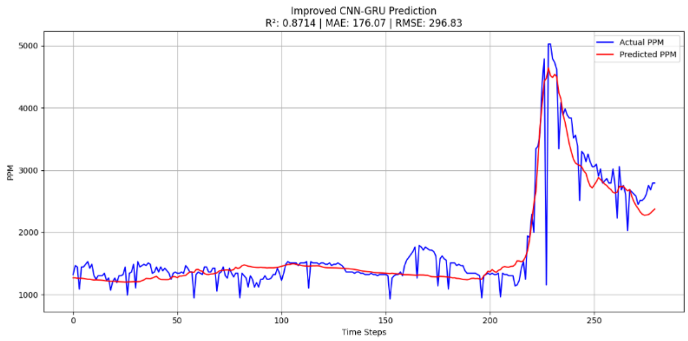

Figure 7. Comparison of Actual and Predicted Values of the CNN-GRU Model

Actual PPM	Predicted PPM
1322.10	1291.949951
1464.77	1292.859985
1443.65	1291.709961
1087.09	1294.040039
1443.65	1291.050049
1443.65	1288.680054
1486.14	1279.949951
1529.64	1266.479980
1422.78	1266.489990
1486.14	1265.770020
1302.68	1263.109985
1245.80	1269.959961
1302.68	1280.780029
1302.68	1287.439941
1302.68	1288.729980

Table 2. Actual vs Predicted PPM CNN-GRU

The presented data compares the actual PPM values, which represent gas concentrations obtained from direct measurements, and the predicted PPM values generated by the CNN-GRU model based on the analyzed input variables. The differences between these two values serve as the basis for calculating evaluation metrics such as the R^2 Score, MAE (Mean Absolute Error), and RMSE (Root Mean Squared Error). These indicators evaluate the model's ability to accurately and consistently predict gas concentrations.

Random Forest processes the data by removing outliers, followed by feature engineering that involves adding time-based and statistical variables, such as hour, minute, rolling averages, and the exponential moving average (EMA). The data is then split into features (X) and target (y), and the features are normalized using MinMaxScaler. The dataset is divided into training and testing sets. Hyperparameter tuning is conducted using RandomizedSearchCV on the RandomForestRegressor with a random distribution of parameters. The best-performing model is then trained and evaluated using performance metrics, achieving an R^2 Score of 0.907, an MAE of approximately 123.31, and an RMSE of 274.45. The prediction results are compared to the actual values and presented in tabular form and line charts to facilitate model performance analysis, as shown in Figure 8.

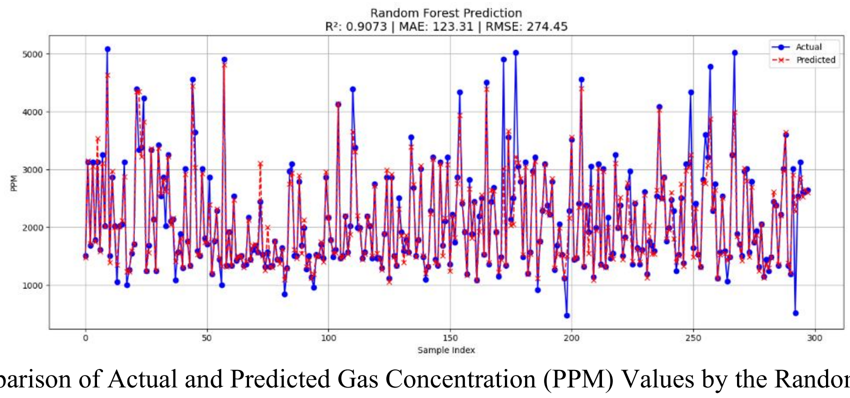

Figure 8. Comparison of Actual and Predicted Gas Concentration (PPM) Values by the Random Forest Model

*Dodi Dores Pane

SinkrOn
JURNAL & PENELITIAN TEKNIK INFORMATIKA
Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Number 2, April 2025
DOI : https://doi.org/10.33395/sinkron.v9i2.14802
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Actual PPM	Predicted PPM
1507.76306	1476.636044
3135.41211	3147.672536
1690.23621	1722.128282
3135.41211	3055.782000
1788.07654	1751.883989
3135.41211	3545.820460
1619.79358	1580.353839
3257.96973	3112.939011
2025.48621	2013.525155
5086.81592	4632.135766
1507.76306	1391.930224
2864.70190	2973.011121
2025.48621	2007.293120
1054.14087	1356.323010
2025.48621	2016.130665

Table 3. The actual vs predicted Random Forest row in the table above represents a single measurement sample,

where the predicted value is expected to approximate the actual value closely. The model's accuracy in replicating or predicting the actual values can be observed by comparing these two columns. The difference or gap between the values serves as an indicator of prediction accuracy. In general, if the predicted PPM values are close to the actual PPM values, the model can be considered to have good performance in forecasting gas concentration.

Model evaluation

Model evaluation is crucial for assessing the effectiveness of the approach in predicting PPM values. At this stage, various evaluation metrics such as R^2, MAE, and RMSE are employed to measure the accuracy and consistency of the prediction results. These three metrics are chosen because they provide a comprehensive overview of how well the model captures data patterns, the level of absolute error, and the distribution of prediction errors. The following table presents a performance comparison of the three main models used in this study.

Table 4. Model Evaluation

Model	R^2	MAE	$RMSE$
Random Forest	**0,9073**	**123,31**	274,45
LSTM	0,8341	175,64	330,98
CNN-GRU	0,8714	176,07	296,83

The Random Forest regression model demonstrated excellent performance, with an R-squared value of 0.9073, indicating that it can explain approximately 90.73% of the variance in the data. Additionally, the model achieved an average prediction error of 123.31 (mean absolute error, MAE) and 274.45 (root mean squared error, RMSE).

DISCUSSIONS

This study compares three algorithms CNN-GRU, LSTM, and Random Forest applied to predict air quality parameters (PPM). Based on the evaluation results, Random Forest demonstrated the best performance with the highest R^2 value and the lowest prediction error compared to CNN-GRU and LSTM. The Random Forest method, an ensemble technique based on decision trees, can handle data with high complexity and nonlinear interactions among features. Its advantage lies in reducing the risk of overfitting and providing more interpretable results than deep learning models. Conversely, the hybrid CNN-GRU model combines Convolutional Neural Networks (CNN) for spatial feature extraction and Gated Recurrent Units (GRU) for capturing temporal patterns. However, effective in recognizing spatiotemporal patterns, this model requires a large dataset and considerable training time. Meanwhile, LSTM, a variant of Recurrent Neural Networks (RNNs) designed to model long-term dependencies in time series data, has higher complexity and requires more complicated parameter tuning than Random Forest.

Several previous studies have also confirmed the superiority of Random Forest in the context of air quality prediction. (He & Guo, 2024) The combined CNN-GRU-LSTM model achieved higher accuracy than the individual CNN, GRU, and LSTM models; however, on smaller datasets with complex features, Random Forest tended to deliver more stable and accurate results. Similarly,(Iqbal et al., 2024) A study comparing several models for predicting the Air Quality Index (AQI) in major Indian cities found that Random Forest outperformed others in terms of MAE and RMSE values. Random Forest excels at handling complex and nonlinear feature relationships, which are commonly found in air quality sensor data.

Nevertheless, this study also faces several limitations and potential threats to the validity of the results. Among these are the limited size and variation of the dataset, which only covers specific locations and periods; hence, the results may not be fully generalizable to other geographic regions or temporal contexts. Furthermore, deep learning models such as CNN-GRU and LSTM are prone to overfitting, particularly when the available data is limited and regularization techniques are not applied adequately. Lastly, although Random Forest is more straightforward to interpret, deep learning models possess complex structures that make it challenging to provide a transparent explanation for decision-making.

CONCLUSION

Random Forest emerged as the best-performing model based on the accuracy evaluation of various air quality prediction models using PPM attributes. This is evidenced by its R^2 score of 0.9073, an MAE of 123.31, and an RMSE of approximately 274.45, outperforming both the LSTM and CNN-GRU models. The LSTM model produced an R^2 of 0.8341, with higher MAE and RMSE values of 175.64 and 330.98, respectively. Meanwhile, the CNN-GRU model achieved intermediate results, with an R^2 of 0.8714, a mean absolute error (MAE) of 176.07, and a root mean squared error (RMSE) of 296.83. Therefore, Random Forest is recommended as the primary model for predicting air quality based on PPM attributes within the context of this study. Further research may also focus on deploying the model on edge devices for real-time monitoring and validating its performance across different locations to ensure broader applicability.

REFERENCES

Albeladi, K., Zafarbahasa, B., & Ahmad, I. (2023). *Peramalan Deret Waktu menggunakan LSTM dan ARIMA*. *1*, 313–320.

Alléon, A., Jauvion, G., Quennehen, B., & Lissmyr, D. (2020). *PlumeNet: Large-Scale Air Quality Forecasting Using A Convolutional LSTM Network*. http://arxiv.org/abs/2006.09204

Alléon, A., & Quennehen, B. (n.d.). *PlumeNet : Large-Scale Air Quality Forecasting Using A Convolutional LSTM Network*.

Alzu'bi, F., Al-Rawabdeh, A., & Almagbile, A. (2024). Predicting air quality using random forest: A case study in Amman-Zarqa. *Egyptian Journal of Remote Sensing and Space Science*, *27*(3), 604–613. https://doi.org/10.1016/j.ejrs.2024.07.004

Aurelius, R., Viadinugroho, A., & Jurusan, D. R. (2021). *Jurnal Fisika : Rangkaian Konferensi Model Jaringan Syaraf Tiruan Memori Jangka Panjang dan Jangka Pendek untuk Peramalan Deret Waktu : Studi Kasus Peramalan IHSG pada Masa Wabah Covid-19 Model Jaringan Syaraf Tiruan Memori Jangka Panjang dan Jangka Pende*.

Babu, S., & Thomas, B. (2023). A survey on air pollutant PM2.5 prediction using random forest model. *Environmental Health Engineering and Management*, *10*(2), 157–163. https://doi.org/10.34172/EHEM.2023.18

Chang, Y. S., Chiao, H. T., Abimannan, S., Huang, Y. P., Tsai, Y. T., & Lin, K. M. (2020). An LSTM-based aggregated model for air pollution forecasting. *Atmospheric Pollution Research*, *11*(8), 1451–1463. https://doi.org/10.1016/j.apr.2020.05.015

Drewil, G. I., & Al-Bahadili, R. J. (2022). Air pollution prediction using LSTM deep learning and metaheuristics algorithms. *Measurement: Sensors*, *24*(September), 1–7. https://doi.org/10.1016/j.measen.2022.100546

Gariazzo, C., Carlino, G., Silibello, C., Renzi, M., Finardi, S., Pepe, N., Radice, P., Forastiere, F., Michelozzi, P., Viegi, G., & Stafoggia, M. (2020). A multi-city air pollution population exposure study: Combined use of chemical-transport and random-Forest models with dynamic population data. *Science of the Total Environment*, *724*. https://doi.org/10.1016/j.scitotenv.2020.138102

He, Z., & Guo, Q. (2024). Comparative Analysis of Multiple Deep Learning Models for Forecasting Monthly Ambient PM2.5 Concentrations: A Case Study in Dezhou City, China. *Atmosphere*, *15*(12). https://doi.org/10.3390/atmos15121432

Imrana, Y., Xiang, Y., Abdullah, M. A., Ali, L., Noor, A., & Sarpong, K. (2024). CNN-GRU-FF : a double-layer feature fusion-based network intrusion detection system using convolutional neural network and gated recurrent units. *Complex & Intelligent Systems*, *10*(3), 3353–3370. https://doi.org/10.1007/s40747-023-01313-y

Iqbal, A., Sarkar, P., & Mukherjee, N. (2024). *Comparative Analysis of Machine Learning Models for AQI Prediction in Indian Metro Cities*. *13*(12).

Joharestani, M. Z., Cao, C., Ni, X., Bashir, B., & Talebiesfandarani, S. (2019). PM2.5 prediction based on random forest, XGBoost, and deep learning using multisource remote sensing data. *Atmosphere*, *10*(7), 12–18. https://doi.org/10.3390/atmos10070373

Komputer, I., Rivers, U. N., & Harcourt, P. (n.d.). *Kata kunci : Indeks Kualitas Udara , Partikel , Model LSTM*

Dua Arah , Parameter Metrologi.

Learning, D., Multisource, U., & Data, S. (2019). *Sensing Data*. *2*, 12–18.

Li, X., & Huo, H. (2023). *Prediksi Konsentrasi PM2 . 5 Berdasarkan Model CNN - BiGRU*. *3*, 1–8.

Luo, J., & Gong, Y. (2023). Air pollutant prediction based on ARIMA-WOA-LSTM model. *Atmospheric Pollution Research*, *14*(6). https://doi.org/10.1016/j.apr.2023.101761

Messan, S., Shahud, A., Anis, A., Kalam, R., Ali, S., & Aslam, M. I. (2022). Air-MIT: Air Quality Monitoring Using Internet of Things †. *Engineering Proceedings*, *20*(1). https://doi.org/10.3390/engproc2022020045

Múnera, D., Diana, V., Aguirre, J., & Gómez, N. G. (2021). IoT-based air quality monitoring systems for smart cities: A systematic mapping study. *International Journal of Electrical and Computer Engineering*, *11*(4), 3470–3482. https://doi.org/10.11591/ijece.v11i4.pp3470-3482

Otto, P., Fusta Moro, A., Rodeschini, J., Shaboviq, Q., Ignaccolo, R., Golini, N., Cameletti, M., Maranzano, P., Finazzi, F., & Fassò, A. (2024). Spatiotemporal modelling of PM 2.5 concentrations in Lombardy (Italy): a comparative study. *Environmental and Ecological Statistics*, *31*(2), 245–272. https://doi.org/10.1007/s10651-023-00589-0

Rubal, & Kumar, D. (2018). Evolving Differential evolution method with random forest for prediction of Air Pollution. *Procedia Computer Science*, *132*, 824–833. https://doi.org/10.1016/j.procs.2018.05.094

Wang, Z., Huang, H., Du, R., Li, X., Yuan, G., Ilmu, F., & Sains, U. (2023). *Model Deteksi Intrusi IoT berdasarkan CNN-GRU*. *2*, 90–95.

Xie, C., Xue, B., Yang, M., Zhang, M., Xu, Z., Wang, J., Chen, S., & Liu, Y. (2025). Evolutionary neural architecture search for automatically designing CNN-GRU-Attention neural networks for turntable servo systems. *Expert Systems with Applications*, *283*(April). https://doi.org/10.1016/j.eswa.2025.127765

Zhu, Y., Al-ahmed, S. A., Zeeshan, M., & Isabelle, J. (2023). *CO Berbasis IoT Berbasis LSTM 2 Peramalan Kondisi Tetap untuk Pemantauan Kualitas Udara Dalam Ruangan*.