IEEE Access
Multidisciplinary ⁝ Rapid Review ⁝ Open Access Journal

Received December 10, 2021, accepted February 2, 2022, date of publication February 18, 2022, date of current version March 9, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3152818

Long Short Term Memory Water Quality Predictive Model Discrepancy Mitigation Through Genetic Algorithm Optimisation and Ensemble Modeling

DHRUTI DHEDA[1], LING CHENG[1], (Senior Member, IEEE), AND ADNAN M. ABU-MAHFOUZ[2], (Senior Member, IEEE)

[1]School of Electrical and Information Engineering, University of the Witwatersrand, Johannesburg 2050, South Africa
[2]Council for Scientific and Industrial Research (CSIR), Pretoria 0001, South Africa

Corresponding author: Ling Cheng (ling.cheng@wits.ac.za)

This work was supported in part by the Department of Science and Innovation-Council for Scientific and Industrial Research (DSI-CSIR)-Inter-bursary Support Programme; and in part by the National Research Foundation (NRF), South Africa, under Grant 132651 and Grant 129311.

ABSTRACT A predictive long short-term memory (LSTM) model developed on a particular water quality dataset will only apply to the dataset and may fail to make an accurate prediction on another dataset. This paper focuses on improving LSTM model tolerance by mitigating discrepancies in model prediction capability that arises when a model is applied to different datasets. Two predictive LSTM models are developed from the water quality datasets, Baffle and Burnett, and are optimised using the metaheuristic genetic algorithm (GA) to create hybrid GA-optimised LSTM models that are subsequently combined with a linear weight-based technique to develop a tolerant predictive ensemble model. The models successfully predict river water quality in terms of dissolved oxygen concentration. After GA-optimisation, the RMSE values of the Baffle and Burnett models decrease by 42.42% and 10.71%, respectively. Furthermore, two ensemble models are developed from the GA-hybrid models, namely the average ensemble and the optimal weighted ensemble. The GA-Baffle RMSE values decrease by 5.05% for the average ensemble and 6.06% for the weighted ensemble, and the GA-Burnett RMSE values decrease by 7.84% and 8.82%, respectively. When tested on unseen and unrelated datasets, the models make accurate predictions, indicating the applicability of the models in domains outside the water sector. The consistent and similar performance of the models on any dataset illustrates the successful mitigation of discrepancies in the predictive capacity of individual LSTM models by the proposed ensemble scheme. The observed model performance highlights the datasets on which the models could potentially make accurate predictions.

INDEX TERMS Ensemble model, environment, genetic algorithm, long short term memory, rivers, water, water quality, water conservation, weight based model fusion.

I. INTRODUCTION

Rivers are valuable inland water resources utilised for human consumption, agricultural needs, industrial and recreational purposes. Increased urbanisation, poor water infrastructure, and climate change have increased pressure on rivers, necessitating efficient water management. To effectively manage rivers, the quality of the water must be continuously monitored [1].

Water quality is commonly evaluated through expensive and time-consuming laboratory analyses. This process includes, but is not limited to, water sample collection from the relevant river, the correct storage and transportation of samples to the laboratory, chemical laboratory tests and analysis, after which the quality of the water can be evaluated. There is more than enough room for error and inefficiency in this layered process [2]. The ability to predict water quality

The associate editor coordinating the review of this manuscript and approving it for publication was Artur Antonyan.

beforehand can greatly increase the efficiency of water management [3].

This study proposes the optimized Long Short-Term Memory (LSTM) model, which is an advanced recurrent neural network (RNN), for water quality prediction. The LSTM is the most appropriate network for sequential data in which temporal dependency is an implicit feature and when the retention of information of the earlier stages of the sequential data is necessary for forecasting future trends [4]. Such is the case with time-sequential water data used for water quality prediction.

Discrepancies in LSTM predictive model capability can arise when the model, developed using a particular water quality dataset, is applied to different water quality datasets for prediction purposes. The model will probably not make an accurate prediction on other water quality datasets. The LSTM models tend to be case study-specific.

This research aims to improve the tolerance of LSTM prediction models by mitigating these discrepancies through optimising the LSTM network using the metaheuristic genetic algorithm (GA). Two different GA-optimised LSTM models, used as base models, will be fused using a linear weight-based approach to create a final tolerant LSTM ensemble model.

This research produced three main contributions. The first was adaptation and optimisation through the successful adaptation of the LSTM network for water quality prediction for two temporal-based water quality datasets taken from different rivers and time periods. Both models were subsequently optimized by GA, to improve efficiency and robustness, resulting in two-hybrid GA-optimised LSTM prediction models. The second contribution was the combination of the two-hybrid GA-LSTM prediction models, using a weight-based technique to develop a single more tolerant ensemble model. Generalization, the third contribution, explored the possible use of the final GA-optimised LSTM-based ensemble prediction model in areas other than water quality. The purpose was to assert the tolerance and thus the relevance of the final ensemble model in the wider field of LSTM and ensemble prediction models.

The paper progresses as follows: Sec. I contains the introduction, Sec. II details the LSTM and Sec. III the GA, Sec. IV describes the weight-based combination technique, while Sec. V discusses the water bodies (rivers) and relevant water quality parameters used. After which, Sec. VI describes the development of the robust and tolerant water quality prediction LSTM based ensemble scheme, while Sec. VII focuses on the results and analysis, and Sec. VIII concludes the paper with further recommendations.

II. LONG-SHORT TERM MEMORY (LSTM) NETWORK

Most artificial neural networks (ANNs) are feed-forward neural networks [5], incapable of capturing sequences and accounting for the temporal nature of data and thus cannot model memory [10]. The RNN is a Deep Neural Network (DNN) that has a looping mechanism, allowing for the information retention of the previously processed elements in the sequence, thus enabling the modeling of memory [8], [11].

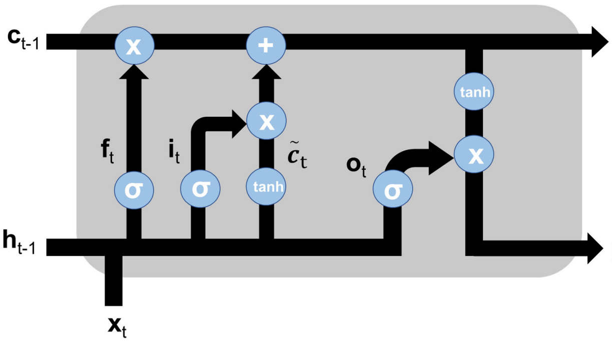


FIGURE 1. Internal structure of LSTM recurrent network cell adapted from [16].

RNNs are incapable of learning long time dependencies and retaining information of the beginning of the sequence, skewing results [10], [12]. RNNs are trained using backpropagation, where the gradients of the previous layer are used to calculate the gradients of the current layer [13]. Small weight adjustments will result in smaller weight adjustments with each subsequent layer [7], [8] until the internal weights are barely adjusted, and earlier network layers cannot learn anything [9]. This shrinking of gradients [13] is referred to as the vanishing gradient [9]. The LSTM is widely used to mitigate the complications created by the vanishing gradient [8] and was first suggested by Hochreiter and Schmidhuber in 1997 [14]. The opposite, the phenomena of gradients increasing exponentially in size [6], [8], referred to as the exploding gradient can also occur [13]. Gradient clipping, which entails shrinking the gradient when norms exceed a particular threshold, is used to mitigate this [6], [8].

The LSTM can scale to longer sequences with its unique architecture [10], which is illustrated in Figure 1 [16]. Each LSTM cell has a cell state, the network memory [6], [15] which is regulated by the forget, input, and output LSTM gates that control which information is added or removed from the cell state, ensuring that only relevant information is used to make predictions [15].

The forget gate ($\mathbf{f}_t$) controls information removal and retention. The input of the LSTM cell at the current time step ($\mathbf{x}_t$) and the output from the previous hidden state ($\mathbf{h}_{t-1}$) are both individually multiplied by the weight of the forget gate (W_f) and summed together, the result of which is added to a bias vector ($\mathbf{b}_f$) then passed through the logistic sigmoid activation (σ) [15] as shown in (1). Notably, each gate has a different set of weights. The equations below are expressed in accordance to Figure 1 [16] and were adapted from [4]:

$$\mathbf{f}_t = \sigma(W_f \mathbf{x}_t + W_f \mathbf{h}_{t-1} + \mathbf{b}_f). \tag{1}$$

When the previous hidden state and current input are multiplied by the weight of the input gate (W_i), added to the bias vector ($\mathbf{b}_i$) and passed through the logistic sigmoid activation, it determines which values are updated. And hence the input gate ($\mathbf{i}_t$) can update the cell state [15] in (2):

$$\mathbf{i}_t = \sigma(W_i\mathbf{x}_t + W_i\mathbf{h}_{t-1} + \mathbf{b}_i). \tag{2}$$

An intermediate cell state ($\tilde{\mathbf{c}}_t$) is calculated, where the output of the previous hidden state and the current input is multiplied by the weight of the intermediate cell state (W_c), the product of which is added to a bias vector ($\mathbf{b}_c$) and passed through a tanh function to regulate the network [6], [15] as expressed in(3):

$$\tilde{\mathbf{c}}_t = \tanh(W_c\mathbf{x}_t + W_c\mathbf{h}_{t-1} + \mathbf{b}_c). \tag{3}$$

The Hadamard element-wise product ($\circ$) of the output of the forget gate and the memory of the previous state ($\mathbf{c}_{t-1}$) is then used to calculate the current cell state ($\mathbf{c}_t$) when it is added (through point-wise addition) to the product of the output of the intermediate cell state and the output of the input gate. Thus the sigmoid output determines what should be retained from the tanh output [15] as expressed in (4):

$$\mathbf{c}_t = \mathbf{f}_t \circ \mathbf{c}_{t-1} + \mathbf{i}_t \circ \tilde{\mathbf{c}}_t. \tag{4}$$

Using the current cell state, the output gate ($\mathbf{o}_t$) determines what the next hidden state should be when the previous hidden state and the current input are multiplied by the weight of the output gate (W_o), then added to a bias vector ($\mathbf{b}_o$) and finally passed through a logistic sigmoid activation [15] as shown in (5):

$$\mathbf{o}_t = \sigma(W_o\mathbf{x}_t + W_o\mathbf{h}_{t-1} + \mathbf{b}_o). \tag{5}$$

The hidden state ($\mathbf{h}_t$) is then calculated by passing the current cell state through the tanh activation and multiplying it by the output gate in (6), determining the information carried by the hidden state to the next time step [15].

$$\mathbf{h}_t = \mathbf{o}_t \circ \tanh \mathbf{c}_t. \tag{6}$$

The output is dependent on the input at the current time step and the previous hidden state and is modulated between one and zero by the sigmoid functions. The gate will block a signal if the output is zero. The model learns the weights and biases of each gate through the minimisation of the difference between the LSTM outputs and the training samples [17].

LSTMs have several parameters, such as the number of layers, the number of units in the hidden layer, time window size, batch size, etc., referred to as hyperparameters [18], which influence network behaviour [4] and thus should be optimised before the training process [18].

Hyperparameter optimisation can be manual or automatic [4]. The manual adjustment of each hyperparameter and the interpretation of the effect of the adjustment on the network are dependent on the knowledge and experience of the researcher. Automatic optimisation ranges from the exhaustive grid search, which explores all the possible hyperparameter combinations, and the random search algorithms, which converge slowly until the global optima are found, to the more complex model-based algorithms [20].

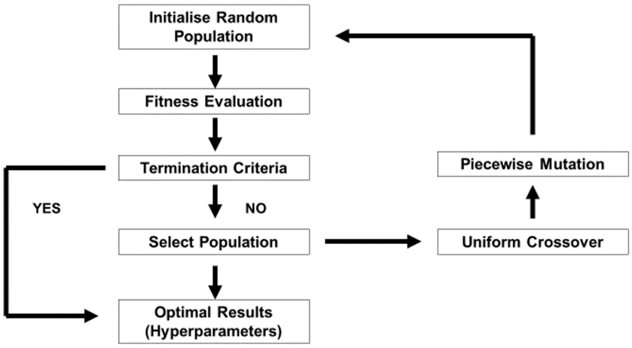


FIGURE 2. Basic process of the genetic algorithm (GA) adapted from [4].

Hyperparameters influence model underfitting or overfitting, affecting the final accuracy of the network [4]. The temporal nature of the chosen data dictates the need for the optimal number of LSTM units in hidden layers and time window size, which should contain an appropriate dataset background enabling the LSTM to learn enough from past information [19]. A large time window will lead to model overfitting, while a small window will neglect important information [10].

III. GENETIC ALGORITHM

Innately stochastic metaheuristic algorithms such as GA mimic observed natural behaviour to solve complex optimization problems within limited time and computational capacity. GA is known to lessen search complexity and to find optimal (or close to optimal) values for tunable LSTM hyperparameters [4]. Thus this paper suggests a hybrid solution integrating GA with the LSTM network [19] to find the optimal hyperparameters.

GA-optimisation of the LSTM network is initialised by a generated population of random individuals [4], where each individual represents a potential solution- hyperparameter values that are expressed as binary strings [19]. Individuals are arbitrarily selected from the search space and are evaluated by the fitness function defined before the optimisation process. Individuals with higher fitness functions are considered to be near-optimal solutions and are preserved for the next generation, while the rest are disregarded [10], replicating the natural evolution where stronger individuals are more likely to reproduce [4]. Figure 2 [4] illustrates the basic GA process with six distinct stages: initialization, fitness calculation, termination condition check, selection, crossover, and mutation [10].

Thus with every generation, the worst performing individuals are removed, and new individuals are added to the population to add genetic diversity to the evolutionary process through crossover, mutation, and selection genetic operators [4]. The crossover operator creates new individuals by replacing some of the portions of the two-parent individuals [4] by only utilising information in the search space, without generating new information [10]. The mutation operator generates new information by mutating portions of individual strings [10] to create unique individuals [4]. An inappropriate definition of the solution representation could lead to incorrect choice of mutation and crossover operators [4]. The selection operator effectively exploits the information accumulated through the GA search by selecting the stronger individuals as their offspring will have higher fitness (fitness function) and thus survive the next generation [4]. The generated individual goes through the process of selection, crossover, and mutation while calculating its fitness to the model and verifying the termination criteria. Once the termination criteria are satisfied, the process stops [10].

Iterative modifications and evaluation of the population allow GA to find optimal or near-optimal hyperparameters in a search space with many peaks and valleys while traditional gradient-based methods get stuck at the local optima. Also, diversity injected into the population enables the execution of a global search by allowing the exploration of new areas in the search space. Thus GA is the appropriate choice for combinatorial optimization where a thorough search of all the possible solutions is required and would demand enormous computational power [4].

IV. ENSEMBLE MODELS THROUGH WEIGHT-BASED FUSION

The inherent human approach to problem-solving that involves making informed decisions based on several inputs [21] forms the basis of ensemble learning [22]. Ensemble models are developed by combining a finite number of different neural networks to improve the overall prediction. Similarly, a human would combine the knowledge of several differently sourced opinions to make a final decision. Individual networks are independently trained, and their predictions are combined using a mathematical rule [23] to form a final single ensemble model prediction. Ensemble forecasting can alleviate challenges in time series forecasting where data is volatile, dynamic, and non-stationary such as in the geology, energy, water quality, and finance sectors, enhancing the forecasting accuracy instead of a single model forecast [24].

Weight-based ensemble approaches use weighting schemes for the ranking of features or models [25], such as weighted linear combination techniques. The weights allocated to each model can be equal, such as in the simple average ensemble model. They can also be determined through a mathematical rule, as with the weighted ensemble model. The most appropriate way to combine individual LSTM model forecasts is unknown and undecided as the research into LSTM applicability in ensemble forecasting is scarce [24]. This study chose to evaluate the combined forecast of the LSTM models through a linear function of the individual model forecasts [26]. These linear combinations can be calculated as follows from [26]: Let $\mathbf{Y}$ be the actual time series that will be forecasted using n different models, where

$$\mathbf{Y} = [y_1, y_2, \ldots, y_N]^T. \tag{7}$$

Let $\hat{\mathbf{Y}}^{(i)}$ be the forecast obtained from the i'th model ($i = 1, 2, \ldots, n$) where [26]:

$$\hat{\mathbf{Y}}^{(i)} = [\hat{y}_1^{(i)}, \hat{y}_2^{(i)}, \ldots, \hat{y}_N^{(i)}]^T. \tag{8}$$

A linear combination of these n forecasted series of the original time series can be shown as follows [26]:

$$\hat{\mathbf{Y}}^{(c)} = [\hat{y}_1^{(c)}, \hat{y}_2^{(c)}, \ldots, \hat{y}_N^{(c)}]^T \tag{9}$$

which is produced by:

$$\hat{y}_k^{(c)} = f(\hat{y}_k^{(1)}, \hat{y}_k^{(2)}, \ldots, \hat{y}_k^{(n)}) \quad \forall k = 1, 2, \ldots, N \tag{10}$$

where f is a linear function of the individual forecasts and which results in [26]:

$$\hat{y}_k^{(c)} = w_1\hat{y}_k^{(1)} + w_2\hat{y}_k^{(2)} + \ldots + w_n\hat{y}_k^{(n)} = \sum_{i=1}^{n} w_i\hat{y}_k^{(i)} \tag{11}$$

where w_i is the weight assigned to the i'th forecast. The added weights amount to unity [26]. All the models are assigned equal weights in the simple average approach [26]:

$$w_i = \frac{1}{n}. \tag{12}$$

For the weighted ensemble model, greater weight is assigned to the more skilled model, indicating greater trust in the model [23]. In contrast, all the models are allocated the same weight regardless of skill in the average ensemble, rendering it incapable of dealing with extreme values, such as outliers and skewed distributions [24]. These weights are small positive values. The sum of all the weight coefficients in the ensemble must be equal to one [23]. The more skilled model is emphasised in the weighted ensemble. Thus it is expected to perform better than the average ensemble [23].

Effective ensemble forecasting requires a considerable amount of diversity between individual LSTM models [24]. This study accounts for diversity in LSTM models through different time window sizes and as a result, a different number of LSTM units, which is advantageous for the modeling of highly non-linear statistical dependencies. Consequently, an ensemble model with LSTM based models of different time window sizes will be capable of handling the non-stationary and dynamic nature of real-world time series [24]. Tuning the hyperparameters of each LSTM based model in the ensemble will increase their quality, thus enhancing the overall quality of the ensemble [27] provided that a sufficient amount of diversity exists between each model. This study highlighted the weight-based approach for combining the two-hybrid GA-LSTM models.

V. WATER BODIES AND WATER QUALITY PARAMETERS

For the development of the models, data was taken from two water bodies (rivers), the Burnett river and the Baffle river.

The Burnett River is in southeast Queensland, Australia, named after J.C. Burnett, the first explorer of the river in 1847 [28]. The river rises on the western slope of the Burnett Range, east of the Eastern Highlands [28] at

Mount Gaete [29] and flows a 435 km course into the Coral Sea of the South Pacific Ocean at Burnett Heads [29]. The Auburn, Boyne, and Barambah rivers are tributaries to the river [28]. Small crops and sugar cane are grown in the areas around the river, which are part of the South East Queensland and Brigalow Belt bio-regions [30]. The Baffle Creek, also known as the Baffle river, was named by politician and pastoralist William Henry Walsh in the 1850s. He was unable to track down raiders through the dense bush along the banks of the creek, leaving him baffled, hence the name [31]. The river is in southeast Queensland, Australia, and flows a 124 km course from Arthurs Seat down into the Coral Sea of the South Pacific Ocean [32]. The tributaries into the river from the right are the Scrubby, Granite, Grevillea, and Three Mile creeks and the Euleilah and Island creeks from the left[33]. The Burnett and Baffle datasets included water quality parameters such as water temperature (°C), pH, electrical conductivity (mS/cm), dissolved oxygen (mg/L), and turbidity (NTU).

The dissolved oxygen (DO) concentration is the most relevant water quality parameter as it reflects the equilibrium between the oxygen-producing and consuming processes in a river [34]; it is the amount of free non-bonded, non-compound oxygen present in water [35]. DO is an obvious criterion of river health, as DO is directly or indirectly influenced by other water quality parameters, such as temperature, salinity, oxygen depletion, pH, turbidity, etc. [34]. River-dwelling organisms, such as fish, invertebrates, plants, and bacteria, use DO in water just as land organisms use atmospheric oxygen. Extreme DO levels can adversely impact water quality and harm aquatic life [35].DO levels range between 6 and 14 mg/L [36], with healthy rivers at 6.5 to 8 mg/L [37]. The monitoring and prediction of DO levels ahead of time with predictive models will aid in optimising water quality control measures [34] and is thus of paramount importance [35].

Water temperature is a physical property and a measure of the average thermal energy of the water [38]. The river temperature is dependent on four factors: the type and depth of the river, the environment surrounding the river, and the season of temperature recording [38]. There are no typical water temperature ranges that apply to all rivers. However, rivers have annual temperature patterns. Temperatures that deviate from the pattern should be viewed in context. Rivers and streams exhibit faster and greater temperature fluctuations than lakes and oceans. Observed seasonal temperatures across American rivers on average can be as low as between 1 to 4.5° C and as high as between 30 to 35° C [38]. Water temperature influences the physical and chemical properties of water. An increase in temperature will decrease the solubility of gases (such as oxygen) in water. Thus warmer waters hold less dissolved oxygen [35]. Other water quality parameters such as electrical conductivity, salinity, compound toxicity, water density, and pH are also affected [38].

The pH is a measure of the activity of the hydrogen ion (H+) in water. pH ranges from 0 to 14 and is reported as the reciprocal of the logarithm of the hydrogen ion activity. A river with a pH of 7 has 10^{-7} moles per liter of hydrogen ions [39]. River pH ranges from 6.5 to 8.5 and is slightly lower for groundwater pH at 6 to 8.5 [39].

TABLE 1. Typical values for water quality parameters.

Water Quality Parameter	Range
Dissolved oxygen	6 mg/L-14 mg/L [36]
pH	6.5-8.5 [39]
Temperature	1-4.5°C - 30-35°C [38]
Conductivity	0.05 mS/cm - 1.5 mS/cm [40]
Turbidity	1 NTU - 1000 NTUs [43]

Conductivity measures the ability of a river to pass an electrical current, which increases with the presence of inorganic dissolved solids with either a negative or positive charge in the water. Organic compounds have low conductivity in water [40]. The conductivity of rivers ranges from 50 to 1500 μmhos/cm (0.05 to 1.5 mS/cm). Some inland freshwaters have observed ranges from 150 to 500 μmhos/cm (0.15 to 0.5 mS/cm) and heavily polluted industrial waters at 10,000 μmhos/cm (10 mS/cm) [40].

Turbidity, an optical characteristic of water, is a measure of the relative clarity of river water [41]. Turbidity measures the light that is scattered by suspended particles when light is shone through the water sample [42]. Solid particles suspended in the water include sediment (clay and silt), a variety of microscopic organisms, fine inorganic and organic matter, algae, and plankton [41].

High turbidity lessens the light penetration of the water, altering the ecological productivity, habitat quality, and aesthetic value of the river, causing harm to fish and aquatic life. Pollutants, such as bacteria and metals, also attach themselves to suspended particles enabling further pollution [42]. At high turbidity levels, suspended solid particles absorb heat, causing an increase in water temperature, thereby causing a decrease in DO concentration. Suspended particles also reduce the sunlight penetrating the river, inhibiting the photosynthesis of river plants, hindering DO production [43]. Turbidity levels can range from 1 to 1000 NTUs. Lower values indicate low turbidity and healthier rivers [43]. On average, river turbidity levels usually range from 10 to 25 NTUs [41].

The observed typical values for each water quality parameter has been summarised in Table 1.

VI. A ROBUST AND TOLERANT WATER QUALITY PREDICTION LSTM BASED ENSEMBLE SCHEME

The methodology used to develop a robust and tolerant water quality prediction GA-optimised LSTM based ensemble scheme is described below through a sequence of steps and is illustrated in Figure 3.

A. WATER QUALITY DATASETS AND DATA PREPARATION

Two water quality datasets were used to develop two LSTM models. The historical water quality data was publicly available from the Ambient Estuarine Water Quality Monitoring Programme on the Queensland Government open data

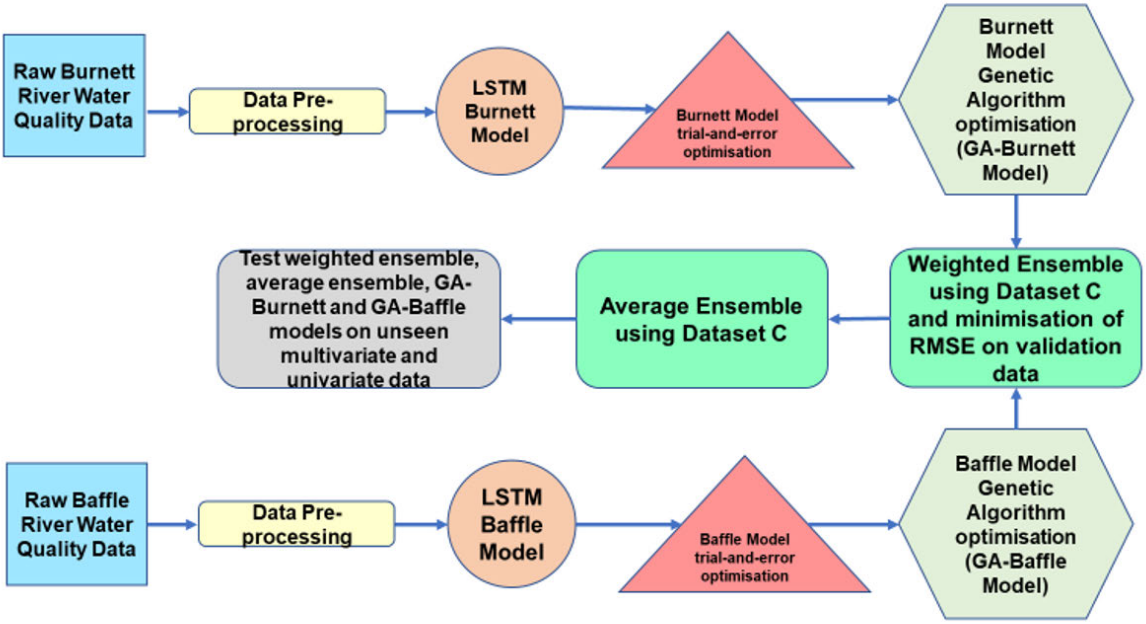

Raw Burnett River Water Quality Data
Data Pre-processing
LSTM Burnett Model
Burnett Model trial-and-error optimisation
Burnett Model Genetic Algorithm optimisation (GA-Burnett Model)
Weighted Ensemble using Dataset C and minimisation of RMSE on validation data
Average Ensemble using Dataset C
Test weighted ensemble, average ensemble, GA-Burnett and GA-Baffle models on unseen multivariate and univariate data
Raw Baffle River Water Quality Data
Data Pre-processing
LSTM Baffle Model
Baffle Model trial-and-error optimisation
Baffle Model Genetic Algorithm optimisation (GA-Baffle Model)

FIGURE 3. Schematic diagram of the development of a robust and tolerant water quality prediction LSTM based ensemble model.

portal [44]. The Burnett river (dataset A) consists of data from the beginning of January 2017 to the end of December 2019, with 48 observations per day, 17 520 observations per year, and thus 52 560 observations over three years. The Baffle river (dataset B) consists of data from the beginning of December 2018 to the end of November 2019, with 144 observations per day and thus 52 560 observations per year.

The data was pre-processed and cleaned in accordance with the following steps, prior to model development (yellow block in Figure 3):

1) Only highly problematic inconsistent observations, such as a negative pH were removed, etc., were strategically removed using Table 1 as a reference as certain incongruous values could represent a reality that should be recorded.
2) Statistical analysis through evaluation of the minimum, maximum, mean, standard deviation and the first, second, and third quartile of each parameter was used to remove outliers along with the interquartile range (IQR), found using the upper boundary Q_3 i.e. (75^{th} percentile) and lower boundary Q_1 i.e. (25^{th} percentile). of each parameter and shown in (13) [45]

$$\mathrm{IQR} = Q_3 - Q_1. \tag{13}$$

3) Duplicate observations were removed to prevent repetition from distorting results.
4) Observations at irregular time intervals were removed, while missing values at regular intervals were calculated through interpolation.

The correlation between the parameters was evaluated using Spearman's Correlation to identify which parameters could be inputs for the predictive models. Correlation measures the association between two parameters and is expressed as a value between −1 and 1. When there is a positive correlation (closer to one), one parameter increases as the other parameter increases. When there is a negative correlation (closer to negative one), one parameter increases as the other parameter decreases. When the correlation is neutral (close to zero), there is no relationship between the parameters. A strong correlation indicates a strong relationship between two parameters, implying they can be used to build a model [6], [46]. The Spearman's coefficients of each parameter in relation to DO (target feature) are shown below in Table 2.

TABLE 2. Spearman's correlation coefficients.

Parameters	Burnett Data	Baffle Data
Temperature	-0.437	-0.419
pH	0.361	-0.055
Dissolved oxygen	1.00	1.00
Electrical conductivity	-0.080	-0.075
Turbidity	-0.129	-0.100

DO shares the strongest negative correlation with temperature for both datasets as shown in Table 2, thus as water temperature increases, DO levels will decrease. The correlation between DO and electrical conductivity and turbidity is insignificant for both datasets. In contrast, pH behaves differently in each dataset and has a moderate positive correlation with DO in the Burnett dataset but a weak negative correlation in the Baffle dataset. Thus highlighting the difference between causation and correlation, which are not equitable and are often confused when working with time-series data [47]. A causal relationship exists between two parameters when three requirements are met: an association between the parameters, appropriate time order, and the elimination of other parameters [48], under experimental conditions. A causal stimulus can also be tested. If manipulation of a parameter causes a sufficient change in the other parameter, a causal relationship can be established [49].

Any assumed causation between pH and DO was disregarded due to the different correlation values in datasets and no documented causal relationship between the parameters at the time of this study. Thus DO and the water temperature were chosen as input parameters for the models. The other parameters were removed, and new Burnett and Baffle datasets only containing the input parameters were created.

B. DEVELOPMENT OF A MULTIVARIATE MULTI-STEP STACKED LSTM MODEL

Two water quality predictive LSTM models (brown circle in Figure 3) were developed and evaluated using the free open source publicly available Keras Python library and were defined by efficient numerical libraries using Tensorflow backend, in Google Colab (Google Colaboratory), a hosted Jupyter notebook service which executes python code through the browser [50].

The Burnett model was developed as a multivariate multi-step stacked LSTM model. Models are defined as a sequence of layers in Keras. DO and water temperature, in the input layer, are used to predict the target parameter, DO. The model has two hidden layers. The first hidden layer has more LSTM units than the second hidden layer. The dense output layer connects the whole model and outputs a multi-step prediction of DO values 24-time steps ahead. Root Mean Square Propagation (RMSprop) is the chosen optimiser, and Rectified Linear Unit (ReLU) is the activation function. The Baffle model was developed with a similar architecture, using the same two input features, two hidden layers (the first layer has more LSTM units than the second layer) and a dense output layer predicting DO values 24-time steps ahead, with RMSprop and ReLU as the optimiser and activation function respectively. The Baffle model has more LSTM units than the Burnett model due to the structural nature of the different datasets.

The Baffle and Burnett datasets both have a total of 52 560 observations. The Burnett dataset has the observations spread across three years, while the Baffle dataset has the observations spread across a single year. Hence it is easier for the LSTM network to pick up variations in the Burnett data and learn trends over three years. Therefore an LSTM model with an overall smaller architecture was used for the Burnett model. The Baffle data is denser, with many close clusters of similar points with little variation between them, increasing the difficulty of learning a trend to make a prediction. Thus a larger LSTM was used to accommodate the densely spaced Baffle dataset.

ReLU is linear for positive inputs, retaining linearity properties when training neural networks with backpropagation and behaves like a nonlinear function when the input is negative and outputs a value of zero [6], [15]. Thus still affording hidden layers the opportunity to be activated, while sigmoid and tanh functions can only approximate values close to zero but not zero [15]. Other advantages of the ReLU include representational sparsity, linear behaviour, and computational simplicity and implementation at a lower computational cost as it excludes the computation of an exponential function, unlike the sigmoid and tanh functions [6], [51].

The Root Mean Square Propagation (RMSprop) optimiser is an adaptive learning rate method developed to address the problem of drastically diminishing learning rates observed when using the Adagrad optimiser [52].

The pre-processed data was split into the training, validation, and testing datasets in a 50%, 20%, 30% ratio, respectively. The training dataset was the largest to ensure there were enough samples for the LSTM network to learn from. Water quality parameters with different ranges and scales were normalised before the data was fed to the models. Normalisation reduced model training difficulty and ensured the model was independent of input unit choice. The mean ($\overline{x}$) and standard deviation (σ_x) of the original data (x) were used to calculate the normalised data (x') in both the training and validation datasets, in accordance to (14) [27]

$$x' = \frac{x - \overline{x}}{\sigma_x}. \tag{14}$$

The test dataset was not normalised, allowing for the generality of the model to be assessed, thus enabling the development of a better predictive algorithm. If the test data is normalised with the entire dataset, the testing process will validate the model and not assess model generality [27]. Model predictions were scaled back to the original scale for each parameter by reversing the normalisation calculation in (14) before the evaluation of performance metrics.

C. TRIAL-AND-ERROR OPTIMISATION OF THE LSTM MODEL

The LSTM models were initially optimised in terms of time window size and the number of LSTM units in the two hidden layers using trial-and-error (orange triangles in Figure 3).

Three arbitrary values were chosen for the time window size, the number of LSTM units in the first hidden layer, and the number of LSTM units in the second hidden layer, and as previously stated, there are more units in the first hidden layer than the second layer.

The LSTM model was trained and a prediction was made by the model. The RMSE value was calculated to evaluate the accuracy of the model prediction. The smaller the RMSE value, the more accurately the model can predict DO from the historical data. In the second run, the three values were arbitrarily lowered. The process was repeated, and the RMSE was recorded. In round three, the values were arbitrarily increased. The process was repeated, and the RMSE was recorded. Thereafter, arbitrary combinations of the number of LSTM units in the two hidden layers were chosen, while the time window size remained constant. The process was repeated, and the RMSE was recorded. The time window size was then changed while the LSTM units remained constant, and the same procedure was followed.

The behaviour of the model was observed with each change and influenced the choice of the values chosen for the subsequent run. If lower RMSE values were achieved at small time window sizes, then consistently lower values were chosen for the window size, until the RMSE values started increasing again at the smaller window size values. This strategy was also used to find the number of LSTM units and the combination of the time window size and LSTM units that would produce the lowest RMSE value.

The hyperparameter values for the Burnett model found through trial and error were 32 LSTM units in the first hidden layer, 16 LSTM units in the second layer, and a time window size of 100-time steps. The Baffle model values were 64 LSTM units in the first hidden layer and 32 LSTM units in the second hidden layer, with a time window size of 150-time steps. As expected, the Baffle LSTM model network architecture is larger than the Burnett model, with a bigger window size and almost double the amount of LSTM units.

D. HYBRID GENETIC ALGORITHM OPTIMISED LSTM BURNETT AND BAFFLE MODELS

The Burnett and Baffle LSTM models were optimised with GA utilising the "Distributed Evolutionary Algorithms in Python" package, referred to as DEAP [53], using Keras and Tensorflow. The models were optimised separately. The hyperparameter values found through the trial-and-error process were used as the initial values for the GA-optimisation process shown in Figure 4 [10]. The GA-optimisation process was used to find the optimal time window size and the optimal number of units in the first and second hidden layers for each model (light green hexagon in Figure 3). Genetic parameters were specified using the DEAP package, such as a population size of 70, mutation rate of 0.15, crossover rate of 0.7, and the number of generations at 10. These values were chosen for similar studies [10]. Ordered crossover, shuffle mutation, and roulette wheel selection, were also chosen and specified using DEAP as they produced the best possible results from the available options [53].

During the optimisation process, the search space was explored by the genetic operators, and the population became composed of possible solutions (optimal hyperparameter values), in the form of chromosomes encoded by binary bits, which represent the number of the LSTM units and the time window size [10]. The binary solution was of length 14. The first six digits were for the time window size, the subsequent four digits were for the number of LSTM units in the first hidden layer, and the last four digits were for the number of LSTM units in the second hidden layer. The selection and recombination operators search for the best solution within the solution-composed population. Each solution is evaluated with the predefined fitness function, and the solution with the best performance is chosen for reproduction [10].

Defining the fitness function before the optimisation process is crucial. The RMSE value was used to evaluate the fitness of each solution [10] and had to be minimised for optimisation to occur. DEAP was used to define the Fitness Maximum as −1.0 for minimisation [53]. The chromosome that provided the combination of hyperparameter values that resulted in the lowest RMSE value was considered the optimal or near-optimal solution. The termination criteria were satisfied by the optimal solution. The optimal solution was implemented by the LSTM model, to create the GA-optimised LSTM version of the model, which could now be used to make a prediction. If the termination criteria were not satisfied, the cycle of selection, crossover, and mutation would continue until the optimal solution was found [10]. The pseudo-code of the GA-Burnett and GA-Baffle models is shown in Algorithm 1.

Algorithm 1 GA-Optimised LSTM Model

```
Split the data into training (30%), validation (50%) and test (20%) data.
The LSTM is evaluated on the validation data.
Initialise the population size (70), the number of generations (10) and the length of the chromosome at 14 (binary style).
Set RMSE as the fitness function.
if $timewindowsize == 0$ or $numberofunitsA == 0$   ▷ number of units in first hidden layer
   or $numberofunitsB == 0$   ▷ number of units in second hidden layer then
    Probability of 0.15 for mutation of new chromosomes;
    Probability of 0.7 for crossover of chromosomes;
    Evaluation of the freshly generated chromosome through use of the fitness function;
    return RMSE of 1000   ▷ Stopping condition as minimisation of RMSE is the aim
end if
Choose the best individual chromosome which represents the optimal time window, the optimal number of units in the first hidden layer and the optimal number of units in the second layer.
Apply the optimal window size and optimal number of units in the two hidden layers in the LSTM and make a prediction on the unseen test data.
```

The optimal time window size for the hybrid GA-optimised Burnett and Baffle LSTM models were 57 and 63-time steps, respectively. The optimal LSTM units in the first hidden layer for the GA-Burnett and GA-Baffle models were 10 and 12 LSTM units, respectively, and 8 and 10 LSTM units in the second layer for the GA-Burnett and GA-Baffle models, respectively. Again, the Baffle model has an overall larger architecture than the Burnett model. These values are summarised in Table 3.

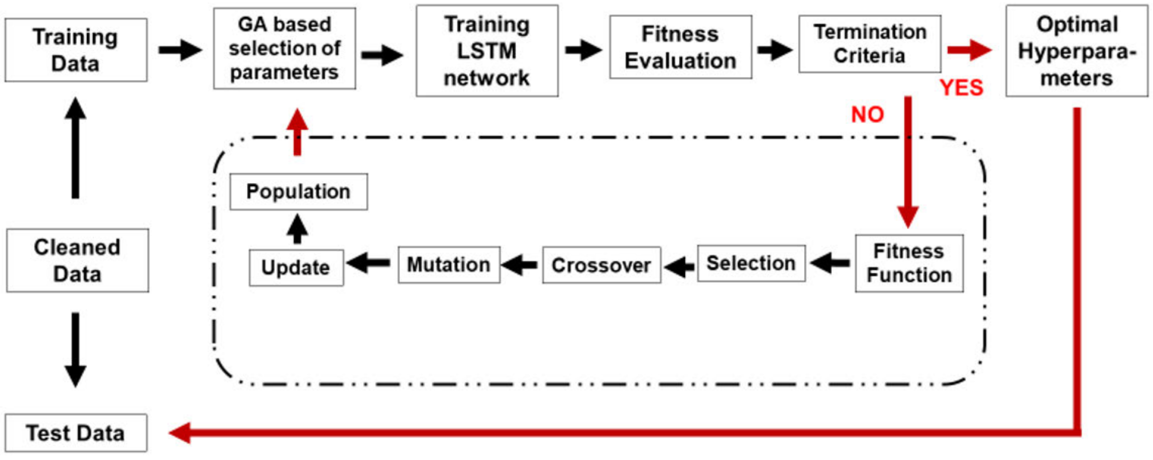


FIGURE 4. Illustration of the optimisation of LSTM by GA adapted from [10].

TABLE 3. Hyperparameter values for LSTM models.

	Burnett	GA-Burnett	Baffle	GA-Baffle
Window Size	100	57	150	63
Units 1st layer	32	10	64	12
Units 2nd layer	16	8	32	10

E. ENSEMBLE LSTM-BASED MODEL THROUGH A WEIGHT-BASED TECHNIQUE

The linear weight-based technique (dark green rectangles in Figure 3) combined the predictions made by the models by allocating a weight to the prediction made by each model, which was indicative of the contribution the model made to the new ensemble model. The ensemble model was created using dataset C, a Baffle river (different to the Baffle dataset 2019 used for the Baffle model) water quality dataset beginning from January 2015 and ending in December 2015, 48 observations per day and a total of 17 520 observations for the year. The process of the development of the weighted and average ensemble models from the two GA-optimised LSTM base models, GA-Burnett model, and GA-Baffle model is detailed below and was inspired by [54]. An illustration of the process is shown in Figure 5.

The Baffle 2015 dataset was divided into the training, validation, and testing datasets at 30%, 50%, and 20%, respectively. The holdout validation dataset, which was unseen by both models during the training process, was larger than the training dataset as it was used to estimate the optimal weight contribution of each model in the ensemble model. The validation dataset had to be large and representative to prevent the model from over-fitting.

GA-Burnett and GA-Baffle models defined according to their optimal hyperparameters were trained on the training dataset and made predictions using the validation dataset as the test dataset. The predictions were evaluated through comparison to the actual DO values by calculating the RMSE value. A lower RMSE value indicated greater model performance.

The optimal weight coefficients for each model prediction were found through an exhaustive grid search, comprised of weight coefficients starting from 0.0, increasing increments of 0.1, and ending in 1.0. The sum of the two possible weight coefficients must be equal to one. The weight coefficient values were multiplied by the GA-Burnett and GA-Baffle model predictions until the optimal weight combination was found, through a function that minimises the RMSE value. The validation dataset, which was unseen by the individual models, was used to simultaneously perform the grid search to find the optimal weight combination, while each model made a prediction on the validation dataset, with RMSE minimisation as the final goal. The performance of the models was compared to each other and the ensemble models, using the RMSE value. The optimal weight combination represented the extent of the contribution of each model to the final weighted ensemble model. The weighted ensemble was evaluated on the test dataset.

In an average ensemble model, each model contributes equally to the ensemble prediction. Thus the continuous-valued output is the average of the individual member predictions. As there are only two members in the ensemble, the weight coefficient of each model is 0.5. The more skilled model has a larger weight coefficient than the less skilled model in a weighted ensemble model. The average ensemble will be compared to the weighted ensemble. A well-configured weighted ensemble model is expected to outperform an average ensemble model. The pseudo-code for the ensemble model is shown in Algorithm 2.

F. PERFORMANCE METRICS

The ensemble models were assessed by computing common performance metrics for continuous output models, such as the Mean Squared Error (*MSE*), Mean Absolute Error (*MAE*), Mean Absolute Error Percentage (*MAPE*) and the

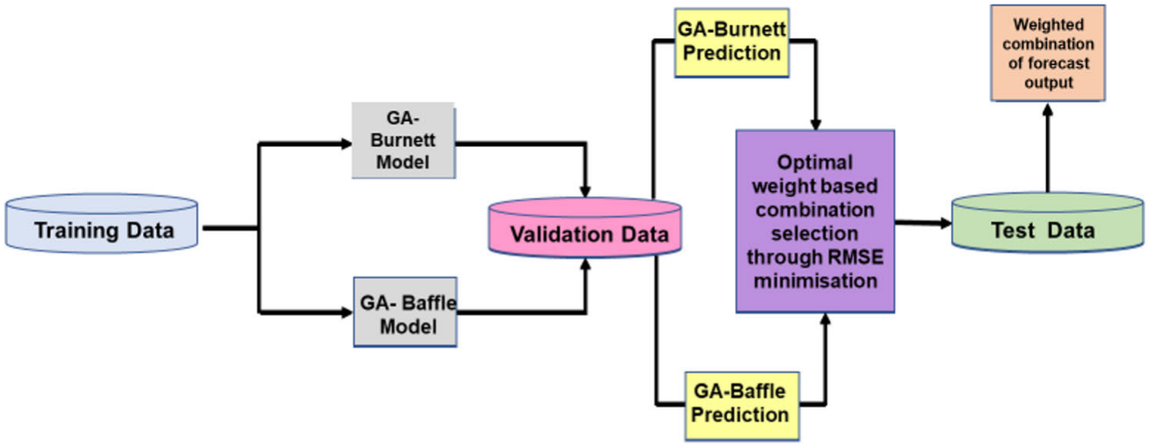


FIGURE 5. Illustration of the development of ensemble models through a linear weight-based technique.

Algorithm 2 Ensemble Model

```
Split the data into training (30%), validation (50%) and test (20%) data;   ▷ validation data must be the biggest dataset
The individual model weight combinations are evaluated on the validation data.
Fit the GA-Burnett model on training data;
Fit the GA-Baffle model on training data;
Make a prediction with each model using the validation data.
Create a grid of weights from 0.0 to 1.0 with increments of 0.1.
Define a function which multiplies a weight from the grid with the prediction from each model and then sums the product of the associated model weight and prediction for each model to make a final prediction:
$pred_{ensemble} = w_a * pred_a + w_b * pred_b$   ▷ a, b = GA-Burnett, GA-Baffle
Define a function to evaluate the ensemble prediction by calculating the RMSE score with the ensemble prediction and the true values of the validation data;
while $w_a + w_b = 1$ and $w_a, w_b > 0$ do
minimise RMSE score for the weighted ensemble
end while
The best weight for each model will form the weight combination that gives the lowest RMSE on the validation data
The best weight combination for the weighted ensemble can be used to make a prediction on the unseen test data
For the average ensemble, $w_a$ and $w_b$ are equal i.e. 0.5
The average ensemble is also used to make a prediction on the unseen test data
```

Root Mean Squared Error (*RMSE*). The *MSE* (15) [10] is a measure of average squared difference between the predicted values and the actual values [55]. The smaller the *MSE*, the more accurate the model prediction:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2. \tag{15}$$

where n is the number of samples, y_i is the desired output and $\hat{y}_i$ is the predicted output value of the observation made by the model i^{th}. The *MAE* (16), *MAPE* (17) and *RMSE* (18) were evaluated in accordance to the following equations [10]:

$$MAE = \frac{\sum_{i=1}^{n}|y_i - \hat{y}_i|}{n}. \tag{16}$$

The *MAE* defined in (16) is the average of the absolute difference between the predicted values and the actual values [56]. The smaller the *MAE* value, the closer the predicted values are to the actual values.

$$MAPE = \frac{\sum_{i=1}^{n}|(y_i - \hat{y}_i)/y_i|}{n} \times 100. \tag{17}$$

The *MAPE* defined in (17) is the average of the absolute percentage of the difference between the estimated values and the actual values, providing a measure of the error in a comprehensible percentage. The smaller the *MAPE*, the better the forecast [57].

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}. \tag{18}$$

The *RMSE* value defined in (18) is the quadratic average of the differences between the predicted and actual values and measures the accuracy of a model on a dataset and not between datasets as it is scale dependent [58]. Outliers disproportionately affect the *RMSE* value [59]. A good prediction has a low *RMSE*. *RMSE* value of zero indicates a perfect fit between model and data [58]. The R^2 score defined by (19) [56] is the coefficient of determination [60] and indicates how well the model fits the data, thus measuring how well unseen data is likely to be predicted by the model. Here $\bar{y}$ is the average of the actual values [56]:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)}{\sum_{i=1}^{n}(y_i - \bar{y})}. \tag{19}$$

The Median Absolute Error, defined by *MedAE* in (20) [56] is a measure of the error and finds the median of all the absolute differences between the predicted and actual values [56]:

$$MedAE = Median(|y_1 - \hat{y}_1|, \ldots, |y_n - \hat{y}_n|). \tag{20}$$

The maximum error defined by *MaxError* in (21) [56], calculates the maximum residual error by capturing the worst error between the predicted value and the desired outcome [56]:

$$MaxError = max(|y_i - \hat{y}_i|). \tag{21}$$

The explained variance defined by the *EV* score in (22) [56], indicates how well the model accounts for the variation of a dataset. The closer the score is to one, the better the performance of the model. *Var* is variance, the square of the standard deviation [56]:

$$EV = 1 - \frac{Var\{y - \hat{y}\}}{Var\{y\}}. \tag{22}$$

G. ASSESSMENT OF MODELS ON UNSEEN DATA

The four models: the GA-Burnett, GA-Baffle, average ensemble, and weighted ensemble, were tested on three multivariate and one univariate publicly available online datasets.

Wind Power Forecasting Dataset [61] includes seven key columns labeled "wp1" to "wp7" for the wind power measurements of seven different wind farms. As the data were normalised by the data provider, the original values, units, and scale are unknown. The data is publicly available and visible on the Kaggle: Machine Learning and Data Science Community as part of their Global Energy Forecasting Competition 2012 on Wind Forecasting. The Air Temperature Dataset is referred to as the Jena Climate Dataset on Kaggle [62] and includes various meteorological parameters along with air temperature, such as air pressure, air density, etc., which are used to predict the air temperature. The Pollution Dataset, referred to as the Beijing PM2.5 Data dataset [63] on Kaggle, includes parameters such as temperature, pollution level, pressure, dew point, wind speed, and direction, etc., which is used to predict the future pollution level in terms of PM2.5 (atmospheric particulate matter (PM) that has a diameter less than 2.5 micrometers) [64]. The daily minimum temperature in Melbourne is a univariate dataset [65] containing the single temperature parameter used to predict the future minimum temperature. It was viewed on Kaggle while the original dataset was hosted by the Data Market Qlik Sense Data Sources.

Comparison is one of the best performance measures. Thus the predictive capability of four classical time series forecasting methods, the Autoregression (AR), Moving Average (MA), Autoregressive Moving Average (ARMA), and Autoregressive Integrated Moving Average (ARIMA) was compared to that of the four models on the univariate dataset.

TABLE 4. Summary of datasets.

Dataset	Parameter	Samples	Mean	[a]Std.Dev
Burnett	DO mg/L	52 560	6.58	0.88
Baffle 2019	DO mg/L	52 560	6.77	0.96
Baffle 2015	DO mg/L	17 520	6.79	0.97
Air Temperature	Temp °C	420 551	9.45	8.42
Pollution	PM2.5 ug/m^3	43 800	94.01	92.25
Minimum Temp	Temp °C	3650	11.18	4.07
[b]Wind Power	Wind Power	18 757	-	-

[a]Standard Deviation
[b]External normalisation thus values unavailable

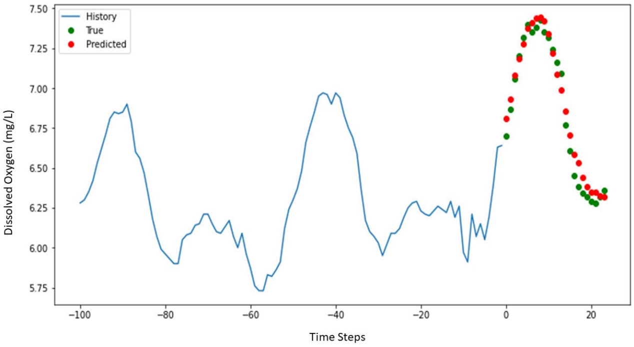


FIGURE 6. Performance of LSTM Burnett model.

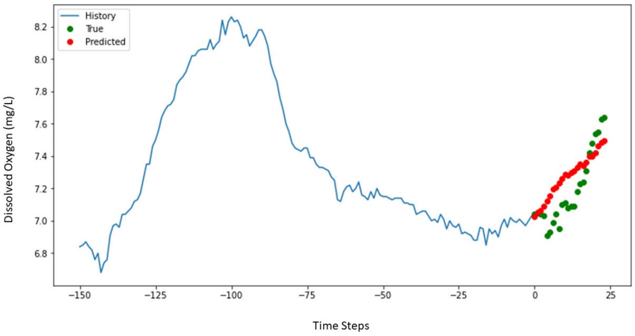


FIGURE 7. Performance of LSTM Baffle model.

A summary of all the datasets used in this study is presented in Table 4. As the wind power dataset was normalized externally and the original values were unknown, the mean and standard deviation could not be calculated.

VII. RESULTS AND ANALYSIS

A. TRIAL-AND-ERROR OPTIMISED BURNETT AND BAFFLE LSTM MODELS PREDICTIVE CAPABILITY

The Burnett model in Figure 6 has a greater ability to predict DO concentration 24-time steps ahead than the Baffle model in Figure 7, from historical water temperature and DO data (blue line) as seen from the well-aligned red points (predicted DO values) and green points (actual DO values). The green and red points in Figure 7 are hardly aligned and only intersect at one point. Table 5 supports this notion with a detailed comparison of model predictive ability in terms of the performance metrics.

TABLE 5. Performance metrics of GA-Burnett, Burnett, GA-Baffle and Baffle LSTM models.

Performance Metric	GA-Burnett	Burnett	Baffle	GA-Baffle
$RMSE$ mg/L	0.25	0.28	0.99	0.57
MAE mg/L	0.17	0.18	0.65	0.42
MSE (mg/L)^2	0.06	0.08	0.98	0.33
$MAPE$ %	2.5	2.7	9.4	6.11
R^2	0.84	0.80	0.36	0.79
EV	0.84	0.81	0.47	0.82
$MaxError$ mg/L	0.22	0.14	0.21	0.59
$MedAE$ mg/L	0.12	0.13	0.40	0.39

The hyperparameters of the Burnett and Baffle LSTM models optimised through the trial-and-error process are in Table 3. The Baffle model *RMSE* value of 0.99 is almost four times greater than the Burnett model at 0.28. Similarly, the Baffle model *MAE* is 0.65, while the Burnett model *MAE* is 0.18. The Baffle model *MSE* value is 0.98, 12 times greater than the Burnett model at 0.08. The *MAPE* value for the Burnett model is 2.7 and 9.4 for the Baffle model, approximately three times bigger. The performance metrics indicate a smaller difference between the predicted and actual DO Burnett values than the difference between predicted and actual DO Baffle values.

The Burnett model has significant predictive capability compared to the Baffle model, achieving a R^2 score value of 0.80, while the Baffle model scores half that value at 0.36. The Burnett model is also more capable of accounting for variation in data with an explained variance (EV) score of 0.81, while the Baffle model scored 0.47. The maximum error from the Burnett model is 0.14 and is less than the 0.21 of the Baffle model. The median absolute errors ($MedAE$) are 0.13 and 0.40 for the Burnett and Baffle models, respectively. The maximum error ($MaxError$) of the Burnett model is close to the median absolute error, while the median absolute error of the Baffle model is greater than the maximum error, further reinforcing the superiority of the Burnett model.

The Burnett model performs better, despite having a smaller time window size of 100-time steps than the Baffle model with 150-time steps per window. The Burnett model is fed data over a longer period of three years than the Baffle model, as is evident by the three peaks in the graph shown as a repeated trend in Figure 6. The Baffle model is fed denser spaced data over a single year which is evident from the single peak in the graph in Figure 7. The greater period of data allowed for the greater diversity of data and the repetition of the same trend three times. That enabled the Burnett model time window to effectively capture the appropriate dataset background, resulting in a better trained Burnett model with a greater predictive capacity than the Baffle model.

B. HYBRID GA-OPTIMISED LSTM BURNETT AND BAFFLE MODELS PERFORMANCE CAPACITY

1) GA-BURNETT AND BURNETT LSTM MODELS

The GA-Burnett model in Figure 8 shows the predicted DO values 24-time steps ahead with a time window size of 57-time steps of historical DO data (blue line), almost half the size of the previous Burnett model. The number of parameters to be trained by the Burnett model decreased from 32 to 10 LSTM units in the first hidden layer and from 16 to 8 LSTM units in the second hidden layer. These hyperparameter values are under GA-Burnett and GA-Baffle headings in Table 3. The graph in Figure 8 is similar to the graph in Figure 6, where the red and green points (predicted and actual DO values) align at numerous instances. This similarity of model performance is mirrored by Table 5. The improvement in the performance of the Burnett model after GA-optimisation, resulting in the GA-Burnett model was notable as shown by the observable difference between the performance metric values achieved by the Burnett and GA-Burnett models on the same dataset in Table 5.

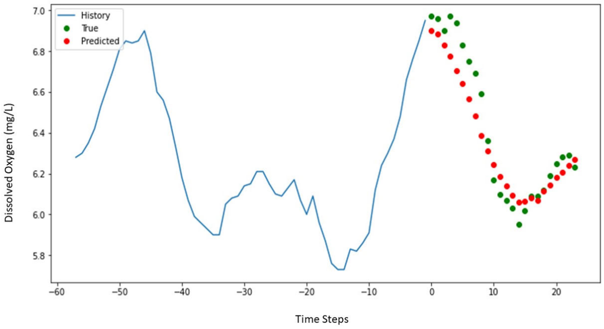


FIGURE 8. Performance of GA-optimised LSTM Burnett model.

The GA-Burnett model has *RMSE*, *MAE*, *MSE* and *MAPE* values of 0.25, 0.17, 0.06 and 2.5 respectively. These values are slightly lower than the Burnett model. After GA-optimisation, the *RMSE* value of the Burnett model decreases by 10.71%. The *MAE* value is reduced by 5.55%, whilst the *MSE* is decreased by 25%. The *MAPE* value is reduced by 0.2%.

The GA-Burnett model has a bigger maximum error than the Burnett model at 0.22, but the median absolute error of the GA-Burnett model at 0.12 is similar to the Burnett model, implying that the maximum error of the GA-Burnett is due to an outlier prediction. Both the R^2 score and explained variance score of 0.84 are only slightly greater than the Burnett model. The R^2 score of the Burnett model increased by 5% and explained variance score improved by 3.7% after GA-optimisation.

There is a notable yet relatively small change in predictive ability, despite the obvious GA-optimisation of the Burnett model. The Burnett model may have already been optimised to a great extent before the GA-optimisation, as shown by the good performance metric values in Table 5 and the close alignment of the green and red points in Figure 6, and any further optimisation would noticeable improve but not greatly alter the predictive ability of the model.

2) GA-BAFFLE AND BAFFLE LSTM MODELS

The performance of the Baffle model has significantly improved after GA-optimisation, as illustrated by the performance of the GA-Baffle model in Figure 9. There is

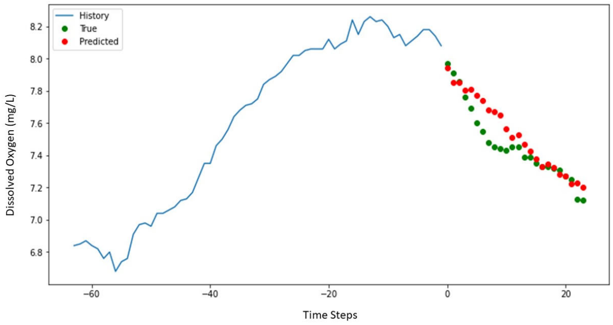


FIGURE 9. **Performance of GA-optimised LSTM Baffle model.**

a closer alignment of the red and green points in the graph, compared to the green and red points that only intersected once in Figure 7 for the Baffle model.

This improved performance between the Baffle and GA-Baffle LSTM models is illustrated by the substantial difference in performance metrics shown in Table 5. The GA-Baffle model had obtained *RMSE*, *MAE*, *MSE*, and *MAPE* values of 0.57, 0.42, 0.33, and 6.11, all of which are significantly lower than the Baffle model. After GA-optimisation, the *RMSE* value of the Baffle model decreased by 42.42%. The *MAE* was reduced by 35.38%, while the *MSE* was decreased by 66.32%. The *MAPE* was reduced by 3.29%.

The most evident improvement is the twice as large R^2 score from 0.36 to 0.79. The R^2 score improved by 119.44%. The explained variance score had also doubled from 0.47 to 0.82, showing an improvement of 74.47% by the Baffle model after GA-optimisation. The GA-Baffle model fits the data better, resulting in greater predictive capability and capacity to account for the variation in the data.

The median absolute error of the GA-Baffle model is very similar to the Baffle at 0.39. The maximum errors are quite different with the GA-Baffle model at 0.59 and the Baffle model at 0.21. The maximum error of the Baffle model is half the size of the absolute median error, showing the inferior performance of the Baffle as the maximum error of the GA-Baffle is probably due to an outlier.

The prominent performance improvement that existed between the Baffle and GA-Baffle models, was not present between the Burnett and GA-Burnett models. After optimization, the time window size for the Baffle model changed from 150-time steps to 63-time steps, less than half the size it was. The units in the first and second hidden layers were reduced from 64 to 12 and from 32 to 10 units, respectively. The architecture of the GA-Baffle model is much smaller than that of the Baffle model. The difference in the architecture of the Baffle model after GA-optimisation is greater than the difference observed by the Burnett model and is responsible for the significant performance improvement.

3) GA-BURNETT AND GA-BAFFLE LSTM MODELS

The Burnett model has performed better than the Baffle model every time, primarily due to the structure of the dataset used to train the Burnett model. The already well-performing Burnett model was only slightly optimised by GA, while the Baffle model was greatly optimised by GA but not enough for the GA-Baffle model to outperform the GA-Burnett model as shown by the values in Table 5.

4) GA-OPTIMISED LSTM MODELS AND LSTM MODELS

Table 5 summarises the performance metrics of all four models alongside one another for comparison. The Baffle model has the largest and the worst *RMSE*, *MSE*, *MAE*, and *MAPE* values, while the GA-Burnett model has the lowest and the best values for these performance metrics. The GA-Burnett, Burnett, and GA-Baffle models all have similar R^2 scores, around 0.8. Similarly, all their explained variance scores are also approximately 0.8. The Baffle model has a much lower R^2 score of 0.36 and explained variance score of 0.47, almost half the value of the other models. The Baffle model only fits the data and accounts for the variation in the data half as well as the other models.

The GA-optimised models have larger maximum errors than their original counterparts. The median absolute errors of the GA-optimised models are smaller than their maximum errors. Despite producing more accurate and consistent predictions, the GA-optimised models seem to be prone to outlier predictions. Overall, the GA-Burnett model shows the best predictive capability, while the Baffle model has the worst performance from all four models. In general, both GA-optimised models show improved predictive performance compared to their respective original counterparts. This performance improvement is representative of the improved robustness of the original LSTM models.

C. ENSEMBLE MODEL PREDICTIVE CAPABILITY

The GA-Burnett and GA-Baffle models contributed equally to the average ensemble model. An optimal weight combination of 60% of the GA-Burnett model and 40% of the GA-Baffle model was found for the weighted ensemble model. There is only a 10% difference in the weight combination of the ensembles. The optimal weight combination did not deviate much from the equal weight combination after the exhaustive grid search. This was largely due to the similarities in the architecture of the two GA-optimised LSTM based models. The only architectural difference between the base models is the number of units in the hidden layers. The GA-Burnett and GA-Baffle models have 1344 and 1904 trainable parameters, respectively, amounting to a difference of 560 parameters and 6 minutes of computation time.

1) AVERAGE AND WEIGHTED ENSEMBLE MODELS

The negligible difference in the performance of the ensemble models is shown in Table 6. The average ensemble achieved *RMSE*, *MSE*, *MAE* and *MAPE* values of 0.188, 0.034, 0.129 and 2.093, respectively while the weighted ensemble achieved similar *RMSE*, *MSE*, *MAE* and *MAPE* values of 0.186, 0.035, 0.129 and 2.083, respectively. The ensemble models have similar R^2 scores for the average ensemble at

TABLE 6. Performance comparison of average and weighted ensemble models and the GA-optimised LSTM models.

Metric	Average	Weighted	GA-Burnett	GA-Baffle
$RMSE$ mg/L	0.188	0.186	0.204	0.198
MSE (mg/L)^2	0.034	0.035	0.042	0.039
MAE mg/L	0.129	0.129	0.139	0.147
$MAPE$ %	2.093	2.083	2.239	2.421
R^2	0.879	0.878	0.853	0.862
EV	0.88	0.879	0.858	0.871
$MaxError$ mg/L	0.31	0.29	0.302	0.499
$MedAE$ mg/L	0.09	0.09	0.096	0.114

0.879 and weighted ensemble at 0.878 and similar explained variance scores for the average ensemble at 0.88 and weighted ensemble at 0.879.

The maximum error of the average ensemble at 0.31 and the weighted ensemble at 0.29 are similar. Both models have the same median absolute error of 0.09. The large difference between the maximum error and median absolute error in both ensembles reaffirms the trend of random outlier predictions observed in the GA-optimised LSTM based models. The behavior of the base models has been transferred to ensemble models.

In Table 6, the overall difference in the performance of the ensemble models is negligible and can be attributed to the LSTM based model weight contribution in each ensemble, where the GA-Burnett model only has 10% greater power in the weighted ensemble than in the average ensemble. Thus the difference in performance and predictive capability of the ensembles can only be slight.

2) ENSEMBLE MODELS AND GA-OPTIMISED LSTM MODELS

Table 6 shows the performance of the GA-optimised and ensemble models on the same Baffle 2015 dataset. The $RMSE$ value of the GA-Baffle model decreased by 5.05% with the average ensemble and 6.06% with the weighted ensemble. A larger decrease was observed by the $RMSE$ value of the GA-Burnett model, which was reduced by 7.84% with the average ensemble and 8.82% with the weighted ensemble.

The MSE value of the GA-Baffle model decreased by 12.82% with the average ensemble and 10.25% with the weighted ensemble. The MSE value of the GA-Burnett model was reduced by 19.047% with the average ensemble and 16.66% with the weighted ensemble. The MAE value of the GA-Baffle model decreased by 12.24% with both the average and weighted ensembles. A larger decrease was observed by the MAE value of the GA-Burnett model, which was reduced by 7.19% with both the average and weighted ensembles. The $MAPE$ values are of all four models are relatively similar.

The R^2 score of the GA-Baffle model increased by 1.97% with the average ensemble and by 1.8% with the weighted ensemble. The R^2 score of the GA-Burnett model improved by 3.05% with the average ensemble and by 2.93% with the weighted ensemble. The explained variance score of the GA-Baffle model increased by 1.03% with the average ensemble and by 0.92% with the weighted ensemble. The explained variance score of the GA-Burnett model improved by 2.56% with the average ensemble and by 2.45% with the weighted ensemble.

The difference in the model performance of all four models can be considered notable but not significant. This demonstrates consistency in model performance and prediction capability and is evidence of model robustness and improved model tolerance. Overall, the weighted ensemble performs the best, with the average ensemble performing almost as well. The performance difference between the ensembles is negligible.

The GA-Baffle and GA-Burnett model performance are surprisingly similar, with the GA-Baffle even outperforming the GA-Burnett at points. Previously the Burnett model always performed better. This is possibly due to the similar structure of the Baffle 2015 dataset and the Baffle 2019 dataset. In the Baffle 2015 and 2019 datasets, the data is spread over a single year and exhibits the trend only once. The Baffle 2015 dataset was used to train and optimise the GA-Baffle model, hence the improved performance of the GA-Baffle model on this particular dataset.

TABLE 7. Prediction capability of ensemble and GA-optimised LSTM models on wind power generation data.

Metric	GA-Burnett	GA-Baffle	Average	Weighted
$RMSE$ mg/L	0.258	0.258	0.257	0.256
MSE (mg/L)^2	0.067	0.067	0.066	0.065
MAE mg/L	0.201	0.206	0.202	0.201
R^2	0.117	0.173	0.183	0.257
EV	0.178	0.176	0.183	0.183
$MaxError$ mg/L	0.204	0.147	0.187	0.190

D. ENSEMBLE AND GA-OPTIMISED LSTM MODELS ON UNSEEN MULTIVARIATE DATA

1) WIND POWER

Table 7 shows the consistency in the performance of all four models. The GA-Burnett, GA-Baffle, average, and weighted ensemble models have similar $RMSE$, MSE, and MAE values with negligible differences. All the R^2 and explained variance (EV) scores are low and very far from the numeric value of one, indicating that despite consistently low $RMSE$, MAE, and MSE values, none of the models were able to predict the generation of wind power accurately. The models achieved consistently low maximum errors. The values used to train the models were normalised (by the data provider) positive decimals less than one. Thus in context, these seemingly low values are quite large. The weighted ensemble performs the best by a small margin. In general, both the ensemble models outperform the GA-optimised LSTM models. Consequently, all four models show consistent but poor predictive capability on this dataset.

The structure and data preparation of the wind power dataset could be responsible for this behaviour. The dataset has more than one parameter, but the parameters refer to the same feature, wind power from different wind farms.

In practice, each of these values is treated as a different parameter as they come from different systems (wind farms), while in essence, they are the same feature- wind power values. This data structure differs from the data structure of the water quality datasets used to develop the models. These datasets had several different parameters, each representative of a unique feature of the same system, such as water temperature, dissolved oxygen, to name a few. The models struggle to make predictions on datasets that do not have multiple features from the same system.

The models were developed on data that was normalised using the mean and standard deviation of the dataset. The manner in which the wind power data was normalised by the data provider is not known and might differ from the normalisation carried out on the other datasets, by this study. This possible difference in data preparation leads to the weaker performance of the models on the dataset. In conclusion, the models exhibit great consistency but low tolerance on the wind power generation multivariate dataset.

2) AIR TEMPERATURE

In Table 8, the similar $RMSE$, MSE, and MAE values of each model point towards the consistency in the predictive capability of each model. The R^2 and explained variance (EV) scores are similar, consistently large, and very close to the numeric value of one, highlighting the good predictive capability and capacity to account for data variation of each model on the air temperature dataset. This good performance contrasts with the poor performance of the models on the wind power generation dataset. As with the wind power generation forecast, the weighted ensemble model performs the best, but the performance is comparable to that of the other models with negligible differences.

The dataset contained various related meteorological parameters in addition to air temperature values, such as air density and wind speed, to name a few, all recorded as part of the same system. Thus the dataset contained several parameters representing different features within the same system, which is similar in structure to the water quality datasets used to develop the models and could be a possible reason for the excellent model performance on this particular dataset.

All the air temperature dataset values were normalised before training using the mean and standard deviation of the data. This method of data preparation was applied to datasets used to develop the models. The similarity in the air temperature data structure and data preparation to the water quality datasets used to develop the models is the cause of the consistent and good model predictive capacity.

The air temperature dataset is the largest dataset with has 420 551 samples. The wind power dataset is one of the smaller datasets with only 18 757 samples as shown in Table 4. Thus the models might be better suited to larger datasets with multiple features from the same system. Interestingly, the GA-Burnett model performed the worst on this dataset. The GA-Burnett has the smallest architecture of all the models and thus a lower capacity for a large dataset. In summary, the models exhibit great consistency and high tolerance on the air temperature multivariate dataset.

TABLE 8. Performance metrics of GA-optimised LSTM models and the average and weighted ensemble models for the prediction of air temperature.

Metric	GA-Burnett	GA-Baffle	Average	Weighted
$RMSE$ mg/L	2.545	2.538	2.528	2.520
MSE (mg/L)^2	6.475	6.440	6.400	6.348
MAE mg/L	1.938	1.925	1.920	1.912
R^2	0.900	0.901	0.902	0.905
EV	0.903	0.903	0.904	0.906

TABLE 9. Performance metrics of GA-optimised LSTM models and the average and weighted ensemble models for the prediction of pollution level.

Metric	GA-Burnett	GA-Baffle	Average	Weighted
$RMSE$ mg/L	68.5	69.0	68.0	67.0
MSE (mg/L)^2	5.0	4.7	4.0	3.0
MAE mg/L	46.0	47.0	45.0	44.0
R^2	0.472	0.472	0.474	0.476
EV	0.473	0.473	0.477	0.478

3) POLLUTION LEVEL

Table 9 shows the model performance consistency, which was present with the previous two datasets. The $RMSE$ values of the models were consistently similar to one another. The same trend was witnessed with the MSE and MAE values. The R^2 and explained variance (EV) scores were exceedingly and consistently similar and close to 0.5. Thus the models are moderately capable of making predictions and have an average ability to account for the variation in the pollution dataset. As with the other datasets, the ensemble models perform better than the individual GA-optimised LSTM models, with the weighted ensemble model performing the best, but by an insignificant margin.

The pollution dataset has 43 800 samples (Table 4 and is smaller than the air temperature dataset but larger than the wind power dataset. The models seem to perform moderately or poorly on smaller datasets. In summary, the models exhibit great consistency and moderate tolerance on the pollution dataset.

E. MODELS AND CLASSICAL TIME SERIES FORECASTING METHODS ON UNSEEN UNIVARIATE DATA

1) MODEL PERFORMANCE COMPARISON

Table 10 shows the model performance on the daily minimum temperature dataset in terms of $RMSE$ values. Although the model performance difference is not large, it is more prominent than the notable but insignificant model performance difference observed on the multivariate datasets. The GA-Burnett and GA-Baffle models that performed similarly on the multivariate datasets exhibit a distinct difference in performance on the univariate dataset.

TABLE 10. Performance metrics of classical time series forecasting methods, GA-optimised LSTM models and ensemble models on minimum temperature dataset.

Method/Model	$RMSE$ mg/L
AR	2.389
MA	2.980
ARMA	2.320
ARIMA	2.316
GA-Burnett	2.920
GA-Baffle	4.280
Average Ensemble	3.330
Weighted Ensemble	2.990

The GA-Burnett model with the smallest network architecture performed better than the other models on the univariate dataset with a *RMSE* of 2.92. The GA-Baffle model performed poorly compared to the other models with a *RMSE* of 4.28, as its architecture might have been too large for the univariate dataset. The average ensemble performed worse than the weighted ensemble, which is composed of 60% GA-Burnett and 40% GA-Baffle, with *RMSE* values of 3.33 and 2.99, respectively. Thus the ensemble model that comprised more of the model with the smaller architecture performed better.

The univariate dataset only considers one feature (temperature) for model training and predictions. The models were developed on multivariate datasets with the two features, DO and water temperature, used for model training. Thus the architecture of the models catered for multi-feature datasets. The univariate dataset is the smallest dataset with 3650 samples. This is possibly why the model with the smallest architecture performs the best on this dataset. In general, the models do not perform well on the dataset, further emphasising that the models do not perform well on smaller datasets.

2) MODELS AND CLASSICAL TIME SERIES FORECASTING METHODS

It can be seen from Table 10 that the classical methods perform better than the models on the univariate dataset. The ARIMA method performs the best with the lowest *RMSE* of 2.316, with the ARMA method achieving almost the same value at 2.320. The AR and MA methods achieved *RMSE* values of 2.389 and 2.98, respectively. MA is the worst performing method. The performance of the classical methods is comparable, rendering the performance difference insignificant.

Many of the classical methods outperform the models on the univariate dataset, shown in Table 10. The ARIMA method achieved the lowest *RMSE* value. As mentioned, the GA-Baffle model has the highest *RMSE* value. MA, the poorest performing classical method, has the same performance capabilities as the best performing model, GA-Burnett. Hence classical time series forecasting methods might still be more appropriate for univariate datasets than LSTM and ensemble models. Of all the models, the GA-Burnett model most closely replicated the behaviour of the classical methods, on the univariate datasets, due to the small architecture of the model. Thus the models show a lower tolerance on univariate datasets than on multivariate datasets. The models with smaller architectures have a higher tolerance on univariate datasets than the models with larger architectures.

TABLE 11. Total number of trainable parameters and the computation time for each model.

Model	No. Trainable Parameters	Computation Time
Burnett	8024	23min 41s
GA-Burnett	1344	13min 6s
Baffle	30360	1h 10min 31s
GA-Baffle	1904	19min 29s
Ensemble	3248	25min 41s

F. COMPUTATION TIME AND TRAINABLE PARAMETERS

Table 11 shows the number of trainable parameters for each model, based on the number of units in the two hidden layers and the computation time taken to train each model. Trainable parameters are the number of trainable elements in a network- the parameters that are changed during gradient computation by the optimiser after the application of backpropagation [66]. All the models were developed, trained, and tested on an Aspire A315-53 laptop, with a processor (CPU) of Intel(R) Core(TM) i5-7200U, installed RAM of 4 GB DDR4 (Double Data Rate 4), of which 3,88 GB is usable and storage of 1 TB HDD (hard disk drive) with an effective storage of 930 GB.

As seen in Table 11 after GA-optimisation the trainable parameters for the Burnett model decreased by 6680 parameters and the model training computation time was reduced by 10 minutes. The trainable parameters of the Baffle model were reduced by 28 456 parameters and the computation time by 50 minutes, thus emphasising the impact of the GA-optimisation on the original Baffle model. The ensemble model has 3248 trainable parameters, which is the sum of the trainable parameters of the GA-Burnett and GA-Baffle models. Training the ensemble model takes 25 minutes and 41 seconds. This is less than the sum of the individual training time of the GA-Burnett and GA-Baffle models, which is 32 minutes and 35 seconds. The training of the ensemble model takes much less time than training the Baffle model and is comparable to the Burnett model training time.

The GA-optimisation of the LSTM models had the greatest impact on model robustness and computation time from all the processes in the ensemble development by decreasing the trainable parameters and hence computation time of the original Baffle and Burnett models. This concurs with literature, especially by Krstanovic and Paulheim [27], which suggested that configuring the individual LSTM base models of an ensemble through hyperparameter tuning will increase the base model quality, enhancing the resultant ensemble model quality.

The weight-based combination of the two models did not impact the number of parameters and caused a slight change

TABLE 12. Overall model performance on datasets.

Dataset	Best Model	Best *RMSE* mg/L	Worst Model	Worst *RMSE* mg/L	Difference mg/L
Baffle 2015	Weighted ensemble	0.186	GA-Burnett	0.204	0.018
Wind power	Weighted ensemble	0.256	GA-Burnett GA- Baffle	0.258	0.002
Air Temp	Weighted ensemble	2.520	GA-Burnett	2.545	0.025
Pollution	Weighted ensemble	67.000	GA-Baffle	69.000	2.000
Minimum Temp	GA[a] - Burnett	2.920	GA-Baffle	4.280	1.360

[a] GA-Burnett with *RMSE* of 2.92 marginally outperforms the weighted ensemble with *RMSE* of 2.99

in computation time. However, it did have an impact on model performance and model tolerance. The weighted ensemble model had the best performance on many of the datasets. This is evident from Table 12, which shows the best and worst-performing models and their performance difference on each dataset in terms of *RMSE* values.

In four out of the five datasets, the weighted ensemble model performed better than the other models to differing extents. At times, the difference in model performance was significant and sometimes notable but insignificant. The worst performing models on each dataset were either the GA-Burnett or GA-Baffle model. Thus the weight-based combination of the GA-optimised models improved the performance of the individual GA-optimised models, even if only marginally in certain instances and without substantially increasing computation time, indicating increased model robustness and tolerance.

G. DESCRIPTIVE STATICS AND CORRELATIONS OF THE WATER QUALITY DATASETS

Table 13 shows Spearman's coefficients for the most significant water quality parameters in relation to DO for the three water quality datasets. The datasets show a significant negative correlation between DO and temperature, implying that as water temperature increases, DO concentration decreases, concurring with the research literature. The negative correlation is similar for the Burnett and Baffle 2019 datasets at -0.437 and -0.419, respectively, and is much stronger for the Baffle 2015 dataset at -0.752.

The relationship between pH and DO is unclear. Both the Burnett and Baffle 2015 datasets show a positive correlation. The Baffle 2015 correlation at 0.731 is much stronger than the Burnett at 0.361. In contrast, pH and DO share a weak negative correlation in Baffle 2019. Correlation does not necessarily translate to causation, and as previously stated, there was no documented correlation between DO and pH at the time of this study. The negligible correlation in Baffle 2019 does not support the existence of a causal relationship, and thus this study excluded pH from the predictive DO models.

TABLE 13. Spearman's coefficients for water quality datasets.

Parameters	Burnett Data	Baffle 2019 Data	Baffle 2015 Data
Temperature	-0.437	-0.419	-0.752
pH	0.361	-0.055	0.731
Dissolved oxygen	1.00	1.00	1.00

TABLE 14. Descriptive statistics of datasets used to develop models.

Parameter	Mean	StdDev[a]	Min[b]	25%[c]	50%[d]	75%[e]	Max[f]
Burnett							
Temp°C	24.45	3.58	11.76	21.07	24.76	27.58	32.29
DOmg/L	6.58	0.88	6.00	6.04	6.54	7.04	13.90
pH	7.88	0.53	6.00	7.74	7.84	7.95	8.41
Baffle 2019							
Temp°C	25.10	3.94	16.82	21.24	25.98	28.50	32.63
DOmg/L	6.77	0.96	4.01	6.06	6.72	7.37	9.00
pH	7.37	1.08	4.53	7.65	7.76	7.85	8.21
Baffle 2015							
Temp°C	24.80	3.80	15.72	21.52	25.24	28.17	32.81
DOmg/L	6.79	0.97	3.93	6.09	6.83	7.53	9.18
pH	7.90	0.37	6.23	7.72	8.01	8.14	8.48

[a] Standard Deviation, [b] Minimum, [c] 25th percentile
[d] 50th percentile, [e] 75th percentile, [e] Maximum

The different pH-DO correlations should not be overlooked. The Baffle 2015 dataset has 17 520 observations spread over a single year. Similarly, the Burnett dataset with a total of 17 520 observations per year. The Burnett dataset has this density of data over three years, culminating in 52 560 observations. Perhaps if the pH-DO relationship is observed for over three years, the correlation becomes weaker, and thus the Baffle 2015 dataset has a higher positive pH-DO correlation. The Baffle 2019 is the densest dataset with 52 560 observations over a single year. Densely spaced datasets indicate closer clusters of similar observations with little variation between them. Thus more detailed daily observations could imply no correlation between pH and DO. Hence it is plausible to assume that if the Baffle 2015 had more observations per day, the strong pH-DO correlation might not exist.

In Table 14 which shows the statics of the water quality datasets, the pH minimum and maximum of the Baffle 2019 dataset are 4.53 and 8.21, respectively. The minimum pH is lower than the typically observed 6.5 to 8.5 pH values, indicating highly acidic waters. It is improbable that an external event caused the low pH as the other water quality parameter values (DO and temperature) are unaffected and in range. There is a greater possibility of the incorrect recording of pH for the Baffle 2019 dataset. Despite the discrepancy in pH values, all the datasets have an average pH of around 7, which falls well within the range of typical pH values and indicates neutral river waters.

From Table 14, the mean temperature of the datasets are similar and fall well within the range of typically observed water temperatures shown in Table 1 with 24.5° C, 25.1° C and 24.8° C for the Burnett, Baffle 2019 and Baffle 2015, respectively. The mean temperatures also fall on the upper end of the typical temperature range, indicating that both rivers are in warm areas. Both rivers are in the fairly warm southeast Queensland, Australia. Thus this is a realistic depiction. The maximum temperature for all the datasets is around 32 to 33° C, and the minimum temperatures for the Burnett, Baffle 2019, and 2015 are 11.8° C, 16.8° C, and 15.7° C, respectively, implying that Baffle river is overall warmer than the Burnett river. It is also possible that the Burnett dataset that spans over three years, unlike the Baffle datasets, would have a larger temperature range due to the longer period.

The Burnett DO values fall within the range of the typical observed values from 6 to 14 mg/L. The minimum DO values of the Baffle 2019 and 2015 datasets are 4.0 mg/L and 3.9 mg/L, respectively. They are out of range, possibly due to the higher temperatures observed in the Baffle river, which would lower the DO concentration. The minimum DO values are observed at the minimum temperature values, which are higher for the Baffle datasets. The average DO values for the datasets are around 6.5 to 7 mg/L, falling within the typical range for healthy river waters from 6.5 to 8 mg/L.

VIII. CONCLUSION AND FUTURE WORK

The proposed LSTM based ensemble scheme improved the tolerance (mitigated the discrepancies of the individual LSTM models) of the hybrid GA-optimised LSTM water quality prediction models, for different water quality datasets taken from different sites and different times. Three main contributions and many observations were made by this study to achieve this result.

A. DEVELOPMENT OF THE LSTM FOR WATER QUALITY PREDICTION

Water quality prediction increases the efficiency of water quality monitoring, enabling effective water management, which is necessary for the preservation of rivers. Two LSTM models, the Burnett and Baffle LSTM models were developed from two different time-sequential water quality datasets with differently spaced data structures from two different time periods. Both models can successfully predict water quality ahead of time in terms of dissolved oxygen concentration, using historical dissolved oxygen concentration values and corresponding water temperature values. This is evident from the low $RMSE$, MSE, MAE, and $MAPE$ values, along with the high R^2 and explained variance scores achieved by the models for water quality prediction.

B. PREDICTIVE LSTM MODELS AND DATA STRUCTURE

Overall, the Burnett LSTM model performed better than the Baffle LSTM model due to differences in dataset structure. The Burnett dataset had 52 560 observations spread over three years, and the Baffle dataset 2019 was more densely spaced, with 52 560 observations over a single year. The Burnett dataset allowed the Burnett model to observe a repeated trend over three years with greater variation in data, while the Baffle dataset only exhibited the trend to the Baffle model once with a single year of densely spread data with little variation.

C. GA-OPTIMISATION OF LSTM MODELS

The Burnett and Baffle LSTM models were successfully optimised using GA to increase their efficiency and robustness, resulting in the hybrid GA-optimised LSTM models, GA-Burnett and GA-Baffle LSTM models. Both the GA-optimised models outperformed their original counterparts, showing an enhancement in model performance through hyperparameter tuning. GA-optimisation had the biggest impact on decreasing the overall computation time and the number of trainable parameters. The computation time of the Baffle model was reduced by 50 minutes and the trainable parameters by 28 456 parameters after GA-optimisation, thus significantly reducing the overall computation time of the final ensemble model. Thus base model optimisation was crucial for ensemble model development.

The improvement in performance by the Baffle model after optimisation was much greater than the Burnett model. Unlike the Baffle model, the Burnett model was well-optimised through the trial-and-error method and could not be further improved. Thus when models, are developed from datasets that are spread over long periods and exhibit repeated trends, they are easier to optimise using trial-and-error methods. While models based on more densely spaced datasets with little variation in observations over shorter periods, not exhibiting repeated trends, require a more powerful optimisation technique, such as GA. After the Burnett and Baffle models were GA-optimised, the performance of the models became comparable. Model performance was not similar before optimisation.

D. WEIGHT BASED ENSEMBLE SCHEME

A linear weight-based technique combined the GA-Burnett and the GA-Baffle LSTM models to create two ensemble models, the average and weighted ensemble models. Due to increased robustness and improved predictive capability, the ensemble models performed better than the individual GA-optimised LSTM base models, even if it was only by a slight margin in certain instances and without significantly increasing the computation time. The performance difference between the two ensembles was notable but not significant, with the weighted ensemble only slightly outperforming the average ensemble, mainly due to the similar weight coefficients of the base models.

The similarity of the weight coefficients was due to the similarity in the architecture of the two base models, as the only difference between the base models was the number of LSTM units in the two hidden layers and the time window size. When two similar base models with similar performance capacities are combined to create an ensemble model, both the base models will have an equal or an almost equal contribution. The weighted ensemble was the more powerful of the two ensembles as the GA-Burnett, which performed better than the GA-Baffle had a greater contribution in the weighted ensemble. The similarity of the base models and the resultant similarity of the ensemble models were the cause of the similar behaviour exhibited by all four models: weighted ensemble, average ensemble, GA-Burnett, and GA-Baffle.

E. GENERALISATION AND MODEL TOLERANCE

The behaviour and predictive capability of all the models were similar and consistent. The weighted ensemble model only marginally outperformed the other models on the multivariate datasets. On particular datasets, the models had consistent good predictive capabilities and consistent moderate to poor predictive capacities on other datasets. Thus this study has mitigated the discrepancies and improved the tolerance of the individual LSTM models developed from different datasets, taken from different rivers and periods through the employment of the GA-optimised LSTM based ensemble scheme, to a large extent. Thus further asserting the relevance and tolerance of the developed models, especially the weighted ensemble model, in the wider field of LSTM and ensemble prediction models.

As the models were developed on multivariate datasets, greater consistency in model performance was observed on the multivariate datasets than on the univariate dataset. Classical forecasting methods outperformed the LSTM models on the univariate dataset, illustrating that the classical methods were more appropriate than LSTM models, even LSTM based ensemble models for making predictions on univariate datasets. The smaller LSTM models were better suited to univariate datasets than bigger LSTM models. In general, the models performed better on bigger datasets than smaller datasets. The models performed better on datasets with multiple related features in a single system and when the data was prepared (normalised) in the same way it was for the development of the models. Hence the models tend to perform well on datasets that are similar in structure, preparation, and size to the datasets used to develop them.

F. FUTURE WORK

This work was limited to two LSTM based models. Future studies can identify the optimum number of LSTM based models required to make the most tolerant ensemble model for water quality prediction and possibly in other areas such as energy, finance, geology, and many more. The Burnett and Baffle river datasets were taken from different sites and times, but both rivers are situated in southeast Queensland, Australia, and flow into the Coral Sea of the South Pacific Ocean. More diverse water quality datasets, possibly from different regions, could be used to increase the tolerance of the final ensemble model. The GA-optimised LSTM based models had a different number of LSTM units and time window sizes but were similar in other regards. The use of LSTM based models with greater architectural differences and diversity in terms of the number of hidden layers, input parameters, various activation functions, and optimisers could be explored to enhance the tolerance of the ensemble. Different metaheuristic algorithms, such as particle swarm optimisation, could be used to optimize the LSTM network to gauge the effect of other optimisation algorithms on the tolerance of the ensemble model. The focus of this study was the prediction of DO levels, but water temperature also influences other water quality parameters. Thus the development of a water temperature prediction model along with a DO prediction model should be encouraged. This paper mentioned the correlation between DO and pH. Whether this correlation translates into causation or any other possible link between the two parameters and what their potential pairing could hold for the development of water quality prediction models should be explored.

REFERENCES

[1] H. Razmkhah, A. Abrishamchi, and A. Torkian, "Evaluation of spatial and temporal variation in water quality by pattern recognition techniques: A case study on Jajrood River (Tehran, Iran)," *J. Environ. Manage.*, vol. 91, no. 4, pp. 852–860, Mar. 2010.

[2] U. Ahmed, R. Mumtaz, H. Anwar, A. A. Shah, R. Irfan, and J. Garcáia-Nieto, "Efficient water quality prediction using supervised machine learning," *Water*, vol. 11, pp. 1–14, Dec. 2019.

[3] Y. Khan and C. S. See, "Predicting and analyzing water quality using machine learning: A comprehensive model," in *Proc. IEEE Long Island Syst., Appl. Technol. Conf. (LISAT)*, Farmingdale, NY, USA, Apr. 2016, pp. 1–6.

[4] S. Bouktif, A. Fiaz, A. Ouni, and M. A. Serhani, "Multi-sequence LSTM-RNN deep learning and metaheuristics for electric load forecasting," *Energies*, vol. 13, pp. 391–412, Dec. 2020.

[5] A. P. Engelbrecht, *Computational Intelligence: An Introduction*. Hoboken, NJ, USA: Wiley, 2007.

[6] D. Dheda and L. Cheng, "A multivariate water quality parameter prediction model using recurrent neural network," 2020, *arXiv:2003.11492*. [Online]. Available: https://arxiv.org/abs/2003.11492

[7] C. Zhou, L. Gao, H. Gao, and C. Peng, "Pattern classification and prediction of water quality by neural network with particle swarm optimization," in *Proc. 6th World Congr. Intell. Control Automat.*, Dalian, China, 2006, pp. 2864–2868.

[8] Y. LeCun, Y. Bengio, and G. Hinton, "Review: Deep learning," *Nature*, vol. 521, pp. 436–444, 280 May 2015.

[9] R. Jozefowicz, W. Zaremba, and I. Sutskever, "An empirical exploration of recurrent network architectures," in *Proc. 32nd Int. Conf. Mach. Learn.*, Lille, France, Jul. 2015, pp. 2342–2350.

[10] H. Chung and K. shik Shin, "Genetic algorithm-optimized long short-term memory network for stock market prediction," *Sustainability Open Access J.*, vol. 10, no. 10, pp. 1–18, 2018.

[11] Q. Ye, X. Yang, C. Chen, and J. Wang, "River water quality parameters prediction method based on LSTM-RNN model," in *Proc. Chin. Control Decis. Conf. (CCDC)*, Nanchang, China, 2019, pp. 3024–3028.

[12] Y. Wang, J. Zhou, K. Chen, Y. Wang, and L. Liu, "Water quality prediction method based on LSTM neural network," in *Proc. 12th Int. Conf. Intell. Syst. Knowl. Eng. (ISKE)*, Nanjing, China, 2017, pp. 1–5.

[13] A. Sherstinsky, "Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network," *Phys. D, Nonlinear Phenomena*, vol. 404, pp. 1–43, Feb. 2020.

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

[15] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning* (Adaptive Computation and Machine Learning series). Cambridge, U.K.: Cambridge Univ. Press, 2016.

[16] M. Aslam, J. Lee, H. Kim, S. Lee, and S. Hong, "Deep learning models for long-term solar radiation forecasting considering microgrid installation: A comparative study," *Energies*, vol. 13, no. 147, pp. 1–15, 2020.

[17] S. Bouktif, A. Fiaz, A. Ouni, and M. A. Serhani, "Optimal deep learning LSTM model for electric load forecasting using feature selection and genetic algorithm: Comparison with machine learning approaches," *Energies*, vol. 11, pp. 1636–1656, Oct. 2018.

[18] N. Gorgolis, I. Hatzilygeroudis, Z. Istenes, and L. Gyenne, "Hyperparameter optimization of LSTM network models through genetic algorithm," in *Proc. 10th Int. Conf. Inf., Intell., Syst. Appl. (IISA)*, Patras, Greece, Jul. 2019, pp. 1–4.

[19] Hendri, R. N. Sari, and A. Wibowo, "Timeseries forecasting using long short-term memory optimized by multi heuristics algorithm," *Int. J. Recent Technol. Eng.*, vol. 8, no. 4, pp. 11492–11500, Nov. 2019.

[20] B. Nakisa, M. N. Rastgoo, A. Rakotonirainy, F. Maire, and V. Chandran, "Long short term memory hyperparameter optimization for a neural network based emotion recognition framework," *IEEE Access*, vol. 6, pp. 49325–49337, 2018.

[21] Z.-H. Zhou, *Ensemble Methods: Foundations and Algorithms*. Boca Raton, FL, USA: CRC Press, 2012.

[22] L. Rokach, *Pattern Classification Using Ensemble Methods* (Machine Perception and Artificial Intelligence). Singapore: World Scientific, 2009.

[23] P. Sollich and A. Krogh, "Learning with ensembles: How over-fitting can be useful," in *Proc. Adv. Neural Inf. Process. Syst.*, Nov. 1995, pp. 190–196.

[24] J. Y. Choi and B. Lee, "Combining lstm network ensemble via adaptive weighting for improved time series forecasting," *Math. Problems Eng.*, vol. 2018, pp. 1–8, 5, Aug. 2018.

[25] L. Wolf and A. Shashua, "Feature selection for unsupervised and supervised inference: The emergence of sparsity in a weight-based approach," *J. Mach. Learn. Res.*, vol. 6, pp. 1855–1887, Dec. 2005.

[26] R. Adhikari and R. K. Agrawal, "Combining multiple time series models through a robust weighted mechanism," in *Proc. 1st Int. Conf. Recent Adv. Inf. Technol. (RAIT)*, Dhanbad, India, Mar. 2012, pp. 455–460.

[27] S. Krstanovic and H. Paulheim, "Ensembles of recurrent neural networks for robust time series forecasting," in *Proc. Int. Conf. Innov. Techn. Appl. Artif. Intell.*, Nov. 2017, pp. 1–14.

[28] A. Tikkanen. *Burnett River*. Accessed: Jan. 20, 2020. [Online]. Available: https://www.data.qld.gov.au/dataset/ambient-estuarine-water-quality

[29] Bonzle Digital Atlas of Australia. *Map of Burnett River*. Accessed: Jan. 2, 2020. [Online]. Available: http://www.bonzle.com/c/a?a=p&p=210993&cmd=sp

[30] Department of Environment and Science, Queensland. *Upper Burnett River Drainage Sub-Basin-Facts and Maps*. Accessed: Jan. 4, 2020. [Online]. Available: https://wetlandinfo.des.qld.gov.au/wetlands/factsmaps/

[31] Business and Industry Queensland Government. *Baffle Creek-Watercourse*. Accessed: Jan. 6, 2020. [Online]. Available: https://www.resources.qld.gov.au/qld/environment/land/place-names/

[32] Department of Environment and Science, Queensland. *Baffle Drainage Basin*. Accessed: Feb. 2, 2020. [Online]. Available: https://wetlandinfo.des.qld.gov.au/wetlands/facts-maps/basin-baffle/

[33] Bonzle Digital Atlas of Australia. *Map of Baffle Creek*. Accessed: Jan. 20, 2020. [Online]. Available: http://www.bonzle.com/c/a?a=p&p=210011&cmd=sp

[34] E. Olyaie, H. Z. Abyaneh, and A. D. Mehr, "A comparative analysis among computational intelligence techniques for dissolved oxygen prediction in Delaware River," *Geosci. Frontiers*, vol. 8, pp. 517–527, Oct. 2017.

[35] Fondriest Environmental. *Dissolved Oxygen Fundamentals of Environmental Measurements*. Accessed: Nov. 2, 2020. [Online]. Available: https://www.fondriest.com/environmentalmeasurements/parameters/

[36] D. W. Connell and G. J. Miller, *Chemistry and Ecotoxicology of Pollution*. Hoboken, NJ, USA: Wiley, 1984.

[37] Environment and Natural Resources. *Dissolved oxygen (DO)*. Accessed: Feb. 2, 2020. [Online]. Available: https://www.enr.gov.nt.ca/sites/enr/files/dissolved_oxygen.pdf

[38] Fondriest Environmental. *Water Temperature. Fundamentals of Environmental Measurements*. Accessed: Feb. 2, 2020. [Online]. Available: https://www.fondriest.com/environmentalmeasurements/parameters/

[39] B. Oram. *Water Research Center: The pH of Water*. Accessed: Mar. 13, 2020. [Online]. Available: https://waterresearch.net/index.php/ph

[40] United States Environmental Protection Agency. *Water: Monitoring & assessment: Conductivity*. Accessed: Jan. 13, 2020. [Online]. Available: https://archive.epa.gov/water/archive/web/html/vms59.html

[41] Minnesota Pollution Control Agency. *Turbidity: Description, Impact on Water Quality, Sources, Measures—A General Overview*. Accessed: Jan. 10, 2020. [Online]. Available: https://www.pca.state.mn.us/sites/default/files/wq-iw3-21.pdf

[42] United States Geological Survey. *Water Science School: Turbidity and Water*. Accessed: Mar. 10, 2020. [Online]. Available: https://www.usgs.gov/special-topic/water-science-school/science/

[43] United States Environmental Protection Agency. *Water: Monitoring & assessment: Turbidity*. Accessed: Jan. 13, 2020. [Online]. Available: https://archive.epa.gov/water/archive/web/html/vms55.html

[44] The State of Queensland. *Ambient Estuarine Water Quality Monitoring Data (Includes Near Real-Time Sites)—2012 to Present day*. Accessed: Jan. 2, 2020. [Online]. Available: https://www.data.qld.gov.au/dataset/ambient-estuarine-water

[45] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*. Burlington, MA, USA: Morgan Kaufmann, 2011.

[46] J. Brownlee, *Statistical Methods for Machine Learning*. Cambridge, U.K.: Cambridge Univ. Press, 2015.

[47] C. Lesmeister, *Mastering Machine Learning With R to Deliver Insights for Complex Projects*. London, U.K.: Packt, 2015.

[48] A. Agresti and B. Finlay, *Statistical Methods for the Social Sciences*. Upper Saddle River, NJ, USA: Prentice-Hall, 1997.

[49] J. Cowls and R. Schroeder, "Causation, correlation, and big data in social science research," *P I Policy Internet*, vol. 7, pp. 447–472, Aug. 2015.

[50] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, and I. Goodfellow, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," *Preliminary White Paper*, vol. 9, pp. 1–19, Nov. 2015.

[51] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proc. 14th Int. Conf. Artif. Intell. Statist.*, 2011, pp. 315–323.

[52] S. Ruder, *An Overview of Gradient Descent Optimization Algorithms*. Dublin, Ireland: Aylien, 2017.

[53] F.-A. Fortin, F.-M. De Rainville, M. Gardner, M. Parizeau, and C. Gagnáe, "DEAP: Evolutionary algorithms made easy," *J. Mach. Learn. Res.*, vol. 13, pp. 2171–2175, Jul. 2012.

[54] J. Brownlee, *Better Deep Learning Train Faster, Reduce Overfitting, Make Better Predictions*. Cambridge, U.K.: Cambridge Univ Press, 2018.

[55] H. Pishro-Nik. *Introduction to Probability Statistics and Random Processes*. Accessed: Jan. 2, 2021. [Online]. Available: https://www.probabilitycourse.com/chapter9

[56] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Oct. 2011.

[57] P. Swamidass, "Mean absolute percentage error (MAPE)," in *Proc. Encyclopedia Prod. Manuf. Manage.*, 2000, p. 30.

[58] R. J. Hyndman and A. B. Koehler, "Another look at measures of forecast accuracy," *Int. J. Forecasting*, vol. 22, pp. 679–688, Oct. 2006.

[59] R. G. Pontius, O. Thontteh, and H. Chen, "Components of information for multiple resolution comparison between maps that share a real variable," *Environ. Ecol. Statist.*, vol. 15, pp. 111–142, Dec. 2008.

[60] S. Kotz, N. L. Johnson, and C. B. Read, *Encyclopedia of Statistical Sciences*, vol. 9. Hoboken, NJ, USA: Wiley, 1988.

[61] Kaggle. *Global Energy Forecasting Competition 2012—Wind Forecasting*. Accessed: Oct. 8, 2020. [Online]. Available: https://www.kaggle.com/c/GEF2012-wind-forecasting/data

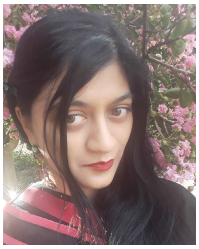
[62] Stytch and Kaggle. *Jena Climate 2009-2016*. Accessed: Dec. 1, 2020. [Online]. Available: https://www.kaggle.com/stytch16/jena-climate-2009-2016

[63] D. Havera and Kaggle. *Beijing Pm25 Data*. Accessed: Dec. 10, 2020. [Online]. Available: https://www.kaggle.com/djhavera/beijing-pm25-data-data-set

[64] M. Siddhartha and Kaggle. *Beijing Multi-Site Air-Quality Data Set*. Accessed: Jun. 5, 2020. [Online]. Available: https://www.kaggle.com/sid321axn/beijing

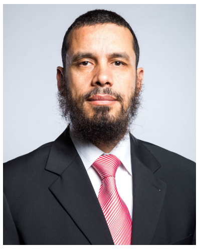
[65] P. Brabban and Kaggle. *Daily Minimum Temperatures in Melbourne*. Accessed: Dec. 5, 2020. [Online]. Available: https://www.kaggle.com/paulbrabban/daily-minimum-temperatures

[66] Tensorflow. *Models and Layers*. Accessed: Nov. 5, 2019. [Online]. Available: https://www.tensorflow.org/js/guide/models_and_layers#model_summary



DHRUTI DHEDA received the B.Sc. (Eng.) and M.Sc. (Eng.) degrees in chemical and metallurgical engineering and the M.Sc. (Eng.) degree*(cum laude)* in electrical and information engineering from the University of the Witwatersrand, Johannesburg, in 2014, 2018, and 2021, respectively, where she is currently pursuing the Ph.D. degree with the School of Electrical and Information Engineering. Her doctoral research explores the use of recurrent neural networks to water and quality monitoring and wastewater analysis. Her research interests include unsupervised learning algorithms and their application to the environmental conservation, water footprinting, and carbon nanotubes.

LING CHENG (Senior Member, IEEE) received the B.Eng. degree *(cum laude)* in electronics and information from the Huazhong University of Science and Technology (HUST), in 1995, the M.Ing. degree *(cum laude)* in electrical and electronics, in 2005, and the D.Ing. degree in electrical and electronics from the University of Johannesburg (UJ), in 2011. In 2010, he joined the University of the Witwatersrand, Johannesburg, where he was promoted to a Full Professor, in 2019. He works as an associate editor of three journals. He has published more than 100 research papers in journals and conference proceedings. He has been a visiting professor at five universities and the principal advisor for over 40 full research master students. His research interests include telecommunications and artificial intelligence. He was awarded the Chancellor's Medals, in 2005 and 2019, and the National Research Foundation ratings, in 2014 and 2020. The IEEE ISPLC 2015 Best Student Paper Award was made to his Ph.D. student in Austin. He is the Vice-Chair of IEEE South African Information Theory Chapter.



ADNAN M. ABU-MAHFOUZ (Senior Member, IEEE) received the M.Eng. and Ph.D. degrees in computer engineering from the University of Pretoria. He is currently the Centre Manager of the Emerging Digital Technologies for 4IR (EDT4IR) Research Centre, Council for Scientific and Industrial Research (CSIR), an Extraordinary Professor with the University of Pretoria, a Professor Extraordinaire with the Tshwane University of Technology, and a Visiting Professor with the University of Johannesburg. His research interests include wireless sensor and actuator networks, low power wide area networks, software defined wireless sensor networks, cognitive radio, network security, network management, and sensor/actuator node development. He is a member of many IEEE Technical Communities. He is an Associate Editor of IEEE ACCESS, IEEE INTERNET OF THINGS, and IEEE TRANSACTION ON INDUSTRIAL INFORMATICS. He has participated in the formulation of many large and multidisciplinary research and development successful proposals (as a Principal Investigator or a main author/contributor). He is the founder of the smart networks collaboration initiative that aims to develop efficient and secure networks for the future smart systems, such as smart cities, smart grid, and smart water grid.