MJFAS
Malaysian Journal of Fundamental and Applied Sciences

RESEARCH ARTICLE

Comparison Between LSTM, GRU and VARIMA in Forecasting of Air Quality Time Series Data

Yu Nie Ng[a], Han Ying Lim[b], Ying Chyi Cham[c], Mohd Aftar Abu Bakar[a]*, Noratiqah Mohd Ariff[a]

[a]Department of Mathematical Sciences, Faculty of Science and Technology, Universiti Kebangsaan Malaysia, 43600 Bangi, Selangor, Malaysia; [b]Institute of Mathematical Sciences, Faculty of Science, Universiti Malaya, 50603 Kuala Lumpur, Malaysia; [c]Faculty of Computer Science and Information Technology, Universiti Malaya, 50603 Kuala Lumpur, Malaysia

Abstract Air quality forecast is essential in alerting the public, especially those who have respiratory diseases, to take necessary precautions beforehand. The public can be forewarned of any worsening of air quality and be aware of the importance of reducing air pollution. In recent years, forecasting techniques based on deep learning algorithms such as recurrent neural network (RNN) have seen improvements in both accuracy and execution speed. Long short-term memory (LSTM) network and gated recurrent unit (GRU) are among the most popular variants of RNN. In this study, the hourly $PM_{2.5}$ concentrations at five selected air quality monitoring stations, provided by the Department of Environment Malaysia, are forecasted using LSTM, GRU and vector autoregressive integrated moving average (VARIMA) models respectively. Data containing missing, negative and zero values are pre-processed using an interpolation technique before being split into training and test sets on an 80:20 ratio basis. Optimal combinations of hyperparameter values are selected via manual tuning based on the 10-fold growing window cross-validation results. The model performance is evaluated based on RMSE, MAE and MAPE. The results demonstrate that neural network models significantly outperform the multivariate time series model in which the LSTM and GRU models have comparable performance in forecasting the hourly $PM_{2.5}$ concentration, with a slightly better prediction in the west coast region for LSTM and the east coast region for GRU. However, due to the complex architecture of neural networks, the computational time to train both LSTM and GRU models is three times longer than that for VARIMA. Additionally, it is observed that a higher percentage of interpolated values leads to lower prediction errors.

Keywords: Air quality, long short-term memory (LSTM), gated recurrent unit (GRU), vector autoregressive integrated moving average (VARIMA), forecasting.

***For correspondence:**
aftar@ukm.edu.my

Received: 25 Jan. 2024
Accepted: 16 Oct. 2024

Introduction

In recent years, air quality has become a critical issue concerning people around the world due to the massive increase in air pollutants caused by rapid urbanisation and industrialisation. Numerous studies have revealed that air pollutants such as ozone and particulate matter could be hazardous to human health, causing chronic illnesses including lung cancer as well as respiratory and cardiovascular diseases [1]. Generally, air pollution can be defined as contamination of the environment by any chemical, physical or biological agent that changes the natural characteristics of the atmosphere [1]. According to the data released by the World Health Organization (WHO), nine out of ten people breathe in polluted air, thus leading to 7 million people being killed annually [2].

In Malaysia, poor air quality is mainly contributed by gas exhaustion from vehicle emissions, haze caused by weather and forest fires in the neighbouring country and air pollutants released by industrial activities [3]. Acknowledging the harmful impacts of air pollution, the Department of Environment Malaysia established the Ambient Air Quality Standard with amended limits for six air pollutants including the particulate matter of 2.5 micrometres or less in diameter which is commonly known as $PM_{2.5}$. Starting from the year 2020 onwards, the average concentration of $PM_{2.5}$ is strictly limited to 15 μg/m^3 each year [4].

The tiny $PM_{2.5}$ particles are most likely to be elevated on windless days [5]. These fine particles can easily penetrate deeply into the lung, causing lung irritation, coughing and shortness of breath, consequently impairing lung function [5,6]. Xing *et al*. [6] have proven that $PM_{2.5}$ can damage the human respiratory system through a few mechanisms such as injury from free radical peroxidation, imbalanced intracellular calcium homeostasis and inflammatory injury.

An accurate air quality forecast that revolves around the level of $PM_{2.5}$ is essential in alerting the public, especially those with compromised health. Those who are vulnerable to poor air quality could take necessary precautions beforehand, such as wearing a face mask and avoiding outdoor activities. Forecasting is also deemed important for the government and relevant authorities to be forewarned of any worsening of air quality so as to implement effective measures in controlling the emission of air pollutants.

In recent decades, various forecasting techniques have been adopted to predict air quality. Some studies were done by using conventional statistical methods including regression model and autoregressive integrated moving average (ARIMA) model, while some researchers proposed deep learning algorithms such as recurrent neural network (RNN) and long short-term memory (LSTM) network to obtain a more precise prediction [7]. Due to the unexpected frequent changes in $PM_{2.5}$ level, Caraka ***et al***. [8] used state Markov chain stochastic process to determine the spreading pattern of $PM_{2.5}$ in Pingtung and Chaozhou. Having classified the $PM_{2.5}$ transition into three risk categories, the Markov chain was used to calculate the probability of changes among the three categories for the upcoming month. A hybrid vector autoregressive, neural network and particle swarm optimisation model (VAR-NN-PSO) was then used to forecast the $PM_{2.5}$ for the next 180 days. On the other hand, Zhou ***et al***. [9] forecasted $PM_{2.5}$ concentration in Beijing during the four seasons using GRU. Having seven input variables with optimal hyperparameter values, the model was proved to be effective in forecasting the $PM_{2.5}$ accurately for readings below 600 μg/m^3.

In addition, LSTM which is capable of learning long-term dependencies, often gives better accuracy in forecasting time series data than the conventional statistical models [10]. Such strength encourages the researchers to use it in various areas of study including meteorology, economy and disease prediction. Uh and Majid [11] found that LSTM outperformed ARIMA in forecasting the daily gold prices. Similarly, in a comparison of influenza-like illness (ILI) and respiratory disease prediction using LSTM and ARIMA done by Tsan *et al*. [12], the results showed that ARIMA predicted more accurately for the five-year dataset whereas LSTM performed better on average for a longer historical timeframe of ten years. Generally, LSTM outperformed the ARIMA model up to seven times in terms of model performance, proving its strength in learning long-term patterns.

The application of deep learning algorithms can be extended to the field of ionosphere monitoring. By using vertical total electron content (VTEC) data recorded by the Global Positioning System (GPS), research done by Tan *et al*. [13] showed that LSTM could model the time series more accurately than GRU. Seeing the potential of neural networks in overcoming the drawbacks of traditional time series models, Mateus *et al*. [14] modelled the daily pulp paper press time series which has six monitored variables and predicted the future sensor values in 30 days using LSTM and GRU. Splitting the dataset into training and test sets based on a 70:30 ratio, the researchers carried out experiments to determine the best combination of hyperparameter values by using different window sizes, resampling rates, layer sizes and activation functions. From the thorough comparative analysis, GRU performed better than LSTM.

While complex deep learning models often achieve superior forecasting performance, simple statistical models can be a valuable starting point due to their ability to provide interpretable and satisfying predictions in certain circumstances. For instance, ArunKumar *et al*. [15] optimised the parameters of ARIMA, seasonal autoregressive integrated moving average (SARIMA), LSTM and GRU using an automated function during the forecasting of the country-wise COVID-19 trends in cumulative confirmed, recovered and deaths. Although the deep learning-based models outperformed the statistical ARIMA and SARIMA models for most of the time series, the classical models did perform better in some countries. VARIMA, being the vector form of the ARIMA, achieves forecasting of higher accuracy by considering the influence of other variables [16]. Setiawan *et al*. [17] used VARIMA and generalised space-time autoregressive integrated moving average (GSTARIMA) to forecast monthly inflation at six capitals in Java Island. Training with data from all six locations simultaneously, the study found that the best VARIMA model was VAR (1) with dummy variables added to handle the overfitting issue.

Aiming to build a time series forecasting model that can achieve high precision in predicting the hourly PM$_{2.5}$ concentration for selected stations in Malaysia, a comparison of the model performance between LSTM, GRU and VARIMA has been carried out based on the accuracy metrics, namely root mean square error (RMSE), mean absolute error (MAE) and mean absolute percentage error (MAPE). In order to achieve higher accuracy, 10-fold growing window cross-validation technique is adopted in this study.

Materials and Methods

Data

In this study, the dataset provided by the Department of Environment Malaysia consists of 13080 rows of hourly PM$_{2.5}$ concentration data that were recorded by five air quality monitoring stations located at either industrial or urban areas in Malaysia, starting from 5 July 2017 to 31 December 2018.

The data contains missing, negative and zero values, of which should not exist in principle. Machine defects and human errors may result in missing values in air quality data [18]. Negative values in PM$_{2.5}$ concentration, which are infrequently recorded by PM$_{2.5}$ instruments, could be due to instrument flaws and technological limitations, as the PM$_{2.5}$ concentration of urban cities are in general above zero level [19]. Of various interpolation techniques, the monotonic piecewise cubic Hermite interpolating polynomial (PCHIP) is chosen to approximate these values. Gariazzo ***et al***. [20] used this method to parameterise the primordial power spectrum as it can avoid spurious oscillations of the interpolated function between the nodes, unlike the spline interpolations.

The missing and negative values are first replaced by zeros. Then, PCHIP is used to interpolate and replace those zeros to ensure the non-zeroness and nonnegativity of the data. The effectiveness of PCHIP in handling successive zeros is shown in Figure 1 by using part of the data extracted from the Kulim Hi-Tech station, in which the zeros are now being approximated based on the neighbouring values.

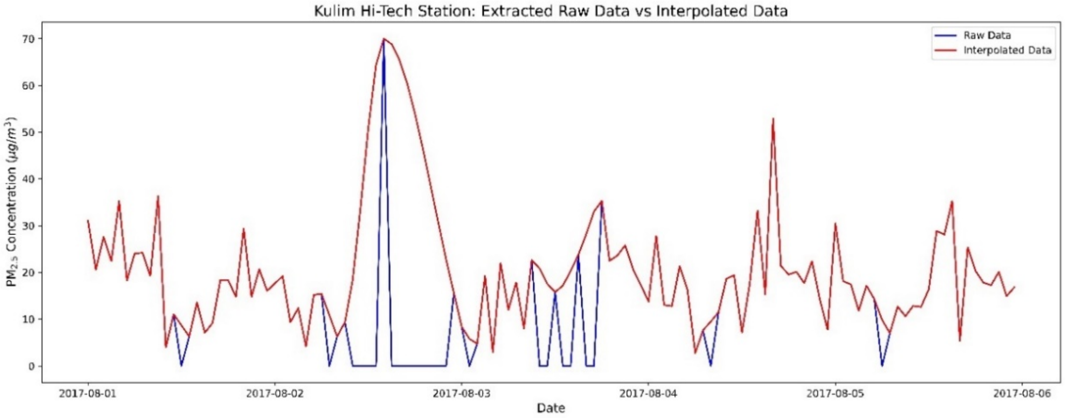


Figure 1. Extracted raw data and interpolated data for Kulim Hi-Tech station

Deep Learning

Deep learning is a subset of the bigger picture, machine learning. The introduction of deep learning by Hinton *et al*. [21] in 2006 has allowed artificial intelligence (AI) to thrive in various fields, such as healthcare and natural language processing [22].

Deep learning can be regarded as the upgraded version of neural networks. Both deep learning and neural networks consist of input and output layers, with the core difference of deep learning having one or more hidden layers. Hidden layers consist of artificial neurons that receive inputs (x_n) from the previous layer, assigning different weights (w) to each corresponding input and summing (Σ) all the values before passing the sum through the activation function (f) to obtain an output (y). There are a lot of activation functions available with rectified linear unit (ReLU) function being the primary choice [23]. Such a process is called forward propagation and is repeated n times for n number of hidden layers in the network. To optimise the network for better accuracy in outputs, backward propagation is carried out based on a cost function. Backward propagation makes use of chain rule differentiation and optimisation in calculus to optimise the network and reassign new weights for neurons in each hidden layer [24].

Figure 2 shows the architecture of an artificial neuron.

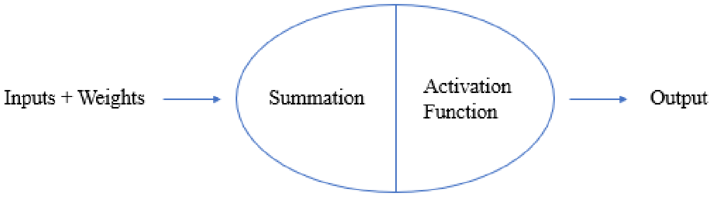


Figure 2. Architecture of an artificial neuron

It could be computationally complex and intensive to have a large number of artificial neurons in multiple hidden layers of a network. However, a model with higher accuracy and shorter running time on test data may be generated as it makes use of multiple levels of abstraction to represent data [25].

LSTM

A sequential artificial neural network known as LSTM was developed in 1997 by Hochreiter and Schmidhuber [26]. As an extension of RNN, LSTM has outperformed it by successfully overcoming several shortcomings of RNN. Since the introduction of LSTM, it has been extensively applied in various fields such as speech recognition, protein homology detection and time series prediction, which is the centre of attention in this research.

A model that is capable of capturing long-term temporal dependencies can produce desired predicted outputs with higher accuracy. This feature has a significant impact on forecasting of time series and sequential data, which is highly dependent on previously received inputs over a long period of time. However, RNN is unable to remember and relate distant data at a time far in the past to the current state in order to predict values of k steps ahead into the future. Such a drawback occurs due to the vanishing and exploding gradient problems that arise during the backpropagation through time (BPTT) training process [27]. When the backpropagated error decreases extremely fast approaching zero, the gradient is said to vanish and become insignificant. Exploding gradients are expected to take place when the backpropagated error increases exponentially to infinitely large weight updates. These problematic complications make it a challenging task to train a RNN model and so to learn effectively.

LSTM, however, having the capability to bridge long time lags, is capable of learning long-term dependencies [27]. Such an amazing improvement could be achieved by altering some features in the RNN architecture. RNN consists of a chain of repeating neural network cells that only has a single hyperbolic tangent (tanh) function within each block. LSTM remains the same chain-like structure but comprises two non-linear activation functions to scale or normalise data, which are sigmoid and tanh functions. In addition, there are three gates in LSTM, namely the forget gate, input gate and output gate, which play distinct roles in controlling the information flow inside the memory cell. In the equations presented later, we will find that there are four kernel weights W_h associated with the hidden state, four recurrent kernel weights W_x associated with the input vector and four bias vectors b.

The architecture of LSTM is illustrated in Figure 3 as follows.

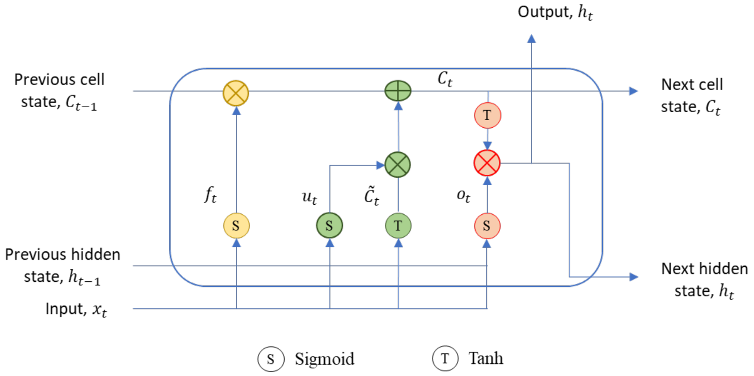


Figure 3. Architecture of LSTM

Basically, the LSTM working mechanism begins with a forget gate. This gate decides what kind of information is to be removed from the cell state [28]. The sigmoid function is applied to the inputs which converts them into values of interval $[0, 1]$. The information should be discarded when the values are close to 0 whereas values approaching 1 signify that the information should be preserved within the cell state. Next, the inputs are sent to the input gate that decides which portion of the latest information should be acquired in updating the cell state [28]. There are two different types of activation functions in the input gate, which are sigmoid and tanh functions. The updated information is stored in the current cell state C_t. Lastly, the output gate plays a vital role in determining the output information based on both input and cell state memory [28]. When the desired final current output is obtained, a copy of the output will be incorporated into the cell state, C_t whereas another copy of it will form an output hidden state, h_t that will flow into the next LSTM block or be used for prediction. The processes taking place in each gate are shown in the equations below.

Forget gate: $$f_t = \sigma\left(W_{hf}\, h_{t-1} + W_{xf}\, x_t + b_f\right) \tag{1}$$

Input gate: $$u_t = \sigma(W_{hu}\, h_{t-1} + W_{xu}\, x_t + b_u) \tag{2}$$
$$\tilde{C}_t = \tanh(W_{hC}\, h_{t-1} + W_{xC}\, x_t + b_C) \tag{3}$$
$$C_t = f_t \cdot C_{t-1} + u_t \cdot \tilde{C}_t \tag{4}$$

Output gate: $$o_t = \sigma(W_{ho}\, h_{t-1} + W_{xo}\, x_t + b_o) \tag{5}$$
$$h_t = o_t \cdot \tanh(C_t) \tag{6}$$

GRU

GRU is an improved version of standard RNN introduced in 2014 by Cho *et al*. [29] which aims to solve the vanishing gradient problem. The key distinction is RNN's support on gating of the hidden state: a mechanism for when a hidden state should be updated and reset. GRU can also be considered as a variation of LSTM since both are designed similarly with excellent predictions. Compared to the three gates in LSTM architecture, there are only two gates involved in each time step, which are reset gate and update gate. Thus, GRU is said to be simpler than LSTM [14].

Figure 4 below illustrates the architecture of GRU.

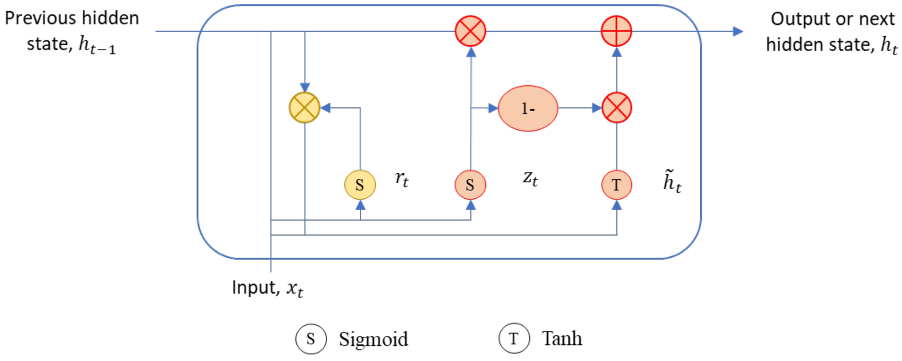

Previous hidden state, h_{t-1}
Output or next hidden state, h_t
1-
S
r_t
S
z_t
T
$\tilde{h}_t$
Input, x_t
S Sigmoid
T Tanh

Figure 4 Architecture of GRU

Similar to LSTM, the reset gate is used to decide how much past information is to forget whereas the update gate combines the roles of forget gate and input gate of LSTM. Based on Figure 4, the inputs include the current time step and the hidden state of previous time step. Outputs of the two gates are given by two fully connected layers with a sigmoid activation function which converts the inputs to values of interval $[0, 1]$.

Mathematically, for a given time step t, suppose there exist an input $x_t \in \mathbb{R}^{n\times d}$ and the hidden state from previous time step, denoted as $h_t \in \mathbb{R}^{n\times h}$, where n is the number of samples, d is the number of inputs and h is the number of hidden units. Thus,

Reset gate: $$r_t = \sigma(x_t\, W_{xr} + h_{t-1}\, W_{hr} + b_r) \tag{7}$$
Update gate: $$z_t = \sigma(x_t\, W_{xz} + h_{t-1}\, W_{hz} + b_z) \tag{8}$$

where $W_{xr}, W_{xz} \in \mathbb{R}^{d\times h}$ and $W_{hr}, W_{hz} \in \mathbb{R}^{h\times h}$ are weight parameters, $b_r, b_z \in \mathbb{R}^{1\times h}$ are biases while σ is sigmoid activation function.

Next, the reset gate, r_t, which is integrated as a regular latent state updating mechanism, leads to the following:

Candidate hidden state: $$\tilde{h}_t = \tanh(x_t\, W_{xh} + (r_t \odot h_{t-1})\, W_{hh} + b_h) \quad (9)$$

where $W_{xh} \in \mathbb{R}^{d\times h}$ and $W_{hh} \in \mathbb{R}^{h\times h}$ are weight parameters and b_h is bias.

The nonlinear tanh activation function is used to ensure the output, which is a candidate hidden state, remains in the interval $[-1, 1]$ [30]. Whenever the entries in the reset gate r_t are close to 1, the candidate hidden state is the outcome of tanh of input x_t and elementwise product between r_t and h_{t-1}. Meanwhile, if the entries in the reset gate r_t are close to 0, it means that the reset information has been stopped.

To determine the extent to which the previous hidden state $h_{t-1} \in \mathbb{R}^{n\times h}$ remains and how much the new candidate hidden state $\tilde{h}_t$ is used, an update gate z_t is incorporated. The process can be written as below.

New hidden state: $$h_t = r_t \odot h_{t-1} + (1 - z_t) \odot \hat{h}_t \quad (10)$$

If the value of z_t is close to 1, the hidden state h_t remains the same as the previous hidden state h_{t-1}, indicating that the information from input x_t is essentially ignored. However, if the value of z_t is close to 0, the new hidden state h_t approaches the candidate hidden state $\tilde{h}_t$. These designs help to cope with the vanishing gradient problem in standard RNN and better capture dependencies for sequences with large time steps [29,31].

Modelling Phase

Aiming to compare the performance of LSTM, GRU and VARIMA models in forecasting the hourly $PM_{2.5}$ concentration, the pre-processed data are split into a ratio of 80:20 for both training and test sets respectively as shown in Figure 5. The training set is used to supervise the training and learning processes of the three models, whereas the test set is used for validation and final evaluation of prediction accuracy.

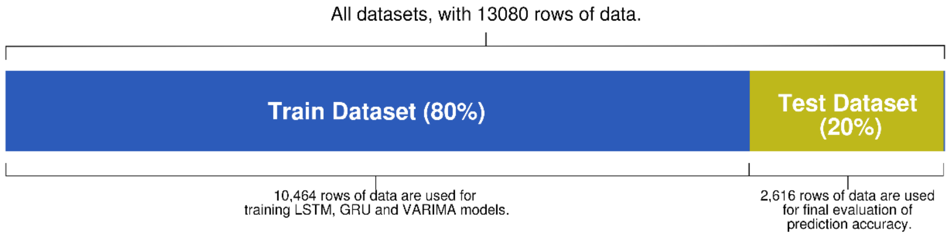

All datasets, with 13080 rows of data.
Train Dataset (80%)
Test Dataset (20%)
10,464 rows of data are used for training LSTM, GRU and VARIMA models.
2,616 rows of data are used for final evaluation of prediction accuracy.

Figure 5. Splitting of dataset.

Having MSE as the loss function and root mean squared propagation (RMSprop) as the optimiser with a dropout rate of 0.2, the number of epochs is fixed at 100 for both LSTM and GRU models to avoid overfitting [10]. The optimal values of other hyperparameters of LSTM and GRU such as number of hidden layers, number of units, batch size and time steps are fine-tuned using manual optimisation method. This approach involved selecting the best hyperparameter settings from a range of tested values based on cross-validation results. Meanwhile, the optimal orders of autoregressive terms (p), nonseasonal differences (d) and lagged forecast errors (q) in VARIMA are also selected from a few candidate combinations using the same technique.

In this context, a specialised form of cross-validation known as the 10-fold growing window cross-validation technique is utilised. Developed as a variant of traditional k-fold cross-validation, the size of training data gradually expands with each split, in contrast to the standard k-fold method where each fold of data serves as the training set exactly once [32]. Such a cross-validation procedure has demonstrated its efficacy in yielding more accurate estimates compared to conventional out-of-sample approaches [33].

Model Evaluation

As aforementioned, the prediction performance of LSTM, GRU and VARIMA models are evaluated and compared based on three accuracy metrics which are defined as follows:

Root mean square error: $$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{t=1}^{n}\left(\hat{Y}_t - Y_t\right)^2} \tag{11}$$

Mean absolute error: $$\mathrm{MAE} = \frac{1}{n}\sum_{t=1}^{n}\left|\hat{Y}_t - Y_t\right| \tag{12}$$

Mean absolute percentage error: $$\mathrm{MAPE} = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{\hat{Y}_t - Y_t}{Y_t}\right| \times 100 \tag{13}$$

where n is the amount of data in the test set, Y_t and $\hat{Y}_t$ are the actual and predicted values at time t respectively. A lower accuracy metric value indicates the model's ability in providing more precise forecasts by effectively learning the historical data trends.

Results and Discussion

Descriptive Analysis

Since the five selected air quality monitoring stations are located in different areas and regions, this might lead to varying concentrations of $PM_{2.5}$. Table 1 below demonstrates the distribution of stations by area and region, along with their respective average $PM_{2.5}$ concentration.

Table 1. Distribution of stations by area and region, along with their average $PM_{2.5}$ concentration (μg/m^3)

Station	Area	Region	Average $PM_{2.5}$ Concentration (μg/m^3)
Kulim Hi-Tech	Industrial	West Coast	16.08
Shah Alam	Urban	West Coast	24.42
Larkin	Urban	West Coast	20.35
Balok Baru, Kuantan	Industrial	East Coast	18.25
Kuala Terengganu	Urban	East Coast	16.63

Out of these five stations, three are located in urban areas, namely Shah Alam station, Larkin station and Kuala Terengganu station. These stations typically exhibit an average $PM_{2.5}$ concentration higher than the others, with Shah Alam station and Larkin station recording average readings exceeding 20 μg/m^3. This results in an overall average concentration of 20.47 μg/m^3 recorded in urban areas. Conversely, the other two stations exhibit relatively lower $PM_{2.5}$ concentrations, leading to an average concentration of 17.17 μg/m^3 in industrial areas.

Such a situation is in accordance with the findings of Abdul Rahman *et al.* [34], who classified industrial areas in Malaysia primarily into Medium Pollution Regions (MPR) and Low Pollution Regions (LPR) based on the $PM_{2.5}$ concentrations, while most of the urban areas are categorised as High Pollution Regions (HPR) and MPR. According to the identified clusters, the mean $PM_{2.5}$ concentrations in HPR, MPR and LPR are 23.04 μg/m^3, 16.41 μg/m^3 and 16.18 μg/m^3, respectively. Due to the frequent emission of pollutants from human activities and vehicles, urban areas with high population density often experience poorer air quality than less developed areas [35]. On the other hand, industrial areas mainly experience air pollution from contaminants released by industrial plants, with fewer contributions from other human activities. Consequently, particulate matter becomes more concentrated in urban areas, resulting in relatively higher $PM_{2.5}$ concentrations.

The stations centred along the west coast of Peninsular Malaysia are more significantly affected by fine particulates as compared to the stations in the east coast region, with overall average $PM_{2.5}$ concentrations recorded at 20.28 μg/m^3 and 17.44 μg/m^3 respectively. This can be attributed to the fact that the states in the west coast region, having their coastlines along the Straits of Malacca, are more developed with higher population density than the east coast region [36], which then causes more emissions of air pollutants into the atmosphere. In addition, it is believed that meteorological factors, particularly wind direction, play a significant role in the transboundary haze pollution originating from a neighbouring country during the northeast monsoon season, subsequently increasing the air pollution levels in the west coast region [34].

Comparison of Model Performance

Selecting suitable hyperparameter values for neural networks is a challenging yet crucial task as it has a significant impact on model performance [7]. The optimal neural network hyperparameter values and orders of the VARIMA model are presented in Table 2 and Table 3 respectively.

Table 2. Optimal hyperparameter values for LSTM and GRU

Hyperparameter	LSTM	GRU
Loss function	MSE	MSE
Activation function	Tanh	Tanh
Recurrent activation	Sigmoid	Sigmoid
Optimiser	RMSprop	RMSprop
Number of hidden layers	1	1
Number of units	20	20
Dropout	0.2	0.2
Number of epochs	100	100
Batch size	64	64
Number of time steps	80	80

Table 3. Optimal parameter orders for VARIMA model

Parameter	Optimal Order
p	1
d	1
q	1

Table 4 below demonstrates the model performance based on RMSE, MAE and MAPE values.

Table 4. Model performance

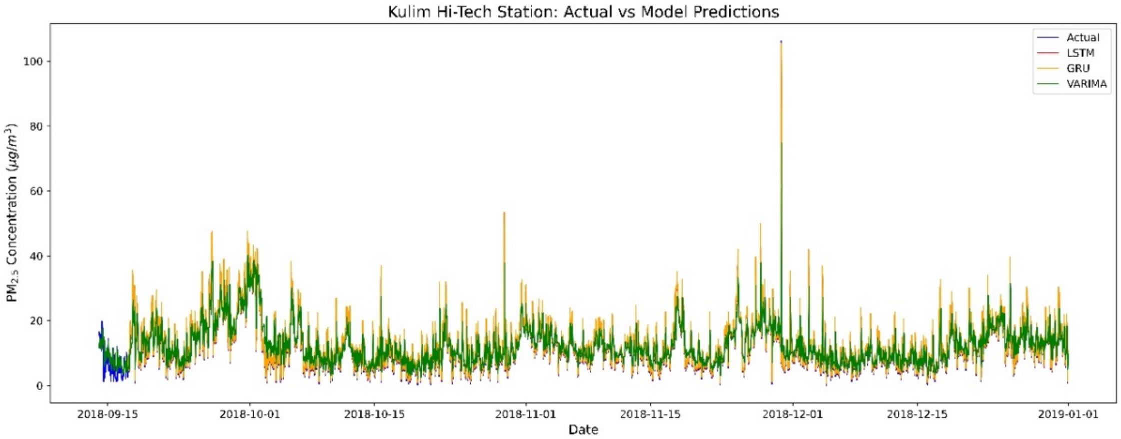
Station	Area	Region	Accuracy Metric	LSTM	GRU	VARIMA
Kulim Hi-Tech	Industrial	West Coast	RMSE	**0.2274**	0.4459	5.7407
			MAE	**0.1934**	0.4350	3.8095
			MAPE	**3.8282**	6.7469	60.2737
Shah Alam	Urban	West Coast	RMSE	**0.3457**	0.4660	6.5139
			MAE	**0.3128**	0.4423	4.8979
			MAPE	2.4745	**2.1785**	31.9425
Larkin	Urban	West Coast	RMSE	0.5829	**0.5636**	6.8953
			MAE	**0.5448**	0.5474	4.7658
			MAPE	5.0522	**4.6102**	36.9937
Balok Baru, Kuantan	Industrial	East Coast	RMSE	1.7274	**0.9051**	6.4796
			MAE	1.5973	**0.8320**	4.2491
			MAPE	28.7889	**14.9920**	66.3043
Kuala Terengganu	Urban	East Coast	RMSE	1.5371	**0.5437**	8.8058
			MAE	1.5045	**0.5106**	4.5163
			MAPE	33.5867	**8.8612**	81.9289

Both LSTM and GRU significantly outperform the VARIMA model with comparable accuracy levels between them. On the contrary, the VARIMA model yields exceptionally high prediction errors for all stations, indicating a poor goodness-of-fit between the fitted model and observed data.

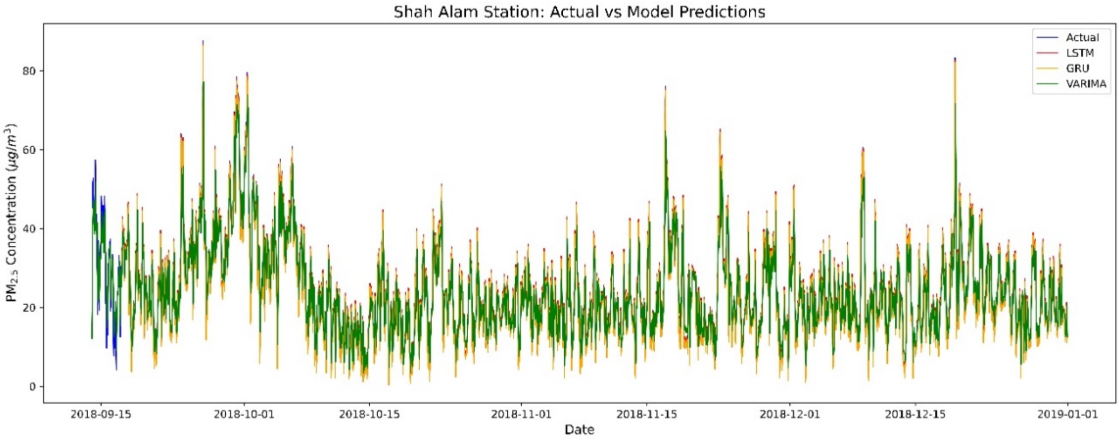
In detail, GRU has the lowest accuracy metric values at three out of five monitoring stations, primarily located in the east coast region. Meanwhile, LSTM, the other variant of neural network model, predictably gives a better prediction accuracy than the statistical time series model across all stations with the best prediction performance achieved for Kulim Hi-Tech station and Shah Alam station which are situated in the west coast region.

Looking at the efficacy of the deep learning models by area, it is observed that GRU predicts slightly better for urban stations. Nevertheless, there is no definitive evidence to deduce that the area of stations significantly influences model performance due to comparable prediction errors between the two models.

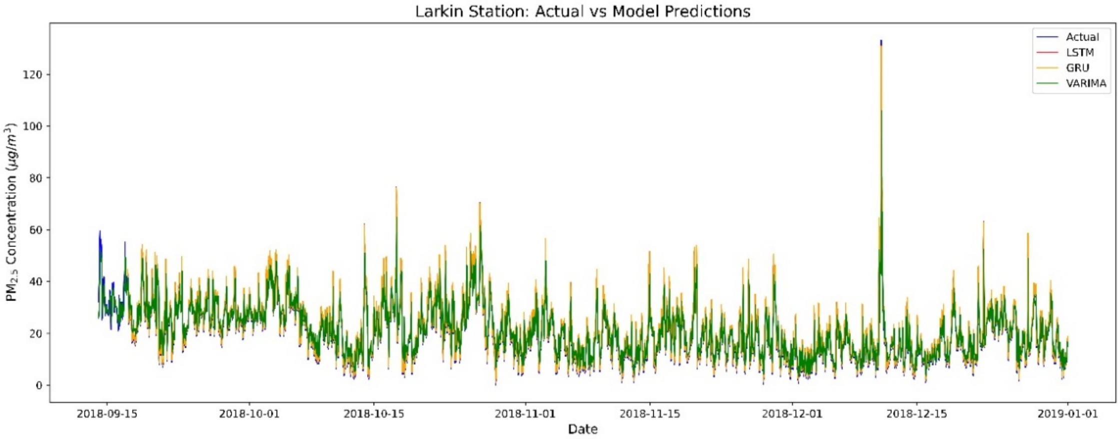
In order to better observe the fitting degree of each model at the five stations, comparisons between actual data and predictions are visualised in Figure 6.



(a)

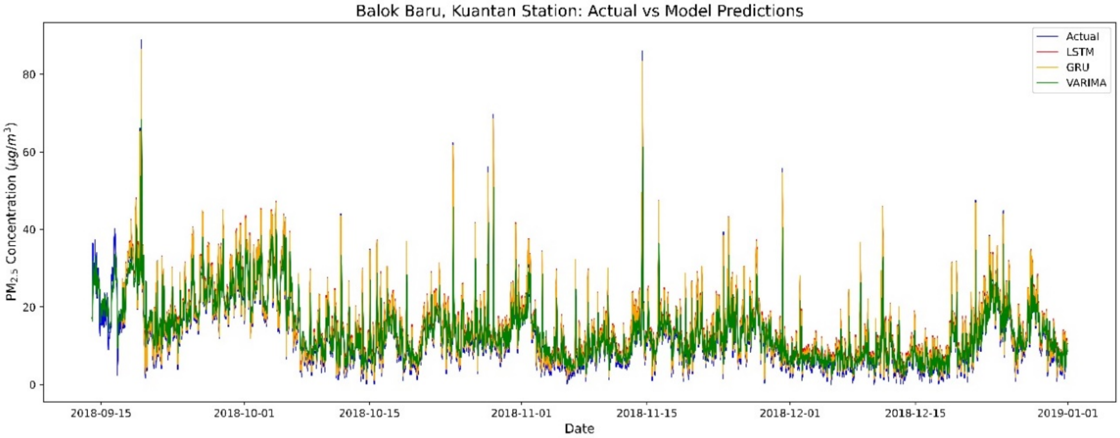


(b)

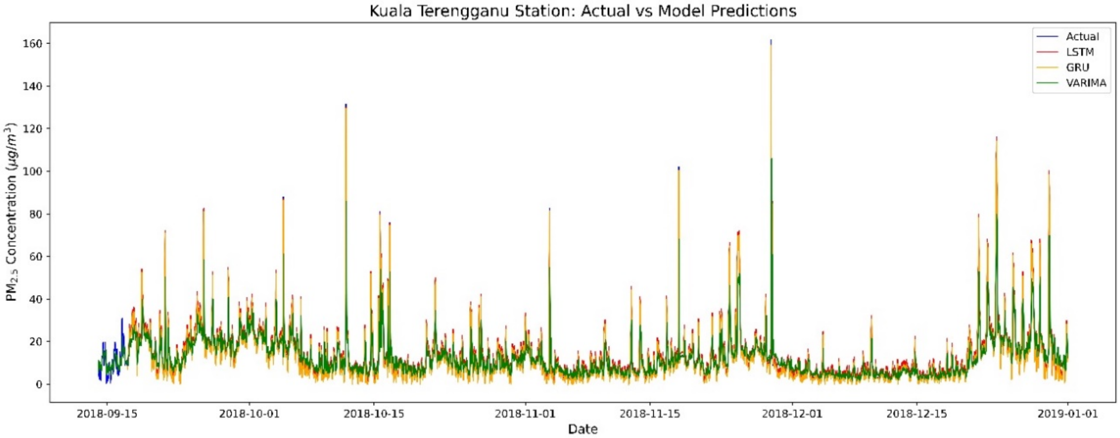


(c)

Continue to next page



(d)



(e)

Figure 6. Comparisons between actual data and predictions. (a) Kulim Hi-Tech station. (b) Shah Alam station. (c) Larkin station. (d) Balok Baru, Kuantan station. (e) Kuala Terengganu station

As depicted in Figure 6, the predictions made by LSTM and GRU exhibit an 80-data-point lag as compared to the VARIMA model which does not consider time step. This is due to the incorporation of time steps into both LSTM and GRU models. Taking time steps into account allows the deep learning models to study the historical data patterns, subsequently producing more accurate predictions.

Consistent with the model performance tabulated in Table 4, both LSTM and GRU are able to fit the actual data much better as compared to the VARIMA model across all stations. They can predict the high spikes most of the time but occasionally fail to capture the magnitude of sudden dips in the dataset. In particular, the GRU predictions are closer to the actual data compared to LSTM for Larkin station, Balok Baru, Kuantan station and Kuala Terengganu station, aligning with the insights gained from the model evaluation.

While these two models demonstrate strong abilities to capture the patterns of the actual time series, the multivariate VARIMA model gives poorer predictions with a more conservative magnitude. Such a situation could be due to the influence of other meteorological factors at each station on the $PM_{2.5}$ concentration [37]. As all five stations are located in different regions and areas, they might have encountered different atmospheric conditions including wind speeds, temperature and relative humidity. Consequently, this contributes to the much higher prediction errors in the VARIMA model when considering the $PM_{2.5}$ concentration at other stations.

These findings prove the capability of novel neural networks in time series forecasting due to their continuous evolution of calculation power, causing them to gain increasing popularity in recent decades [14]. Specifically, LSTM and GRU models have showcased superior performance in accurately predicting sequential time series data [38]. Research by Mitrea *et al*. [39] revealed that the predictive capability of neural network models is much better in comparison with traditional forecasting methods such as ARIMA in forecasting the inventory level. Additionally, the LSTM model also performed significantly better than the ARIMA model in predicting PM_{10} concentrations in Peninsular Malaysia [10].

The limitations of conventional statistical models such as ARIMA and VARIMA in capturing the stochastic nature of the data are made apparent by the high accuracy metric values obtained [14]. Consequently, neural network models are often found to be efficient in dealing with complex modelling of time series data, especially in the realm of environmental forecasting [40].

Apart from prediction accuracy, computational time could also be taken into account when selecting the most suitable model for forecasting, especially when weighing the trade-off between these two factors. Due to the complex architecture of neural networks, these deep learning approaches often come with the drawback of being computationally expensive as compared to classical statistical algorithms [41]. This is proven when both LSTM and GRU require about three times the computational time taken to train the traditional VARIMA model as shown in Table 5.

Table 5. Computational time (seconds)

Station	LSTM	GRU	VARIMA
Kulim Hi-Tech	342.6219	358.4327	142.3039
Shah Alam	355.3194	381.5150	142.3039
Larkin	320.7117	355.3540	142.3039
Balok Baru, Kuantan	334.9806	373.7051	142.3039
Kuala Terengganu	359.7846	394.8448	142.3039

Beyond the preceding analysis, it is noteworthy to emphasise the implication of employing PCHIP on prediction performance, as demonstrated in Table 6.

Table 6. Number of interpolated values and the best model prediction accuracy

Station	Total Number of Interpolated Values	Best Model	RMSE	MAE	MAPE
Kulim Hi-Tech	775 (5.93%)	LSTM	0.2274	0.1934	3.8282
Shah Alam	452 (3.46%)	LSTM	0.3457	0.3128	2.4745
Larkin	257 (1.96%)	GRU	0.5636	0.5474	4.6102
Balok Baru, Kuantan	194 (1.48%)	GRU	0.9051	0.8320	14.9920
Kuala Terengganu	532 (4.07%)	GRU	0.5437	0.5106	8.8612

As depicted in Table 6, the accuracy metric values are higher for Larkin station and Balok Baru, Kuantan station which have the lowest percentages of interpolated values in comparison with other stations. This suggests a relatively poorer model performance for these stations. It is observed that, in general, the greater the number of interpolated values, the lower the prediction errors tend to be.

Such findings are in accordance with the results obtained by Sobolewski and Miczulski [42] in which they applied PCHIP function in preparing data for GMDH-type neural network with one-day interval. As a result, the best prediction of local time scales was achieved by using pre-processed time series, as evidenced by the comparison of residuals and prediction quality measures. Meanwhile, Jaffar A. *et al*. [43] implemented PCHIP to substitute missing hydrological data, thereby mitigating potential bias in the interpretation of conclusive hydrological parameter analysis. Of the three examined interpolation methods, PCHIP delivers the most identical interpolated data to the original data. This is mainly due to the monotonic nature of PCHIP which minimises the oscillation effects when substituting the data, subsequently avoiding overshooting issue when there is an abrupt change in the data pattern. Such a smoother curve enhances data fitting and improves prediction accuracy.

Conclusions

This study explores the potential use of LSTM, GRU and VARIMA models in predicting the hourly $PM_{2.5}$ concentrations. The comparative analysis of prediction accuracy among these models reveals that LSTM and GRU significantly outperform the conventional time series VARIMA model across all five stations in terms of prediction accuracy and the ability to capture the actual data patterns.

In addition to proving the strength of deep neural networks in time series forecasting as compared to the conventional statistical models, the present study also highlights the effectiveness of PCHIP in interpolating data of blank, zero and negative values while ensuring the monotonicity of the interpolated values, which may come in handy for future research.

As the air quality may be influenced by other air pollutants not included in this dataset, future research may consider investigating the impact of primary air pollutants such as carbon monoxide (CO), nitrogen dioxide (NO_2) and particulate matter of 10 micrometres or less in diameter (PM_{10}) on air quality. Besides, meteorological factors such as air humidity, wind speed and wind direction are also deemed significant in causing the fluctuation of $PM_{2.5}$ concentration. Therefore, future work should incorporate more comprehensive data on air pollutants and meteorological conditions to further enhance $PM_{2.5}$ concentration predictions, while simultaneously gaining deeper insights into the relationships among these factors.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgement

Utmost appreciation to Universiti Kebangsaan Malaysia (UKM) for fully supporting the authors in conducting this research and to the Ministry of Higher Education for the research grants (FRGS/1/2024/ICT06/UKM/02/2 and FRGS/1/2019/STG06/UKM/02/4). Also, the authors would like to thank the Department of Environment Malaysia for providing the data that has made this research possible.

References

[1] Mabahwi, N. A. B., Ling, O. H. L., & Omar, D. (2014). Human health and wellbeing: Human health effect of air pollution. *Procedia – Social and Behavioral Sciences, 153,* 221–229.

[2] WHO. (2018). 9 out of 10 people worldwide breathe polluted air, but more countries are taking action. *WHO News Release.*

[3] Global Environmental Forum. (2000). *Overseas Environmental Measures of Japanese Companies (Malaysia): Research Report on Trends in Environmental Considerations Related to Overseas Activities of Japanese Companies FY 1999.* Tokyo: Ministry of the Environment, Government of Japan.

[4] Nur-Nabilah, M. N., Nor-Amani-Filzah, M. K., Norzila, O., Azra-Munirah, M. D., Nurul-Bahiyah, A. W., & Khairuddin, M. K. (2021). Discovering source of residents' complaint on air quality: Preliminary studies on particulate matter (PM2.5) and sulphur dioxide (SO2). *IOP Conference Series: Materials Science and Engineering, 1144,* 012045.

[5] Su, B. D., Zhan, M. J., Zhai, J. Q., Wang, Y. J., & Fischer, T. (2015). Spatio-temporal variation of haze days and atmospheric circulation pattern in China (1961–2013). *Quaternary International, 380-381,* 14–21.

[6] Xing, Y. F., Xu, Y. H., Shi, M. H., & Lian, Y. X. (2016). The impact of PM2.5 on the human respiratory system. *Journal of Thoracic Disease, 8*(1), E69–E74.

[7] Ao, D., Cui, Z., & Gu, D. (2019). Hybrid model of air quality prediction using k-means clustering and deep neural network. *Proceedings of the 38th Chinese Control Conference,* 8416–8421.

[8] Caraka, R. E., Chen, R. C., Toharudin, T., Pardamean, B., Yasin, H., & Wu, S. H. (2019). Prediction of status particulate matter 2.5 using state Markov chain stochastic process and HYBRID VAR-NN-PSO. *IEEE Access, 7,* 161654–161665.

[9] Zhou, X., Xu, J., Zeng, P., & Meng, X. (2019). Air pollutant concentration prediction based on GRU method. *Journal of Physics: Conference Series, 1168,* 032058.

[10] Bakar, M. A. A., Ariff, N. M., Nadzir, M. S. M., Ong, L. W., & Suris, F. N. A. (2022). Prediction of multivariate air quality time series data using long short-term memory network. *Malaysian Journal of Fundamental and Applied Sciences, 18,* 52–59.

[11] Uh, B. H., & Majid, N. (2021). Comparison of ARIMA model and artificial neural network in forecasting gold price. *Journal of Quality Measurement and Analysis, 17*(2), 31–39.

[12] Tsan, Y.-T., Chen, D.-Y., Liu, P.-Y., Kristiani, E., Nguyen, K. L. P., & Yang, C.-T. (2022). The prediction of influenza-like illness and respiratory disease using LSTM and ARIMA. *International Journal of Environmental Research and Public Health, 19*(3), 1858.

[13] Tan, W. M., & Othman, Z. (2021). Ramalan jumlah kandungan elektron GPS menggunakan ingatan jangka pendek yang panjang dan unit berulang berpagar. *Undergraduate Dissertation,* Universiti Kebangsaan Malaysia.

[14] Mateus, B. C., Mendes, M., Farinhaa, J. T., Assis, R., & Cardoso, A. M. (2021). Comparing LSTM and GRU models to predict the condition of a pulp paper press. *Energies, 14*(21), 6958.

[15] ArunKumar, K. E., Kalaga, D. V., Kumar, C. M. S., Kawaji, M., & Brenza, T. M. (2022). Comparative analysis of gated recurrent units (GRU), long short-term memory (LSTM) cells, autoregressive integrated moving average (ARIMA), seasonal autoregressive integrated moving average (SARIMA) for forecasting COVID-19 trends. *Alexandria Engineering Journal, 61*(10), 7585–7603.

[16] Rusyana, A., Tatsara, N., Balqis, R., & Rahmi, S. (2020). Application of clustering and VARIMA for rainfall prediction. *IOP Conference Series: Materials Science and Engineering, 769*(1), 012063.

[17] Setiawan, A., Aidi, M. N., & Sumertajaya, I. M. (2015). Modelling of forecasting monthly inflation by using VARIMA and GSTARIMA models. *Forum Statistika dan Komputasi: Indonesian Journal of Statistics, 20*(2), 60–63.

[18] Zainuri, N. A., Jemain, A. A., & Muda, N. (2015). A comparison of various imputation methods for missing values in air quality data. *Sains Malaysiana, 44*(3), 449–456.

[19] Jiang, N., Akter, R., Ross, G., White, S., Kirkwood, J., Gunashanhar, G., Thompson, S., Riley, M., & Azzi, M. (2023). On thresholds for controlling negative particle (PM2.5) readings in air quality reporting. *Environmental Monitoring and Assessment, 195,* 1187.

[20] Gariazzo, S., Giunti, C., & Laveder, M. (2015). Light sterile neutrinos and inflationary freedom. *Journal of Cosmology and Astroparticle Physics, 04,* 023.

[21] Hinton, G. E., Osindero, S., & Teh, Y.-W. (2006). A fast learning algorithm for deep belief nets. *Neural Computation, 18,* 1527–1554.

[22] Sarker, I. H. (2021). Deep learning: A comprehensive overview on techniques, taxonomy, applications and research directions. *SN Computer Science, 2,* 420.

[23] Dubey, S. R., Singh, S. K., & Chaudhuri, B. B. (2022). Activation functions in deep learning: A comprehensive survey and benchmark. *Neurocomputing, 503,* 92–108.

[24] Wechsler, H. (1992). *Neural networks for perception.* Michigan: Academic Press.

[25] Zhu, N., Liu, X., Liu, Z., Hu, K., Wang, Y., Tan, J., Huang, M., Zhu, Q., Ji, X., Jiang, Y., & Guo, Y. (2018). Deep learning for smart agriculture: Concepts, tools, applications, and opportunities. *International Journal of Agricultural and Biological Engineering, 11*(4), 32–44.

[26] Taweh Beysolow II. (2017). *Introduction to deep learning using R.* California: Apress Berkeley.

[27] Ghatak, A. (2019). *Deep learning with R.* Singapore: Springer.

[28] Jason B. (2017). *Long short-term memory networks with Python.* Australia: Jason Brownlee.

[29] Cho, K., Merrienboer, B., Gulcehre, C., 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,* 1724–1734.

[30] Zhang, A., Lipton, Z. C., Li, M., & Smola, A. J. (2022). *Dive into deep learning.* Amazon Science.

[31] LazyProgrammer. (2016). *Deep learning: Recurrent neural networks in Python: LSTM, GRU and more RNN machine learning architectures in Python and Theano* (Kindle ed.). Kindle Scribe.

[32] Schnaubelt, M. (2019). A comparison of machine learning model validation schemes for non-stationary time series data. *FAU Discussion Papers in Economics.* Friedrich-Alexander University Erlangen-Nuremberg, Institute for Economics.

[33] Cerqueira, V., Torgo, L., & Mozetič, I. (2020). Evaluating time series forecasting models: An empirical study on performance estimation methods. *Machine Learning, 109,* 1997–2028.

[34] Ab. Rahman, E., Hamzah, F. M., Latif, M. T., & Dominick, D. (2022). Assessment of PM2.5 patterns in Malaysia using the clustering method. *Aerosol and Air Quality Research, 22,* 210161.

[35] Leh, O. L. H., Ahmad, S., Aiyub, K., Jani, Y. M., & Hwa, T. K. (2012). Urban air environmental health indicators for Kuala Lumpur City. *Sains Malaysiana, 41*(2), 179–191.

[36] Ramli, N., Abdul Hamid, H., Yahaya, A. S., Ul-Saufie, A. Z., Mohamed Noor, N., Abu Seman, N. A., Kamarudzaman, A. N., & Deák, G. (2023). Performance of Bayesian model averaging (BMA) for short-term prediction of PM10 concentration in the Peninsular Malaysia. *Atmosphere, 14*(2), 311.

[37] Ariff, N. M., Bakar, M. A. A., & Lim, H. Y. (2023). Prediction of PM10 concentration in Malaysia using k-means clustering and LSTM hybrid model. *Atmosphere, 14*(5), 853.

[38] Sugiyarto, A. W., & Abadi, A. M. (2019). Prediction of Indonesian palm oil production using long short-term memory recurrent neural network (LSTM-RNN). *Proceedings of the 2019 1st International Conference on Artificial Intelligence and Data Sciences (AiDAS),* 53–57.

[39] Mitrea, C. A., Lee, C. K. M., & Wu, Z. (2009). A comparison between neural networks and traditional forecasting methods: A case study. *International Journal of Engineering Business Management, 1*(2), 19–24.

[40] Krishan, M., Jha, S., Das, J., Singh, A., Goyal, M. K., & Sekar, C. (2019). Air quality modelling using long short-term memory (LSTM) over NCT-Delhi, India. *Air Quality, Atmosphere & Health, 12,* 899–908.

[41] Kontopoulou, V. I., Panagopoulos, A. D., Kakkos, I., & Matsopoulos, G. K. (2023). A review of ARIMA vs. machine learning approaches for time series forecasting in data driven networks. *Future Internet, 15*(8), 255.

[42] Sobolewski, Ł., & Miczulski, W. (2021). Methods of constructing time series for predicting local time scales by means of a GMDH-type neural network. *Applied Sciences, 11*(12), 5615.

[43] Jaffar, A., Thamrin, N. M., Ali, M. S. A. M., Misnan, M. F., Yassin, A. I. M., & Zan, N. M. (2022). Spatial interpolation method comparison for physico-chemical parameters of river water in Klang River using MATLAB. *Bulletin of Electrical Engineering and Informatics, 11*(4), 2368–2377.