IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 24, 2022, accepted June 1, 2022, date of publication June 6, 2022, date of current version June 10, 2022.

*Digital Object Identifier 10.1109/ACCESS.2022.3180482*

# Water Quality Prediction for Smart Aquaculture Using Hybrid Deep Learning Models

**K. P. RASHEED ABDUL HAQ, (Student Member, IEEE), AND V. P. HARIGOVINDAN, (Senior Member, IEEE)**
Department of Electronics and Communication Engineering, National Institute of Technology Puducherry, Karaikal 609609, India

Corresponding author: K. P. Rasheed Abdul Haq (rasheedabdulhaq@gmail.com)

This work was supported by the Project titled "Design and Development of IoT-Based Low Cost Water Quality Monitoring and Reporting System for Aquaculture" by the Department of Science & Technology (DST) (Science for Equity Empowerment and Development (SEED) Division), Ministry of Science and Technology, Government of India.

**ABSTRACT** Water quality prediction (WQP) plays an essential role in water quality management for aquaculture to make aquaculture production profitable and sustainable. In this work, we propose hybrid deep learning (DL) models, convolutional neural network (CNN) with the long short-term memory (LSTM) and gated recurrent unit (GRU) for aquaculture WQP. CNN can effectively fetch the aquaculture water quality characteristics, whereas GRU and LSTM can learn long-term dependencies in the time series data. We conduct experiments using the two different water quality datasets and present an extensive study on the impact of hyperparameters on the performance of the proposed hybrid DL models. Furthermore, the performance of hybrid CNN-LSTM and CNN-GRU models are compared with different baseline LSTM, GRU and CNN DL models and also with attention-based LSTM and attention-based GRU DL models. The results show that the hybrid CNN-LSTM outperformed all other models in terms of prediction accuracy and computation time.

**INDEX TERMS** Aquaculture, CNN, deep learning, GRU, hybrid models, LSTM, water quality prediction.

## I. INTRODUCTION

Aquaculture is a vital component for ensuring global food security, as the world's population is predicted to exceed 9.8 billion by 2050 [1]. Fishing from the sea is currently far beyond its limits and will not be in a position to meet the growing demand for food. Aquaculture has progressed steadily to meet the demand for fish as food while protecting marine life from overfishing by maintaining a consistent fish supply [2]. Nowadays, with the application of IoT, artificial intelligence, data analysis, etc., aquaculture systems are upgraded to smart aquaculture systems for improving performance and efficiency [3], [4].

To maintain and manage aquaculture, water quality monitoring and water quality prediction (WQP) are essential [5]. In aquaculture, water quality is the most influential and critical factor that affects production as well as product quality. The vital water quality parameters in aquaculture are salinity, pH, dissolved oxygen (DO), and temperature [6]. Water quality is affected by many factors, such as fish density, quality of the feed, feeding interval, climate, and more. The change in water quality will upset the balance of the system with algae bloom, bacterial growth, etc. [7]. This can lead to severe problems, such as trigger stress, lack of food intake, vulnerability to diseases, and increased mortality rate of fish [8]. Hence, if we forecast the trend in water quality changes, we can employ safeguards in advance to avoid imbalances in the ecosystem and also ensure suitable conditions for optimum growth of the fish. Therefore, accurate WQP can drastically improve productivity to make aquaculture more profitable and sustainable.

Aquaculture water quality is affected by meteorological conditions and complex interdependency relations between different water quality parameters. Hence the change of water quality parameters exhibit non-linear characteristics, which results in low prediction accuracy. The time-series data shows periodic variations depending on the seasons and climatic conditions. Time series data is used for analysis and prediction in various fields such as the stock market, medical field, energy consumption, weather forecasting,

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

solar radiation etc. [9]–[14]. The major approaches for WQP include classical prediction mechanisms such as the autoregressive moving average model (ARMA), autoregressive integrated moving average (ARIMA), seasonal autoregressive integrated moving average model (SARIMA), seasonal auto-regressive integrated moving average with exogenous factors (SARIMAX), Holt-Winters exponential smoothing (HWES), Markov model, Grey model, and support vector regression (SVR) etc. [15]. However, these models are least suitable for aquaculture WQP because they assume a linear relationship with water quality parameter data. These models are less accurate and take a long prediction time, making them unsuitable for predicting the non-linear aquaculture water quality parameters [16], [17].

The deep learning (DL) models like long short-term memory (LSTM), gated recurrent unit (GRU), and convolutional neural network (CNN) have the flexibility in capturing the non-linear nature of the aquaculture WQP [18]. These DL models have overcome the limitations of Recurrent neural networks (RNNs) and achieved great success in various applications. Recent advancements in artificial intelligence have made DL one of the most proficient methods, and DL is popularly applied in different fields such as image processing, speech recognition, text prediction, etc. [19]–[21]. DL makes time series prediction more precise and efficient in terms of training time and required processing power. In addition, DL models consider the non-linear characteristics of water quality changes. LSTM is the widely used DL method for WQP due to its remarkable performance in time series prediction [22].

LSTM and GRU models are not very capable of maintaining long-term memory for time series prediction, especially for long sequences [23]–[25]. In time series forecasting, the prediction value at the current time step is impacted mainly by past observations. In some cases, the ones with a significant influence might have appeared at the time step long back from the current one. Hence the neural networks must have the prolonged memory holding capacity to hold memory for the learned long-term dependencies. The ability of LSTM models to capture long-term dependencies information from historical observations is still considered a critical performance bottleneck, as indicated in recent studies [26], [27]. The work [23] has theoretically proven that standard LSTM does not have long memory from a statistical perspective. Hence, building a time-series prediction model capable of capturing and remembering complex dependencies is a crucial problem that needs to be addressed.

In [25], [28], the authors have discussed how incorporating an attention module can improve the prediction accuracy of LSTM. Different works have combined the attention model to LSTM to capture the relevant data from long time series data, helping to improve its prediction performance. LSTM network with an attention mechanism has an adaptive decay rate of long-term memory. This decay rate is much lower than the polynomial or exponential decay rate. In the work [29], the reading of the geo-sensor is predicted for a few hours by using multi-level attention-based RNN by feeding data from multiple sensors and meteorological data along with spatial information of these sensors. This work uses the attention mechanism to model the dynamic spatio-temporal relations within the sensors. A fusion module is designed to input the effects of external factors from different domains. The authors have tested their model on a water quality and air quality dataset.

The hybrid CNN-LSTM models also work similarly, helping the LSTM model capture the relevant data. In hybrid models, CNN with maxpooling is used to reduce the length of the input sequence, which is fed into the LSTM and GRU models. The hybrid models are not affected by exponential decay, as the CNN with maxpooling will input the key features and helps LSTM in forecasting with better accuracy. Thus the hybrid models can achieve higher accuracy in prediction. Hybrid models aim to build neither overfit nor underfit models with strong generalization ability and require less training time for fitting the models. The convolutional layer with maxpooling enhances the capacity of the LSTM and GRU models to learn and store the critical features in the non-linear water quality parameters. Thus convolution layer with maxpooling reduces exponential decay in LSTM and GRU, bypassing selective information to the LSTM/GRU layer using convolution and maxpooling [30], [31]. CNNs are faster by design since the computations in CNNs can happen in parallel, while RNNs need to be processed sequentially since the subsequent steps depend on previous ones. In [32], authors introduced the Quasi-Recurrent Neural Networks that use some of the CNN components to imitate RNNs while speeding them up. CNN leads to various complexity reductions by concentrating on the key features. The use of convolution layers leads to a reduction in the size of tensors and with Maxpooling leads to a further decrease in training time. Reducing the training time of the hybrid models.

CNN and its hybrid variations are used in many research work for time series prediction. DL models have been extensively used in many fields of time series prediction, including finance (stock price, cryptocurrency, and precious metal prediction) [33], [34]. The authors [35] have used the CNN-LSTM model for gold price prediction. They have proposed two simple hybrid models and compared the results with other DL models. The authors also compared LSTM models with the hybrid models and observed that hybrid models perform better. In [36] to predict blood glucose (BG) levels, authors have used multiple layers of CNN along with LSTM. This hybrid model has shown superior performance in BG prediction. In another work, the authors have used Bi-LSTM with CNN for air quality prediction [37]. They have done an extensive study with two datasets and shown results proving the model's capability for accurate air quality prediction. There have been multiple works based on hybrid DL models for the prediction of air quality ($PM_{2.5}$) based on CNN and LSTM models [38], [39].

### A. MAIN CONTRIBUTIONS OF THE WORK

In this work, we propose to combine RNN and CNN, making a new hybrid model for WQP having the advantages of both models. RNN has a vanishing gradient problem while training with large data and affects the learning from large datasets [40]. This problem is solved by introducing two specialised variants, LSTM and GRU. LSTM and GRU can learn long term dependencies from training data, while CNN is good with feature extraction [41]. We propose hybrid DL models that combine the advantages of CNN with LSTM and GRU for aquaculture WQP. The significant contributions of this research work are as follows:

1) Three years of data (January 2016 to December 2018) is collected from aquaculture ponds located in Kerala under ADAK. The data is pre-processed to remove abnormalities in data affecting the prediction accuracy. The linear interpolation (LIN) and smoothing methods are used to fill the missing data and correct abnormal data, respectively.
2) We have proposed hybrid DL models, CNN-LSTM and CNN-GRU, that combine the advantages of CNN with both LSTM as well as GRU for aquaculture WQP. The proposed hybrid models are trained and tested with the data collected from ADAK.
3) We have also trained and tested the proposed CNN-LSTM and CNN-GRU models with another water quality parameters dataset provided in [16]. This data is collected from the marine aquaculture base in Xincun Town, LingShui County, Hainan Province, China.
4) The performance of hybrid CNN-LSTM and CNN-GRU models are compared with baseline DL models LSTM, GRU, and CNN and with attention-based LSTM and attention-based GRU DL models. Results show that the hybrid models significantly improve accuracy compared to the baseline models for aquaculture WQP. Also, the hybrid models outperform all the DL models in terms of computation time.

To the best of the authors' knowledge, this is the first research work to propose and analyse the performance of hybrid CNN–LSTM and CNN-GRU models to predict water quality parameters for aquaculture.

### B. PAPER ORGANIZATION

The reminder of the paper is arranged as follows: In Section II, the materials and methods are described, in which we have explained in detail the data that we have collected. Section III explains the proposed hybrid CNN-LSTM and CNN-GRU models. Section IV explains in detail the experiments that we have done to analyse and compare the performance of proposed hybrid DL models with the baseline DL models for different hyperparameters ($h_p$) variations. Section V further analyse the performance of all the DL models for a selected set of $h_p$ and give the final results comparing the models. Finally, section VI summarise the results and concludes the work.

**TABLE 1. Average water quality parameters monitored in 2016, 2017 and 2018.**

| Water quality parameter | Year | Mean | Min | Max |
|---|---|---|---|---|
| Salinity (ppt) | 2016 | 14.3791 | 13.186 | 15.736 |
| pH | | 7.3338 | 7.192 | 7.458 |
| DO (ml/L) | | 5.3581 | 5.237 | 5.512 |
| Temperature (°C) | | 24.6277 | 23.299 | 25.685 |
| Salinity (ppt) | 2017 | 15.4084 | 14.261 | 17.395 |
| pH | | 7.3326 | 7.192 | 7.46 |
| DO (ml/L) | | 5.2995 | 5.058 | 5.499 |
| Temperature (°C) | | 24.9244 | 22.999 | 27.278 |
| Salinity (ppt) | 2018 | 15.2589 | 14.307 | 16.818 |
| pH | | 7.3411 | 7.24 | 7.483 |
| DO (ml/L) | | 5.2856 | 5.116 | 5.391 |
| Temperature (°C) | | 25.1092 | 24.294 | 26.681 |

## II. MATERIALS AND METHODS

### A. ACQUISITION OF DATA

The data of aquaculture water quality parameters used in this work is collected from an aquaculture farm under ADAK at Kollam, Kerala, India. We have collected water quality parameter data for three years, from January 2016 to December 2018. Water quality parameters are collected from the pond on a daily basis. The trends in variations of water quality parameters of the aquaculture pond are studied. Fig. 1 shows the annual trends of salinity, pH, DO, and temperature. From the plots, it is clear that all water quality parameters are non-stationary in nature.

Table 1 shows the average of the water quality parameter data of each year. The salinity in 2016 was between 13.18 ppt and 15.73 ppt; in 2017, it was between 14.26 ppt and 17.39 ppt, and in 2018 it was between 14.30 ppt and 16.82 ppt. The pH in 2016 was between 7.20 and 7.45, and in 2017, it was between 7.19 and 7.46. Moreover, in 2018, it was between 7.24 and 7.48. The DO in 2016 was between 5.23 ml/L and 5.51 ml/L; in 2017, it was between 5.05 ml/L and 5.49 ml/L, and in 2018 it was between 5.11 ml/L and 5.39 ml/L. The temperature value in 2016 was between 23.30 °C and 25.69 °C; in 2017 it was between 22.99 °C and 27.27 °C, and in 2018 it was between 24.29 °C and 26.68 °C. The mean DO minimum in 2018, where the mean temperature is also highest, indicates the relation between DO and temperature. When temperature increases, DO will decrease. In all three years, the average salinity value is coming down in June - July, coinciding with the monsoon season in Kerala.

### B. CORRELATION ANALYSIS

Correlation analysis is performed on data from January 2016 and December 2018 to analyse the relationship between these water quality parameters (salinity, pH, DO and temperature). Pearson's correlation coefficient shows the correlation between the variables. A correlation ratio of more

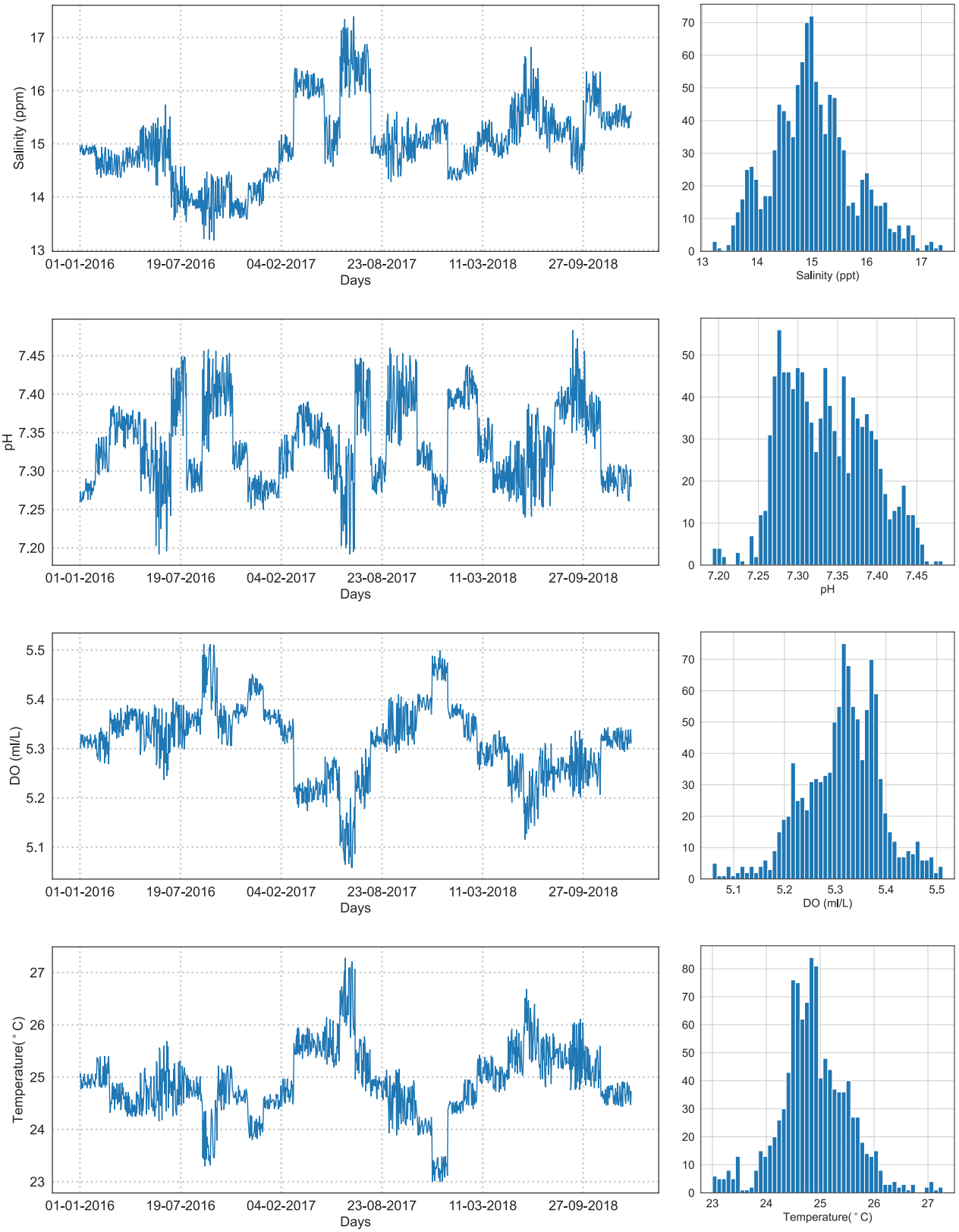



**FIGURE 1.** Water quality parameters variation trend of ADAK water quality dataset.

**TABLE 2. Correlation coefficient matrix of ADAK water quality parameters dataset.**

| | Salinity | pH | DO | Temperature |
|---|---|---|---|---|
| Salinity | 1.0000 | -0.0880 | -0.7203 | 0.5278 |
| pH | -0.0880 | 1.0000 | 0.0083 | 0.0185 |
| DO | -0.7203 | 0.0083 | 1.0000 | -0.9692 |
| Temperature | 0.5278 | 0.0185 | -0.9692 | 1.0000 |

than 0.5 indicates a direct correlation with the parameter and a significant interdependence between the parameters. A value between 0.5 to 0.2 demonstrates a correlation with interdependence between them to some extend. A value of less than 0.1 suggests a poor correlation with the parameter and an insignificant interdependence between the parameters. The data is standardised to eliminate dimensional influence, and then the Pearson correlation coefficient matrix is applied. Table 2 shows the correlation coefficient matrix. The correlation between salinity, pH, DO, and temperature is given in Table 2. The findings in Table 2 show that temperature and salinity have a significant influence on the DO. Moreover, pH does not have any impact on the other water quality parameters monitored here.

Temperature and DO show the most correlation. Whereas temperature and pH show the minimum correlation. The pH has an insignificant correlation with salinity, DO and temperature. DO has a higher correlation with temperature, followed by salinity and an insignificant correlation with pH. Similarly, the temperature has a high correlation with DO and a moderate correlation with salinity. However, the correlation of temperature with pH is insignificant.

### C. DATA PRE-PROCESSING

Aquaculture water quality parameters data measured from the ponds may have anomalies such as missing data or abnormal data. This can be due to problems with sensors or possible errors while storing the data. These variations will lead to excessive deviation of predicted values from actual monitored values. In order to improve the accuracy of prediction, we must provide predictive models with clean, reliable and succinct data. Unavailable data can no longer be retrieved, and missing data can only be computed as accurately as possible. Throughout this work, the missing part of the water quality parameters data is first filled using LIN algorithm [42], [43], and then this data is used to forecast the water quality parameters. As illustrated in Fig. 1, the water quality parameters show continuous seasonal variation and also exhibit correlation with time.

Any data which is taken in a definite time interval is time-series data. The data set of water quality parameters obtained from a source is also time series data. It is an organized set consisting of measured values of water quality parameters at regular intervals. Water quality parameters are measured at a specified time every day.

$$TS_{i,n} = ((y_{i,1}, T_1), \ldots, (y_{i,n}, T_n)) \tag{1}$$

In (1) $TS_{i,n}$ is defined as $n$-length time-series with $i$th water quality parameter. The sampling interval for all parameters is one day i.e., all parameters are measured at the same time on a daily basis. If the value $y_{i,v}$ at $T_v$ is missing, then we can obtain its approximated value using the LIN algorithm. The LIN function can be constructed as:

$$Y_{i,v} = y_{i,u} + \frac{y_{i,u} - y_{i,w}}{T_u - T_w}(T_v - T_u) \tag{2}$$

When a water quality data is missing at any point, the LIN algorithm first finds the two closest moments represented as $T_u$ and $T_w$. Then calculate the missing value at that moment $T_v$ by utilizing the parameter values $y_{i,u}$ and $y_{i,w}$ at the moments $T_u$ and $T_w$ respectively based on (2). Where $Y_{i,v}$ is the estimated value of missing value $y_{i,v}$.

### D. DESCRIPTIVE STATISTICS

Descriptive statistics of the water quality parameter data used in this work, including minimum, mean, maximum, median, standard deviation, skewness and kurtosis, can be found in Table 3. Standard deviation is highest for salinity (0.7400 ppt) and temperature (0.6522 °C), while DO (0.0759 ml/L) and pH (0.0540) have the lowest standard deviation. The DO concentration ranged between 5.06 and 5.51 ml/L, with a mean of 5.31 ml/L. The minimum and maximum values of temperature are 22.99 and 27.27 °C respectively (with a mean of 24.89 °C). The skewness and kurtosis for the salinity, pH, DO, and temperature are provided in Table 3. The skewness and kurtosis were within $-2$ to $+2$, indicating that these water quality parameters follow a normal (Gaussian) distribution.

## III. PROPOSED HYBRID CNN-LSTM/GRU MODEL

### A. LSTM DL NEURAL NETWORK

LSTM DL neural network was proposed in 1997 by authors of [44] to avoid long-term dependency problems in RNN. In LSTM, a control unit is introduced to store information, unlike a hidden layer in RNN. This hidden state is divided to memory cells $c_t$ and working memory $h_t$. The $c_t$ is responsible for sequence features retention, and previous sequence memory is controlled by the forgetting gate $f$. The portion of the current memory $c_t$ is controlled by output gate $o_t$ and $h_t$ is used as the output. The current state $h_{t-1}$ and current input $x_t$ written to memory cells are responsibility of the input gate $i$. The LSTM architecture [45] is shown in Fig. 2, and memory units are defined as follows in 3:

$$\begin{aligned} f_t &= \sigma(W_f \times [h_{t-1}, x_t] + b_f) \\ i_t &= \sigma(W_i \times [h_{t-1}, x_t] + b_i) \\ \tilde{c}_t &= tanh(W_c \times [h_{t-1}, x_t] + b_c) \\ c_t &= f_t \times c_{t-1} + i_t \times \tilde{c}_t \\ o_t &= \sigma(W_o \times [h_{t-1}, x_t] + b_o) \\ h_t &= o_t \times tanh(c_t) \end{aligned} \tag{3}$$

In this architecture $f_t$, $i_t$, and $o_t$ are forget, input and output gate layer respectively. $\tilde{c}_t$ and $c_t$ are new and final memory

**TABLE 3.** Statistical summary of the ADAK water quality parameters dataset.

| Water quality parameter | Min | Mean | Max | Median | SD | Skewness | Kurtosis |
|---|---|---|---|---|---|---|---|
| Salinity (ppt) | 13.1860 | 15.0149 | 17.3950 | 14.9640 | 0.7400 | 0.3394 | 0.0390 |
| pH | 7.1920 | 7.3359 | 7.4830 | 7.3325 | 0.0540 | 0.2119 | -0.6277 |
| Dissolved oxygen (ml/L) | 5.0580 | 5.3144 | 5.5120 | 5.3210 | 0.0759 | -0.3610 | 0.4804 |
| Temperature(°C)' | 22.9990 | 24.8869 | 27.2780 | 24.8420 | 0.6522 | 0.1409 | 1.0892 |

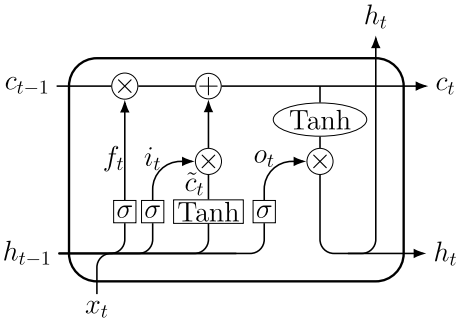



**FIGURE 2.** LSTM cell structure.

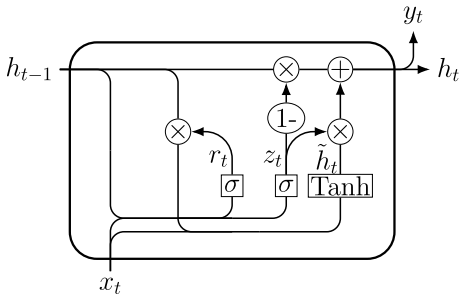



**FIGURE 3.** GRU cell structure.

cell, $w$ is weight matrices, $b$ is bias vectors, $\sigma$ is the sigmoid activation function.

### B. GRU DL NEURAL NETWORK

GRU DL neural network was proposed by authors of [46] in 2014. It similar to LSTM, but required less computing power. GRU is an improved version of RNN with only two gates, an update gate and a reset gate. There are no additional memory cells to store information; GRU can control information inside the unit. The update gate decides whether to pass the previous output $h_{t-1}$ to the next cell. The reset gate reads the input sequences when the gate is set to zero and forgets the previously calculated state. As a result, GRU has fewer tensor operations than LSTM and runs typically faster than LSTM. The GRU architecture [45] is shown in Fig. 3 and memory units are defined as follows:

$$\begin{aligned} z_t &= \sigma(W_z \times [h_{t-1}, x_t]) \\ r_t &= \sigma(W_r \times [h_{t-1}, x_t]) \\ \tilde{h}_t &= tanh(w \times [r_t \times h_{t-1}, x_t]) \\ h_t &= (1 - z_t) \times h_{t-1} + z_t \times \tilde{h}_t \end{aligned} \tag{4}$$

In this architecture $z_t$ is the reset and $r_t$ is the update gate, $\tilde{h}_t$ is process input, and $h_t$ is hidden state update, $w$ is weight matrices and $\sigma$ is the sigmoid activation function.

### C. HYBRID CNN LSTM/GRU DL NEURAL NETWORK

The Hybrid CNN-LSTM and CNN-GRU DL neural network structures are shown in Fig. 4 and Fig. 5. The two models are the hybrid of CNN with LSTM and CNN with GRU. The first part of the model is CNN, to which the data is fed, and it extracts the features. There is a dropout layer after the convolution layer (Conv1D) and pooling layer (MaxPooling1D). The second part has the LSTM or GRU, followed by a dense layer to give the output.

In the CNN-LSTM model, we use Conv1D with 32 filters, a kernel size of 3 and ReLU is used as the activation function. A pooling layer MaxPooling1D follows this with a pooling size of 2. The CNN layer extracts all the features and then feeds them to the LSTM layer. The output of the CNN layer is fed to the LSTM layer after pooling through the flatten layer. Then, the LSTM layer outputs its output to the dense layer through the flatten layer, and finally, the prediction is output at the dense layer.

The CNN-GRU models use Conv1D with 64 filters, and a kernel size of 5 and ReLU is used as the activation function. A pooling layer MaxPooling1D follows this with a pooling size of 4. The CNN layer extracts all the features and then feeds them to the GRU layer. The output of the CNN layer is fed to the LSTM layer after pooling and 0.2 dropout. After that, the LSTM layer outputs its output to the dense layer, and finally, the prediction is output at the dense layer.

The structure of both CNN-LSTM and CNN-GRU is further detailed in Table 4. These models use a learning rate of 0.0008 and Adam for optimisation [47]. Here, MSE is used as the loss function. The algorithm is based on CNN, LSTM, and GRU DL neural network implemented using Python, Keras and Tensorflow.

### D. MODEL IMPLEMENTATION

The water quality dataset is used to train the DL models to predict each water quality parameter. First, the dataset is divided into two; 80% for the training and 20% for the testing. The training dataset is used to develop the models, while the testing dataset is used to validate and compare the performance of models developed. All input and output variables were scaled between 0 and 1 via normalization

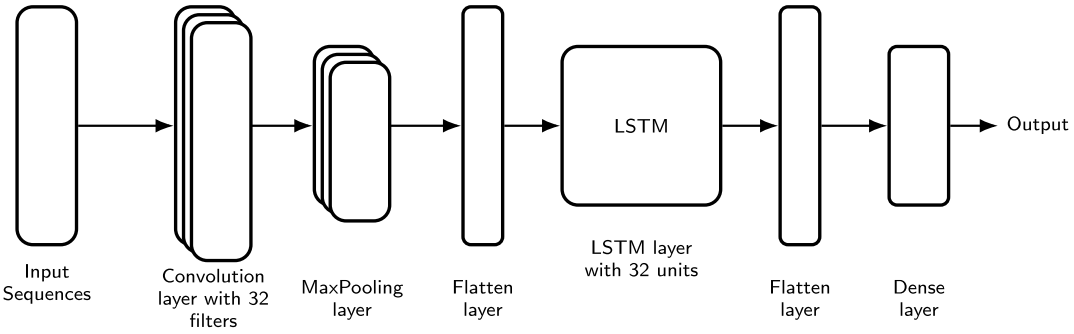



**FIGURE 4.** Proposed Hybrid CNN-LSTM DL neural network model structure.

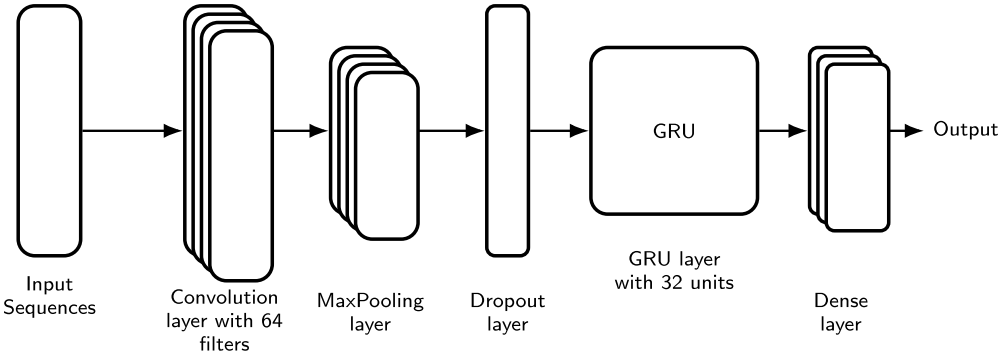



**FIGURE 5.** Proposed Hybrid CNN-GRU DL neural network model structure.

**TABLE 4.** The structures of proposed hybrid CNN-LSTM and CNN-GRU DL models.

| Model | Structure |
|---|---|
| CNN-LSTM | Conv1D layer (32 filters + 3 filter sizes, ReLU activation)<br>Maxpooling (2 pooling size)<br>Flatten ()<br>LSTM layer (32 units, ReLU activation)<br>Flatten ()<br>Dense layer (1 neuron)<br>Compile (MSE loss, Adam optimizer, lr = 0.0008) |
| CNN-GRU | Conv1D layer (64 filters + 5 filter sizes, ReLU activation)<br>Maxpooling (4 pooling size)<br>Dropout layer (0.2)<br>GRU layer (32 units, ReLU activation)<br>Dense layer (30 neuron, ReLU activation)<br>Dense layer (10 neuron, ReLU activation)<br>Dense layer (1 neuron)<br>Compile (MSE loss, Adam optimizer, lr = 0.0008) |

and minimum-maximum scaling techniques, using the MinMaxScaler in the scikit-learn preprocessing library using Python [48]. The training and testing datasets are transformed into supervised learning frames with various array manipulation techniques. The time series sequences are converted to input-output equations using the sliding window procedure. This transformed data is used to train the models for WQP.

## IV. EXPERIMENTAL RESULTS AND DISCUSSIONS

This research is to analyse the performance of hybrid models CNN-LSTM and CNN-GRU. We use two water quality data sets to conduct experiments, analyse and evaluate the performance of the proposed models. Furthermore, by comparing proposed models with other DL models, the prediction performance and effectiveness of the proposed models are validated.

### A. DATASETS

#### 1) ADAK WATER QUALITY DATASET

The first dataset used in this research work is the aquaculture water quality data that we have collected from aquaculture farms under ADAK, Kerala, which includes the data of water quality parameters such as salinity, pH, DO and temperature. The preprocessing and cleaning of this data was done as part of this work. We have three years of data with 1096 samples. The data is collected on a daily basis at the same time. This is a medium-size dataset compared to the MAC dataset used in this research work.

#### 2) MAC WATER QUALITY DATASET

The second dataset is taken from another research work which is made available publically by the authors [16]. This data is collected from the marine aquaculture base in Xincun Town, LingShui County, Hainan Province, China. It also includes water quality data parameters such as salinity, pH, DO, and temperature and the data is collected every 5 minutes. This data have a total of 23200 samples collected over 80 days. This is a large dataset compared to the ADAK dataset.

### B. EXPERIMENTAL SETUP

The experimental environment is Microsoft Azure Virtual Machines with specifications: Inter(R) Xeon (R) 8272CL CPU @2.60GHz, 32 GB RAM, Windows 10 (64-bit) operating system, Visual studio code IDE, and we have implemented the neural network model using Python 3.9.6, Keras 2.6.0 and Tensorflow 2.6.0.

The hybrid models are compared with CNN, LSTM, and GRU DL models. We use 80% of data to train the model, and the remaining 20% is utilised to test the prediction accuracy of results. The most critical part of building a DL model is tuning and optimisation of the $h_p$. For the model to be most effective and efficient, we need to tune as many $h_p$. In this work, we have selected four $h_p$ for tuning: learning rate, epochs, window size and batch size. In each of the models, we have kept other $h_p$, the number of layers and neurons in each layer fixed. A dropout of 0.2 is applied after the primary layers to avoid over-fitting, and *ReLU* is applied as the activation function. In addition, the optimiser adopted in all the DL models in this work is Adam.

### C. MODEL PERFORMANCE METRICS

The performance of the prediction models are evaluated using mean absolute error (MAE), mean square error (MSE), root mean squared error (RMSE) and mean absolute percentage error (MAPE), computed by the set of equations given below:

$$\begin{aligned} MAE &= \frac{1}{n}\Sigma_{i=1}^{n}|\,(A_i - Y_i)\,| \\ MSE &= \frac{1}{n}\Sigma_{i=1}^{n}(A_i - Y_i)^2 \\ RMSE &= \sqrt{\frac{1}{n}\Sigma_{i=1}^{n}(A_i - Y_i)^2} \\ MAPE &= \frac{1}{n}\Sigma_{i=1}^{n}\frac{|(A_i - Y_i)|}{A_i} \end{aligned} \tag{5}$$

where $A_i$ is the actual value of $i_{th}$ sample, $Y_i$ is the predicted value of $i_{th}$ sample and $n$ is the number of samples.

### D. EXPERIMENTS

This work aims to improve the accuracy of aquaculture WQP and reduce the computation time. We have done five sets of experiments. Four sets are done by varying each of the $h_p$, and the fifth experiment is done by changing the step size. We have plotted learning rate, epoch, window size, and batch size versus RMSE. We have plotted each hyperparameter versus the computation time as well. The experiments are done on both datasets (ADAK dataset and MAC dataset). This will help to analyse the performance of these DL models on a medium-sized dataset and large dataset. The ADAK water quality dataset, even though it has three years of data, it has only 1096 data points, as the data is recorded only once a day. In comparison, the MAC water quality dataset has 23200 data points, which is from 80 days since data is recorded every 5 minutes.

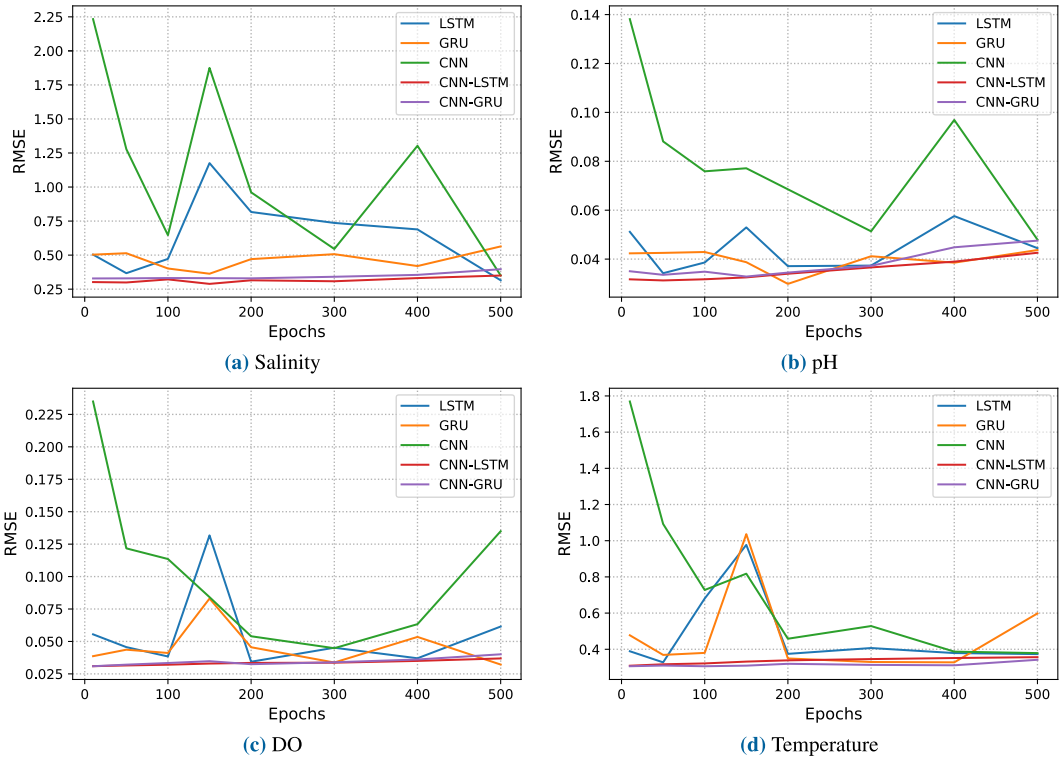
#### 1) EPOCH

The outcome of choosing different epochs {10, 50, 100, 150, 200, 300, 400, 500} for each DL model is studied. Fig. 6 shows the RMSE vs epoch for different water quality parameters using ADAK water quality dataset for the proposed hybrid DL models and LSTM, GRU as well as CNN DL models. Here the proposed hybrid DL models CNN-LSTM and CNN-GRU maintain better performance than baseline DL models LSTM, GRU and CNN. Hybrid models achieve good performance within 50 to 100 epochs, and the generalization is attained within 100 epochs. Prediction accuracy performance is maintained for all four water quality parameters for epochs 10 to 500. Also, the prediction performance of our hybrid DL models is higher compared to the baseline DL models. As well as baseline DL models are taking at least 400 epochs to achieve generalization and is under-fitting training data for epochs less than 400. The baseline DL models are close to the performance of the proposed models at 500 epochs. The baseline models require more computation resources to achieve similar results. The hybrid models start to overfit a little after 150 epochs for the ADAK water quality dataset. Still, the accuracy is maintained, indicating the adaptability of the proposed DL model to different datasets.

To further evaluate the performance, we have repeated the experiment with the MAC water quality dataset. Fig. 7 shows the RMSE vs epoch for all four water quality parameters on the MAC dataset of the proposed hybrid DL models, LSTM, GRU and CNN. Here the proposed hybrid DL models CNN-LSTM maintain better performance than the other DL models for predictions of water quality parameters. In this dataset, our model attained the required accuracy and generalization in 50 epochs. Furthermore, the accuracy loss is minimal after increasing epochs, and the CNN-LSTM model is neither over-fitting nor under-fitting the training data as we increase the epochs from 10 to 500 epochs.

#### 2) WINDOW SIZE

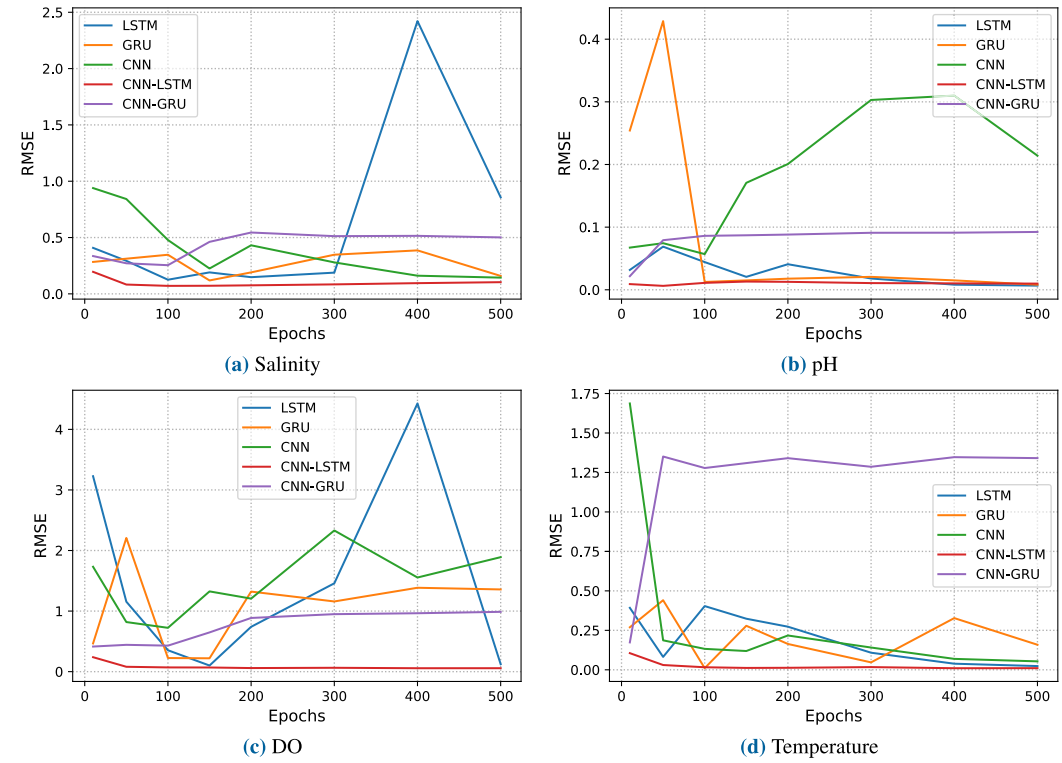
The influence of choosing different window size {10, 20, 30, 40, 50, 60, 70, 80} for each DL model is analysed. Fig. 8 shows the RMSE vs window size for different water quality parameters on the ADAK water quality dataset with the proposed hybrid DL models and the baseline DL models. The proposed hybrid DL models perform better than baseline DL models. As the window size is increased from 10 to 80, the error remains constant for hybrid DL models. In this experiment, we can see those baseline models are comparable in performance with hybrid DL models for some window sizes for each water quality parameter. However, no baseline models are performing consistently well for all four water quality parameters. For example, the performance of LSTM is slightly better than CNN-LSTM at window size 40 and 80 for pH. However, for the other three water quality parameters, the performance of LSTM is not good. In comparison, proposed hybrid models consistently show good performance for the




(a) Salinity (b) pH

(c) DO (d) Temperature

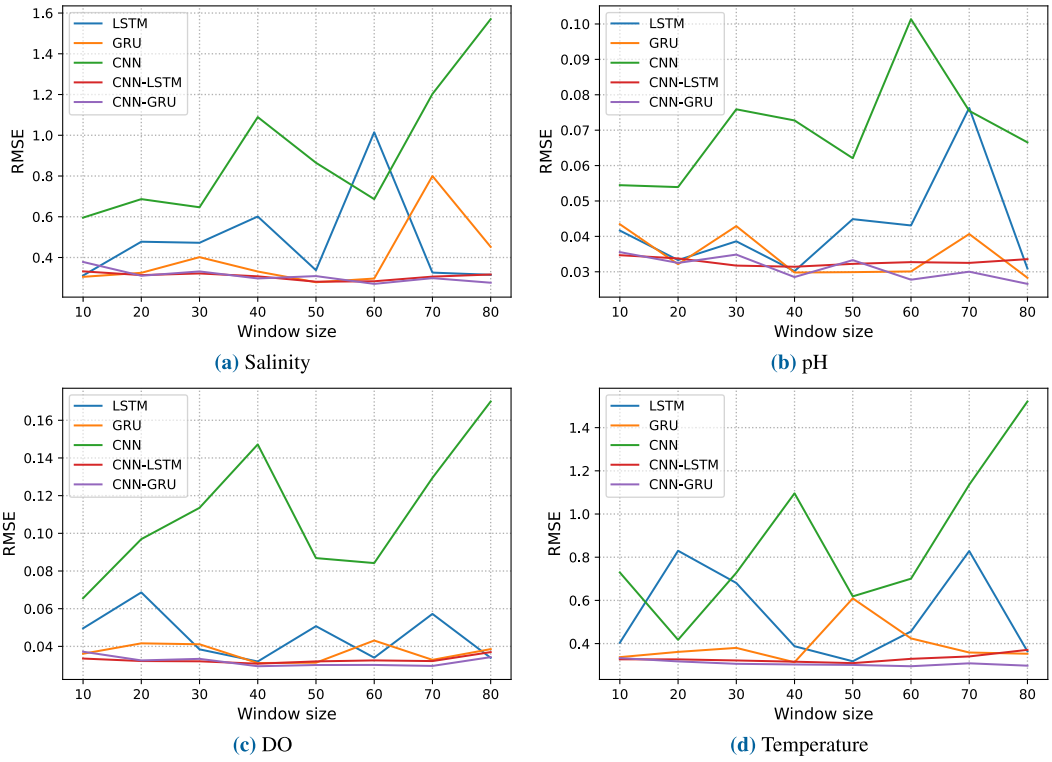
**FIGURE 6.** Comparison of RMSE values for prediction using LSTM, GRU, CNN, CNN-LSTM and CNN-GRU DL models for different epochs {10, 100, 150, 200, 300, 400, 500 } and other hyper parameters constant (window size = 30, learning rate = 0.0008, batch size = 32) on the ADAK dataset.




(a) Salinity (b) pH

(c) DO (d) Temperature

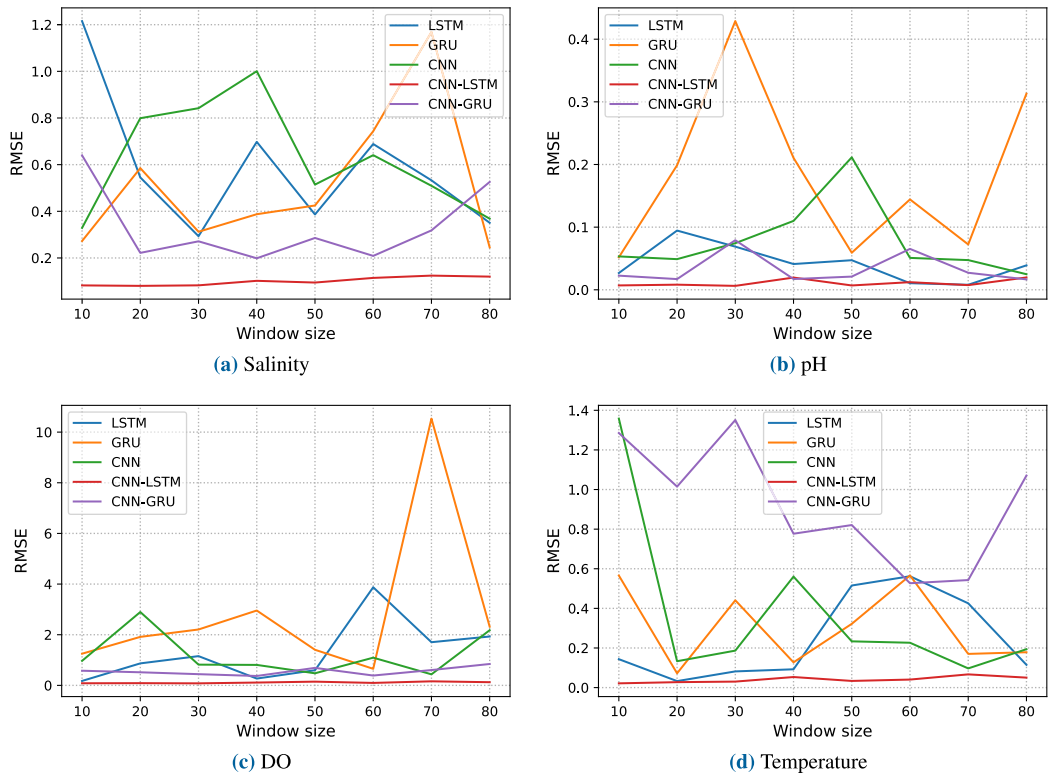
**FIGURE 7.** Comparison of RMSE values for prediction using LSTM, GRU, CNN, CNN-LSTM and CNN-GRU DL models for different epochs {10, 100, 150, 200, 300, 400, 500 } and other hyper parameters constant (window size = 30, learning rate = 0.0008, batch size = 64) on the MAC dataset.




(a) Salinity (b) pH

(c) DO (d) Temperature

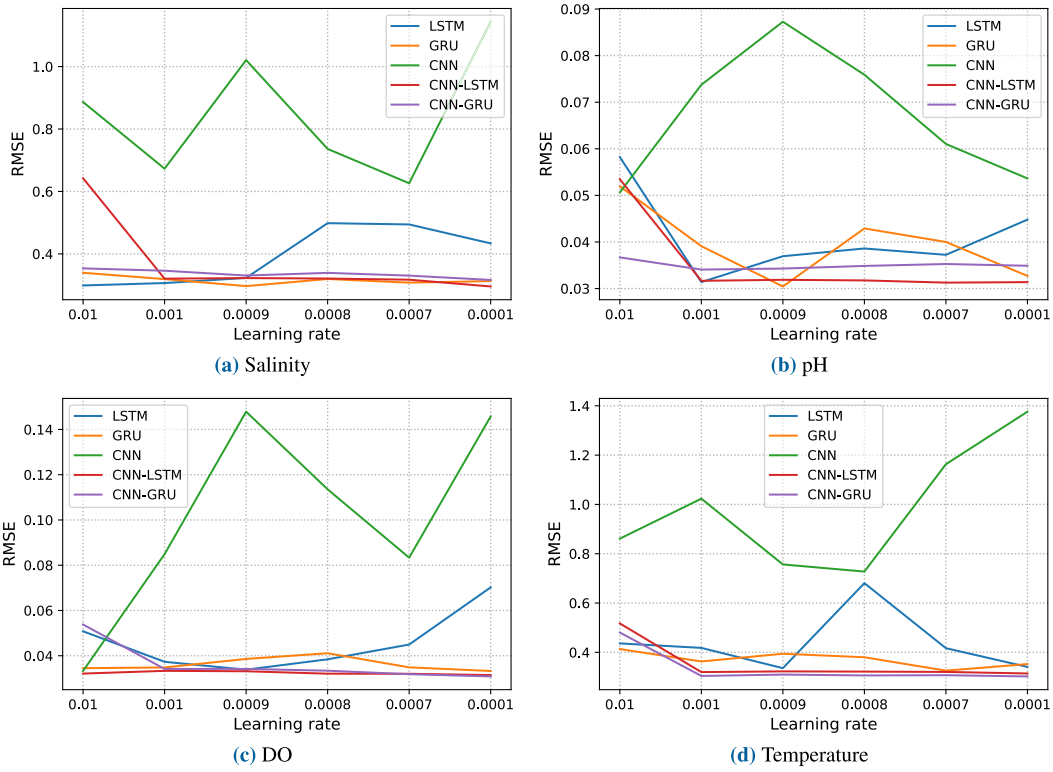
**FIGURE 8.** Comparison of RMSE values for prediction using LSTM, GRU, CNN, CNN-LSTM and CNN-GRU DL models for different Window sizes {10, 20, 30, 40, 50, 60, 70, 80 } and other hyper parameters constant (epoch = 100, learning rate = 0.0008, batch size = 32) on the ADAK dataset.




(a) Salinity (b) pH

(c) DO (d) Temperature

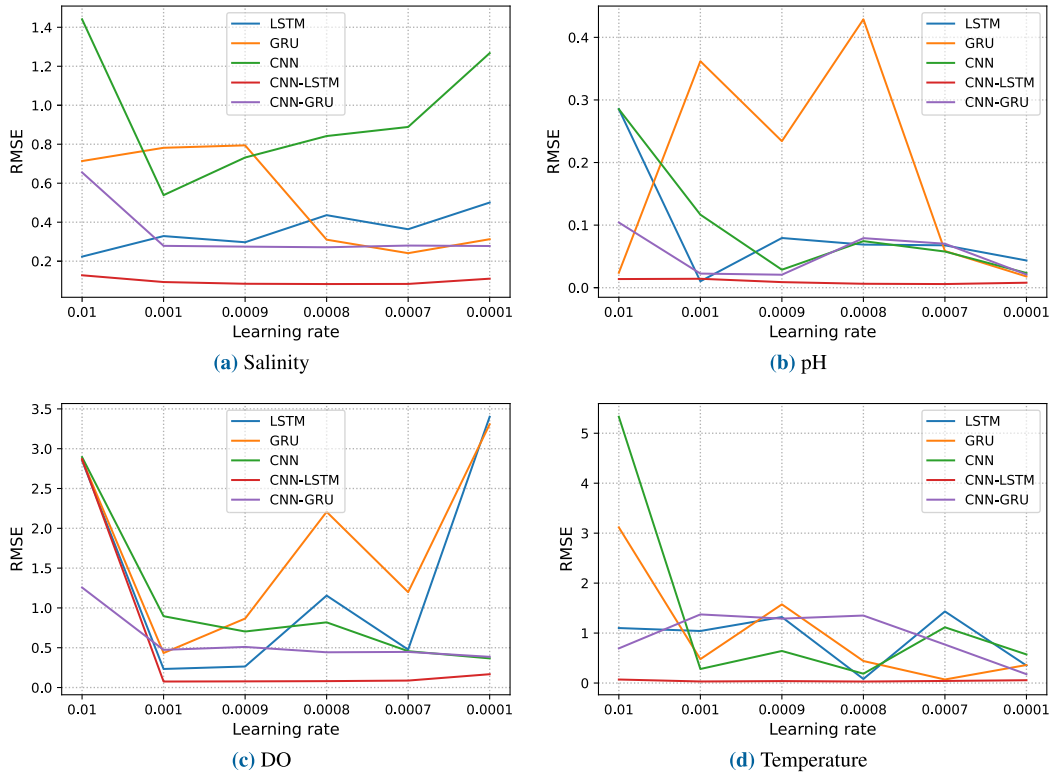
**FIGURE 9.** Comparison of RMSE values for prediction using LSTM, GRU, CNN, CNN-LSTM and CNN-GRU DL models for different Window sizes {10, 20, 30, 40, 50, 60, 70, 80 } and other hyper parameters constant (epoch = 50, learning rate = 0.0008, batch size = 64) on the MAC dataset.




(a) Salinity (b) pH

(c) DO (d) Temperature

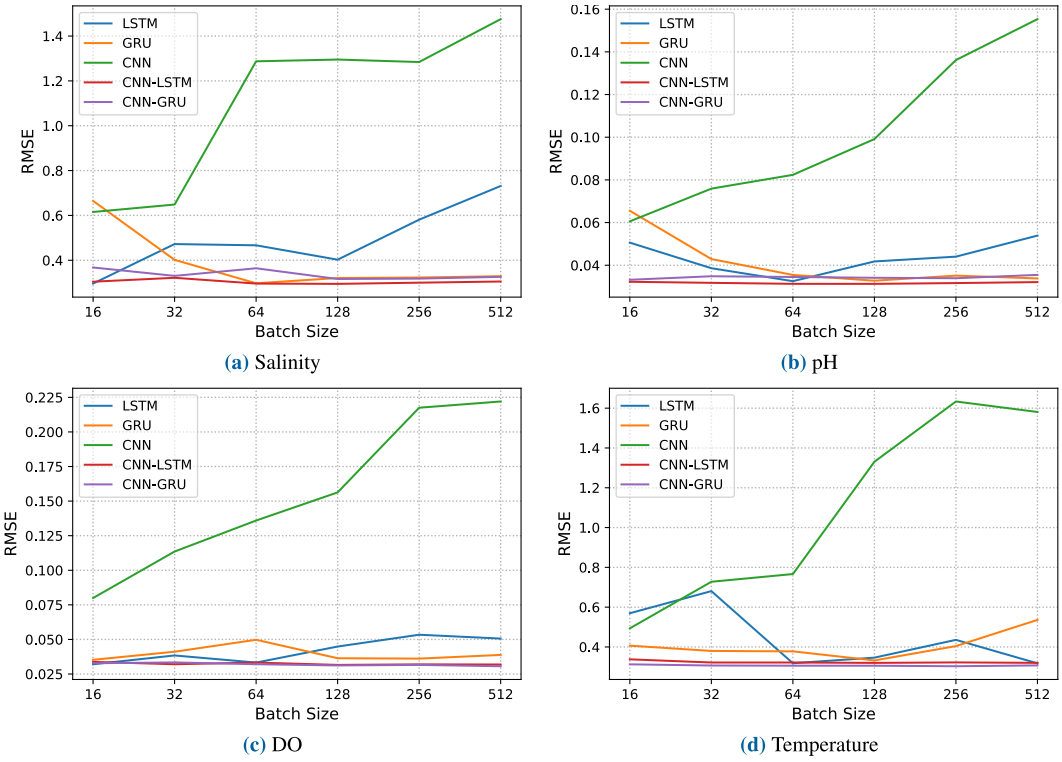
**FIGURE 10.** Comparison of RMSE values for prediction using LSTM, GRU, CNN, CNN-LSTM and CNN-GRU DL models for different learning rates {0.01, 0.001, 0.0009, 0.0008, 0.0007, 0.0001 } and other hyper parameters constant (epoch = 100, window size =30, batch size = 32) on the ADAK dataset.




(a) Salinity (b) pH

(c) DO (d) Temperature

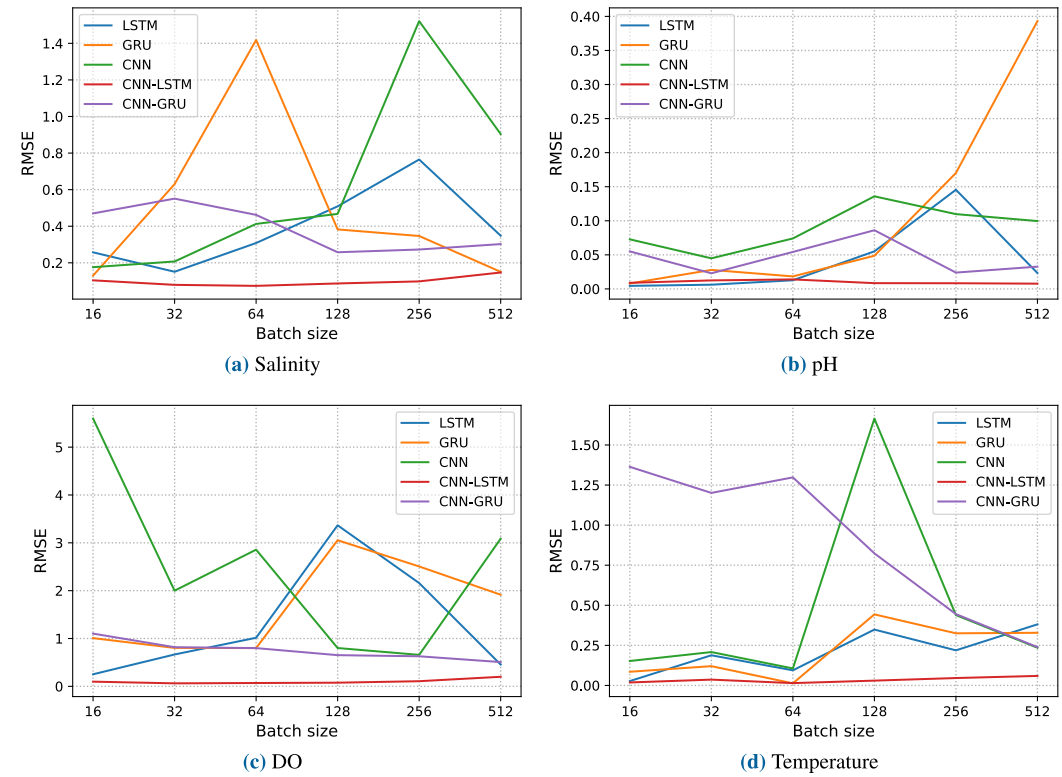
**FIGURE 11.** Comparison of RMSE values for prediction using LSTM, GRU, CNN, CNN-LSTM and CNN-GRU DL models for different learning rates {0.01, 0.001, 0.0009, 0.0008, 0.0007, 0.0001 } and other hyper parameters constant (epoch = 50, window size =30, batch size = 64) on the MAC dataset.




(a) Salinity (b) pH

(c) DO (d) Temperature

**FIGURE 12.** Comparison of RMSE values for prediction using LSTM, GRU, CNN, CNN-LSTM and CNN-GRU DL models for different batch sizes {16, 32, 64, 128, 256, 512 } and other hyper parameters constant (epoch = 100, window size = 30, learning rate = 0.0008) on the ADAK dataset.




(a) Salinity (b) pH

(c) DO (d) Temperature

**FIGURE 13.** Comparison of RMSE values for prediction using LSTM, GRU, CNN, CNN-LSTM and CNN-GRU DL models for different batch sizes {16, 32, 64, 128, 256, 512} and other hyper parameters constant (epoch = 50, window size = 30, learning rate = 0.0008) on the MAC dataset.

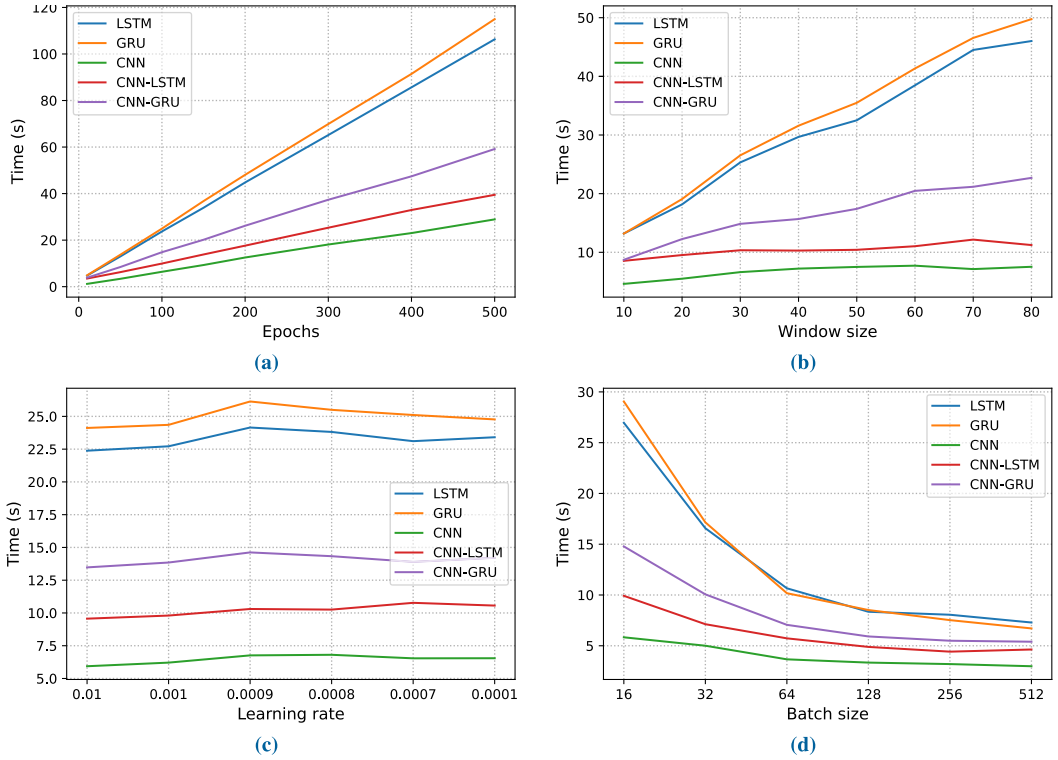



**FIGURE 14.** Comparison of computation time for prediction using LSTM, GRU, CNN, CNN-LSTM and CNN-GRU DL models. (a) epochs {10, 100, 150, 200, 300, 400, 500}, (b) window sizes {10, 20, 30, 40, 50, 60, 70, 80}, (c) learning rates {0.01, 0.001, 0.0009, 0.0008, 0.0007, 0.0001}, and (d) batch sizes {16, 32, 64, 128, 256, 512} on ADAK dataset.

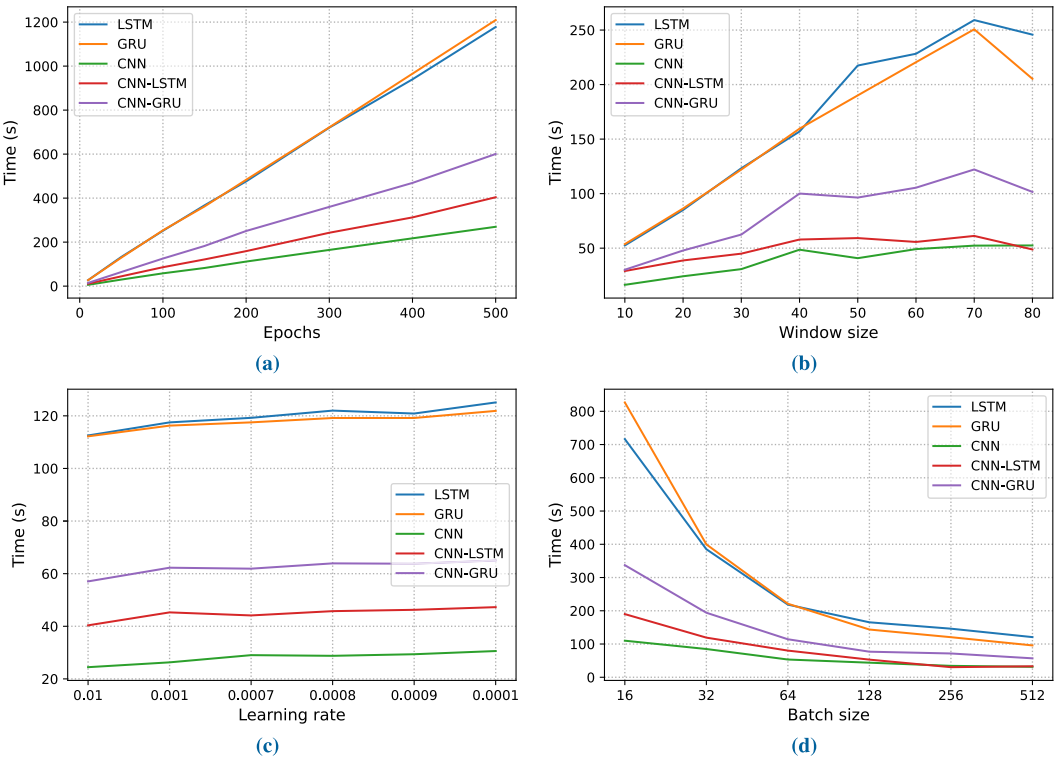



**FIGURE 15.** Comparison of computation time for prediction using LSTM, GRU, CNN, CNN-LSTM and CNN-GRU DL models. (a) epochs {10, 100, 150, 200, 300, 400, 500}, (b) window sizes {10, 20, 30, 40, 50, 60, 70, 80}, (c) learning rates {0.01, 0.001, 0.0009, 0.0008, 0.0007, 0.0001}, and (d) batch sizes {16, 32, 64, 128, 256,512} on MAC datset.

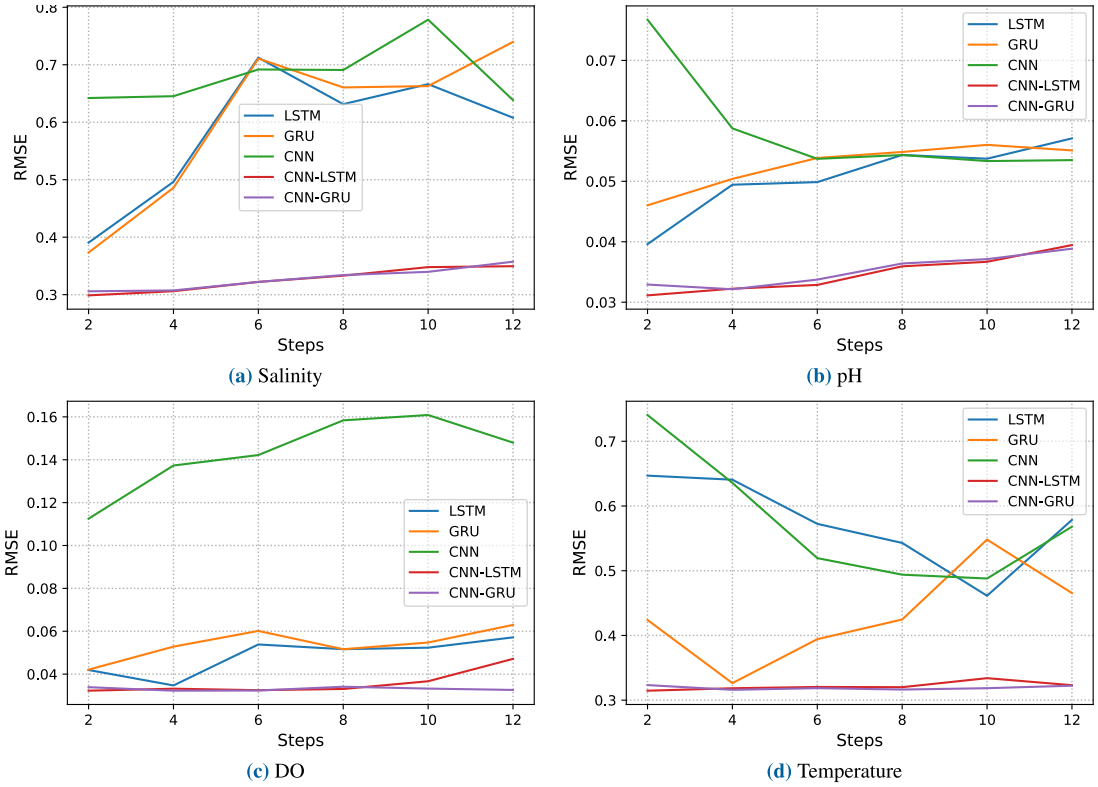



**FIGURE 16.** Comparison of RMSE values for prediction using LSTM, GRU, CNN, CNN-LSTM and CNN-GRU DL for different Step sizes {2, 4, 6, 8, 10, 12} and other hyper parameters constant (epoch = 100, learning rate = 0.0008, window size =30, batch size = 32) on the ADAK dataset.

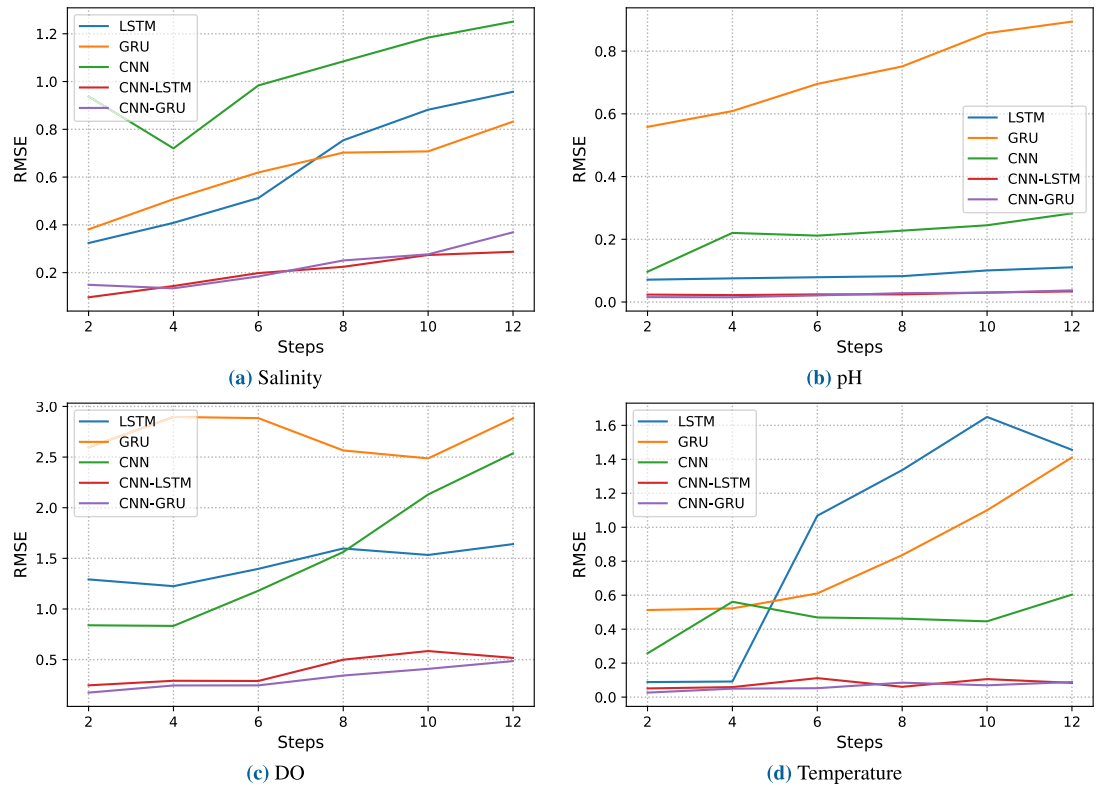



**FIGURE 17.** Comparison of RMSE values for prediction using LSTM, GRU, CNN, CNN-LSTM and CNN-GRU DL models for different step sizes {2, 4, 6, 8, 10, 12} and other hyper parameters constant (epoch = 50, learning rate = 0.0008, window size =30, batch size = 64) on the MAC dataset.

four water quality parameters and window sizes that we experimented with.

To further analyse the performance of these models, we repeat the experiment with the MAC water quality dataset. Fig. 9 shows the RMSE vs window size for all four water quality parameters on the MAC dataset with the proposed hybrid DL models, LSTM, GRU as well as CNN. Furthermore, we can observe from Fig. 9 that the hybrid CNN-LSTM model is having good performance consistently when compared to the other models.

#### 3) LEARNING RATE

The impact of learning rate {0.01, 0.001, 0.0009, 0.0008, 0.0007, 0.0001} on the performance for each of the DL models is analysed. In Fig. 10, we compare RMSE vs learning rate performance of proposed hybrid DL models with baseline models using the ADAK water quality dataset. The proposed hybrid DL models perform consistently better than baseline DL models. However, the performance of baseline DL models is not consistent for various learning rates and different water quality parameters. Hence using the baseline DL models for WQP is not practical.

To further analyse the performance of these models, we repeat the experiment with the MAC water quality dataset. In Fig. 11, we compare RMSE vs learning rate performance of proposed hybrid DL models with baseline models using the MAC water quality dataset. LSTM, GRU and CNN-GRU show moderate performance at some learning rates. However, the performance of CNN does not come close to the performance of hybrid models at any point. Here, we can observe from Fig. 11 that the hybrid CNN-LSTM model has the best and consistent performance compared to other models.

#### 4) BATCH SIZE

In this experiment, we analyse the performance of each DL model on different batch sizes {16, 32, 64, 128, 256, 512}. Fig. 12 shows the RMSE vs batch size for different water quality parameters on the ADAK water quality dataset for the proposed hybrid DL models with the baseline DL models. Compared to the proposed hybrid DL models, it can be observed that the performance of the baseline DL models is inconsistent and inferior. Furthermore, we can see that the performance of CNN is reducing with the increase of the batch size.

Fig. 13 shows the RMSE vs batch size for different water quality parameters using the MAC water quality dataset with the proposed hybrid DL models and the baseline DL models. As this dataset has more data points, we can see from the plots the performance difference of each model. From the results, it can be observed that the proposed hybrid CNN-LSTM models have better performance compared to other baseline models and the hybrid CNN-GRU model. Here also, the baseline DL models are inconsistent and underperform compared to the proposed hybrid DL models.

#### 5) COMPUTATION TIME

In the above four experiments, the computation time was also simultaneously measured and stored. Fig. 14 and Fig. 15 plot the computation time for each hyperparameter for the ADAK water quality dataset and MAC water quality dataset, respectively.The computation time required by each DL model is different. Here we can observe some common trends from the plots. One among them is that the computation time increases with an increase in epochs and window size. But computation time is reduced with an increase in batch size. We cannot notice much change in computation time for different learning rates since we have selected the learning rate within a limited range for both of the datasets for all models.

In summary, for WQP, the proposed CNN-LSTM model maintains the best performance in various experimental conditions in terms of accuracy and computation time, indicating the adaptability of the proposed CNN-LSTM DL model.

#### 6) MULTI-STEPS PREDICTION

In this experiment, we have analysed the performance of each of the DL models for different step sizes {2,4,6,8,10,12}. Fig. 16 shows the RMSE vs step size for various water quality parameters in the ADAK water quality dataset for the proposed hybrid DL models with the baseline DL models. The results show that the proposed hybrid DL models perform much better than the baseline DL models. Furthermore, we can see that the performance of CNN-LSTM and CNN-GRU is reducing with the increase in the step size.

Fig. 17 shows the RMSE vs step size for different water quality parameters in the MAC water quality dataset with the proposed hybrid DL models and the baseline DL models. From the results, it can be observed that the proposed hybrid CNN-LSTM models have better performance compared to other baseline models and the hybrid CNN-GRU model. Compared to the proposed hybrid DL models for both datasets, the baseline DL models are underperforming.

## V. RESULT ANALYSIS

The objective is to analyse the performance of hybrid DL models, CNN-LSTM and CNN-GRU, in comparison with LSTM, GRU, CNN, attention-based LSTM and attention-based GRU DL models to predict aquaculture water quality parameters (salinity, ph, DO and temperature). In this section, the performance of proposed hybrid DL models is compared with baseline DL models and attention-based DL models with a fixed set of $h_p$ on the two datasets.

The $h_p$ is selected by studying the performance of the DL models for various hyperparameters (epochs, window size, learning rate and batch size) in terms of prediction accuracy and computation time. The performance of each model is studied through various experiments by varying

the hyperparameters. The optimal set of hyperparameters in terms of prediction accuracy and computation time is selected to run the final experiment for comparing the performance of all the DL models.

### A. ADAK WATER QUALITY DATASET

Fig. 18 compares the predicted values with the true data on the ADAK water quality dataset using different DL models. All these models are trained using 80% of the ADAK water quality dataset and tested with 20% of the data. The predicted values are compared with the true data. Here we have a fixed set of $h_p$ (epochs = 100, learning rate = 0.0008, batch size = 32, window size = 30 and Adam optimizer). The models are evaluated using performance metrics MAE, MSE, RMSE, and MAPE in the training and testing periods. The computation time required for each of the models is also calculated.

For the ADAK water quality dataset, the prediction performance of all models is shown in Table 5. From the results its clear that the proposed CNN-LSTM model (MAE = 0.2509 ppt, MSE = 0.1035 ppt, RMSE = 0.3217 ppt and MAPE = 0.0164 ppt) outperforms all baseline DL models in the training and testing period. For the salinity prediction, the LSTM model (MAE = 0.3942 ppt, MSE = 0.2230 ppt, RMSE = 0.4722 ppt and MAPE = 0.0257 ppt) outperforms the CNN model (MAE = 0.4800 ppt, MSE = 0.4180 ppt, RMSE = 0.6466 ppt and MAPE = 0.0312 ppt). When we compare the performance of WQP in other works [49], we can see similar results where LSTM have outperformed CNN in prediction accuracy. However, it is noticeable that the computation time required for training CNN is just 5.55s which is only 23.64% the computation time required by LSTM, which requires 23.72s. The LSTM model has the ability to learn from long-term dependencies, and CNN can learn features from a large dataset. However, the hybrid CNN-LSTM requires a computation time of 10.27s only for training the model, which is 43.75% of the computation time of LSTM saving 56.24% in computation time and outperforming all other models in terms of prediction accuracy except the attention-based LSTM and GRU models. The attention-based models have slightly better performance than the hybrid models. But the computation time required for training the attention-based LSTM is 21.54s, and the attention-based GRU is 22.57s which is more than twice the computation time required by the hybrid CNN-LSTM model.

For pH also, the CNN-LSTM model (MAE = 0.0237, MSE = 0.0010, RMSE = 0.0317 and MAPE = 0.0032) outperforms all other baseline DL models in terms of prediction accuracy. The same performance is repeated for DO (MAE = 0.0250 ml/L, MSE = 0.0010 ml/L, RMSE = 0.0321 ml/L, and MAPE = 0.0047 ml/L) and temperature (MAE = 0.2471 °C, MSE = 0.1036 °C, RMSE = 0.3219 °C, and MAPE = 0.0098 °C) prediction as well. Also, the computation time required for CNN-LSTM is around 45% of the computation time needed by LSTM, GRU, attention-based LSTM and attention-based GRU models. CNN-LSTM outperforms CNN with prediction accuracy even though the CNN model requires lesser computational time when compared to CNN-LSTM. The attention-based models have slightly better performance than the hybrid CNN-LSTM models in terms of prediction accuracy. The hybrid model outperforms them in training time. Hence, we can conclude that CNN-LSTM is the best DL model for ADAK water quality data.

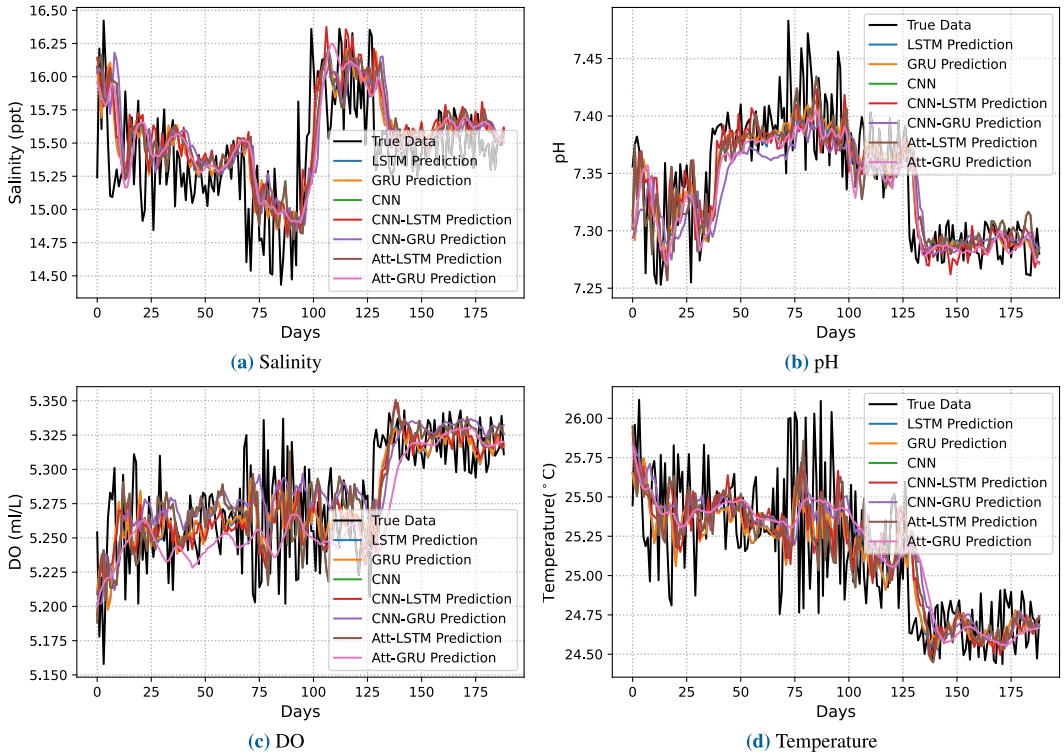
We have also compared the performance of classical model ARIMA with all the DL models for the ADAK dataset in Table 7. The results show that all the DL models perform better than the classical model ARIMA for the ADAK dataset.

### B. MAC WATER QUALITY DATASET

Fig. 19 compares the prediction of all DL models with the true data on the MAC water quality dataset. All these models are trained using 80% of the ADAK water quality dataset and predicted the 20% values. The predictions are compared with the true data. Here we have a fixed set of $h_p$ (epochs = 50, learning rate = 0.0008, batch size = 64, window size = 30 and Adam optimizer). The models are evaluated using MAE, MSE, RMSE, and MAPE in the training and testing periods. The computation time required for each of the models is also calculated.

For the MAC water quality dataset, the prediction performance of all models is shown in Table 6. From the table, it is clear that the proposed CNN-LSTM model (MAE = 0.0381 ppt, MSE = 0.0068 ppt, RMSE = 0.0827 ppt and MAPE = 0.0011 ppt) outperforms all baseline DL models in the training and testing period. For the salinity prediction, the LSTM model (MAE = 0.0535 ppt, MSE = 0.0860 ppt, RMSE = 0.2933 ppt and MAPE = 0016 ppt) outperforms the CNN model (MAE = 0.4783 ppt, MSE = 0.7089 ppt, RMSE = 0.8240 ppt and MAPE = 0.0141 ppt). But it is noticeable that the computation time required for training CNN is just 21.88s which is only 19.57% the computation time required by LSTM, which requires 111.75s. The hybrid CNN-LSTM requires a computation time of 36.83s only for training the model, which is 32.95% of LSTM saving 67.04% in computation time and outperforming all other models in terms of prediction accuracy except the attention-based LSTM and GRU models. The attention-based models have slightly better performance than the hybrid models. But the computation time required for training the attention-based LSTM is 122.42s, and the attention-based GRU is 117.30s, which is more than three times the computation time required by the hybrid CNN-LSTM model.

For pH also, the CNN-LSTM model (MAE = 0.0042, MSE = 0.00004, RMSE = 0.0063 and MAPE = 0.0005) outperforms all other models in terms of prediction accuracy. The same performance is repeated for DO (MAE = 0.0566 ml/L, MSE = 0.0065 ml/L, RMSE = 0.0804 ml/L, and MAPE = 0.0145 ml/L) and temperature (MAE =



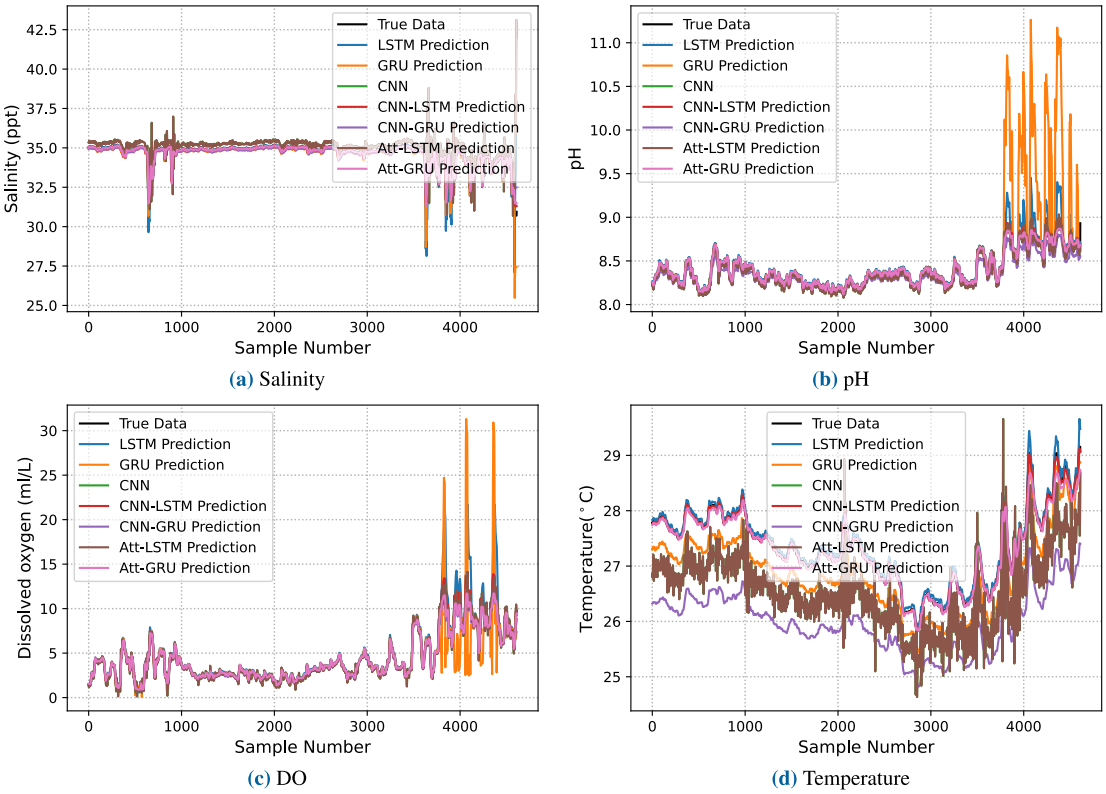

(a) Salinity (b) pH (c) DO (d) Temperature

**FIGURE 18.** Plots of predicted values using LSTM, GRU, CNN, CNN-LSTM, CNN-GRU, Attention-based LSTM and Attention-based GRU predictions, and ADAK water quality observed data. (a) Salinity, (b) pH, (c) DO, (d) Temperature.

**TABLE 5.** Evaluation of Prediction accuracy of LSTM, GRU, CNN, CNN-LSTM, CNN-GRU, Attention-based LSTM and Attention-based GRU for ADAK water quality dataset.

| Water Quality Parameter | Model | Training Period | | | | | Testing Period | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | MAE | MSE | RMSE | MAPE | Time (s) | MAE | MSE | RMSE | MAPE | Time (s) |
| Salinity (ppt) | LSTM | 0.2741 | 0.1287 | 0.3588 | 0.0184 | 23.47 | 0.3942 | 0.2230 | 0.4722 | 0.0257 | 0.05 |
| | GRU | 0.3550 | 0.1843 | 0.4293 | 0.0237 | 25.02 | 0.3270 | 0.1614 | 0.4017 | 0.0212 | 0.05 |
| | CNN | 0.4180 | 0.2911 | 0.5396 | 0.0282 | 05.55 | 0.4800 | 0.4180 | 0.6466 | 0.0312 | 0.04 |
| | **CNN-LSTM** | **0.2274** | **0.0917** | **0.3028** | **0.0153** | **10.27** | **0.2509** | **0.1035** | **0.3217** | **0.0164** | **0.04** |
| | CNN-GRU | 0.2434 | 0.1066 | 0.3264 | 0.0164 | 14.00 | 0.2475 | 0.1100 | 0.3317 | 0.0161 | 0.04 |
| | Att-LSTM | 0.2257 | 0.0967 | 0.3110 | 0.0151 | 21.54 | 0.2376 | 0.0959 | 0.3097 | 0.0154 | 0.05 |
| | Att-GRU | 0.2220 | 0.0919 | 0.3032 | 0.0148 | 22.57 | 0.2227 | 0.0836 | 0.2891 | 0.0144 | 0.05 |
| pH | LSTM | 0.0321 | 0.0017 | 0.0409 | 0.0044 | 22.72 | 0.0300 | 0.0015 | 0.0386 | 0.0041 | 0.05 |
| | GRU | 0.0341 | 0.0018 | 0.0427 | 0.0046 | 25.32 | 0.0353 | 0.0018 | 0.0429 | 0.0048 | 0.05 |
| | CNN | 0.0498 | 0.0037 | 0.0608 | 0.0068 | 05.49 | 0.0600 | 0.0058 | 0.0759 | 0.0082 | 0.04 |
| | **CNN-LSTM** | **0.0220** | **0.0009** | **0.0293** | **0.0030** | **10.37** | **0.0237** | **0.0010** | **0.0317** | **0.0032** | **0.04** |
| | CNN-GRU | 0.0259 | 0.0012 | 0.0345 | 0.0035 | 14.93 | 0.0269 | 0.0012 | 0.0349 | 0.0037 | 0.04 |
| | Att-LSTM | 0.0238 | 0.0010 | 0.0324 | 0.0032 | 21.84 | 0.0236 | 0.0010 | 0.0312 | 0.0032 | 0.05 |
| | Att-GRU | 0.0231 | 0.0010 | 0.0317 | 0.0031 | 24.50 | 0.0230 | 0.0009 | 0.0306 | 0.0031 | 0.05 |
| DO (ml/L) | LSTM | 0.0290 | 0.0014 | 0.0374 | 0.0055 | 24.64 | 0.0295 | 0.0015 | 0.0384 | 0.0056 | 0.05 |
| | GRU | 0.0331 | 0.0017 | 0.0414 | 0.0062 | 25.49 | 0.0319 | 0.0017 | 0.0411 | 0.0060 | 0.05 |
| | CNN | 0.1072 | 0.0151 | 0.1229 | 0.0201 | 06.13 | 0.0974 | 0.0129 | 0.1135 | 0.0185 | 0.04 |
| | **CNN-LSTM** | **0.0226** | **0.0009** | **0.0292** | **0.0042** | **10.68** | **0.0250** | **0.0010** | **0.0321** | **0.0047** | **0.04** |
| | CNN-GRU | 0.0258 | 0.0012 | 0.0346 | 0.0049 | 14.21 | 0.0262 | 0.0011 | 0.0334 | 0.0050 | 0.04 |
| | Att-LSTM | 0.0244 | 0.0010 | 0.0321 | 0.0046 | 23.04 | 0.0242 | 0.0010 | 0.0313 | 0.0046 | 0.05 |
| | Att-GRU | 0.0239 | 0.0010 | 0.0314 | 0.0045 | 24.35 | 0.0237 | 0.0010 | 0.0309 | 0.0045 | 0.05 |
| Temperature(°C) | LSTM | 0.5552 | 0.4080 | 0.6388 | 0.0222 | 24.40 | 0.6071 | 0.4633 | 0.6807 | 0.0240 | 0.05 |
| | GRU | 0.2852 | 0.1357 | 0.3684 | 0.0114 | 24.88 | 0.2951 | 0.1444 | 0.3800 | 0.0116 | 0.05 |
| | CNN | 0.5346 | 0.5422 | 0.7363 | 0.0217 | 05.51 | 0.5229 | 0.5295 | 0.7276 | 0.0207 | 0.04 |
| | **CNN-LSTM** | **0.2252** | **0.0846** | **0.2908** | **0.0091** | **10.60** | **0.2471** | **0.1036** | **0.3219** | **0.0098** | **0.04** |
| | CNN-GRU | 0.2560 | 0.1162 | 0.3409 | 0.0103 | 14.08 | 0.2410 | 0.0938 | 0.3063 | 0.0096 | 0.04 |
| | Att-LSTM | 0.2738 | 0.1281 | 0.3579 | 0.0110 | 22.26 | 0.2387 | 0.0897 | 0.2995 | 0.0095 | 0.05 |
| | Att-GRU | 0.2484 | 0.1057 | 0.3251 | 0.0100 | 23.34 | 0.2394 | 0.0918 | 0.3030 | 0.0095 | 0.05 |

Hyper parameter settings: epochs 100, lookup size 30, batch size 32, learning rate 0.0008 and ADAM optimizer.




(a) Salinity (b) pH (c) DO (d) Temperature

**FIGURE 19.** Plots of predicted values using LSTM, GRU, CNN, CNN-LSTM, CNN-GRU, Attention-based LSTM and Attention-based GRU predictions, and MAC water quality observed data.(a) Salinity, (b) pH, (c) DO, (d) Temperature.

**TABLE 6.** Evaluation of Prediction accuracy of LSTM, GRU, CNN, CNN-LSTM, CNN-GRU, Attention-based LSTM and Attention-based GRU for MAC water quality dataset.

| Water Quality Parameter | Model | Training Period | | | | | Testing Period | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | MAE | MSE | RMSE | MAPE | Time (s) | MAE | MSE | RMSE | MAPE | Time (s) |
| Salinity (ppt) | LSTM | 0.0173 | 0.0010 | 0.0320 | 0.0005 | 111.75 | 0.0535 | 0.0860 | 0.2933 | 0.0016 | 0.35 |
| | GRU | 0.0210 | 0.0007 | 0.0258 | 0.0006 | 114.13 | 0.0571 | 0.0975 | 0.3123 | 0.0018 | 0.31 |
| | CNN | 0.3612 | 0.1455 | 0.3815 | 0.0104 | 021.88 | 0.4783 | 0.7089 | 0.8420 | 0.0141 | 0.11 |
| | **CNN-LSTM** | **0.0241** | **0.0011** | **0.0338** | **0.0007** | **36.83** | **0.0381** | **0.0068** | **0.0827** | **0.0011** | **0.17** |
| | CNN-GRU | 0.0989 | 0.0159 | 0.1259 | 0.0029 | 055.74 | 0.1460 | 0.0736 | 0.2714 | 0.0043 | 0.19 |
| | Att-LSTM | 0.0324 | 0.0037 | 0.0606 | 0.0009 | 122.42 | 0.0536 | 0.0180 | 0.1341 | 0.0016 | 0.38 |
| | Att-GRU | 0.0553 | 0.0050 | 0.0705 | 0.0016 | 117.30 | 0.0706 | 0.0155 | 0.1246 | 0.0021 | 0.34 |
| pH | LSTM | 0.0166 | 0.0003 | 0.0175 | 0.0020 | 116.86 | 0.0339 | 0.0047 | 0.0689 | 0.0039 | 0.35 |
| | GRU | 0.0041 | 0.0001 | 0.0101 | 0.0005 | 115.93 | 0.1358 | 0.1839 | 0.4288 | 0.0154 | 0.31 |
| | CNN | 0.0611 | 0.0042 | 0.0645 | 0.0074 | 022.56 | 0.0696 | 0.0055 | 0.0744 | 0.0083 | 0.11 |
| | **CNN-LSTM** | **0.0031** | **0.00002** | **0.0045** | **0.0004** | **039.07** | **0.0042** | **0.00004** | **0.0063** | **0.0005** | **0.17** |
| | CNN-GRU | 0.0429 | 0.0024 | 0.0492 | 0.0051 | 058.74 | 0.0673 | 0.0063 | 0.0792 | 0.0079 | 0.19 |
| | Att-LSTM | 0.0069 | 0.00008 | 0.0089 | 0.0008 | 132.60 | 0.0110 | 0.0003 | 0.0183 | 0.0013 | 0.40 |
| | Att-GRU | 0.0047 | 0.00004 | 0.0065 | 0.0006 | 125.74 | 0.0087 | 0.0003 | 0.0166 | 0.0010 | 0.36 |
| DO (ml/L) | LSTM | 0.3148 | 0.1310 | 0.3620 | 0.0803 | 116.31 | 0.4767 | 1.3341 | 1.1550 | 0.0873 | 0.36 |
| | GRU | 0.2235 | 0.0762 | 0.2760 | 0.0569 | 119.47 | 0.6789 | 4.8710 | 2.2070 | 0.0785 | 0.32 |
| | CNN | 0.3300 | 0.1454 | 0.3813 | 0.0866 | 023.20 | 0.4492 | 0.6705 | 0.8188 | 0.0952 | 0.11 |
| | **CNN-LSTM** | **0.0467** | **0.0033** | **0.0572** | **0.0129** | **040.76** | **0.0566** | **0.0065** | **0.0804** | **0.0145** | **0.18** |
| | CNN-GRU | 0.1094 | 0.0263 | 0.1623 | 0.0346 | 060.21 | 0.2439 | 0.1961 | 0.4428 | 0.0509 | 0.20 |
| | Att-LSTM | 0.1129 | 0.0202 | 0.1423 | 0.0330 | 139.23 | 0.2219 | 0.1721 | 0.4148 | 0.0464 | 0.40 |
| | Att-GRU | 0.0467 | 0.0046 | 0.0675 | 0.0146 | 125.98 | 0.1463 | 0.0982 | 0.3133 | 0.0268 | 0.36 |
| Temperature(°C) | LSTM | 0.0663 | 0.0053 | 0.0728 | 0.0026 | 118.71 | 0.0672 | 0.0067 | 0.0818 | 0.0025 | 0.36 |
| | GRU | 0.3977 | 0.1607 | 0.4009 | 0.0157 | 120.67 | 0.4373 | 0.1943 | 0.4408 | 0.0160 | 0.32 |
| | CNN | 0.1138 | 0.0333 | 0.1825 | 0.0046 | 023.76 | 0.1441 | 0.0349 | 0.1867 | 0.0052 | 0.12 |
| | **CNN-LSTM** | **0.0166** | **0.0006** | **0.0246** | **0.0007** | **043.21** | **0.0216** | **0.0009** | **0.0303** | **0.0008** | **0.18** |
| | CNN-GRU | 0.8879 | 0.9603 | 0.9799 | 0.0340 | 062.96 | 1.3416 | 1.8252 | 1.3510 | 0.0489 | 0.20 |
| | Att-LSTM | 0.0316 | 0.0022 | 0.0468 | 0.0013 | 125.28 | 0.0500 | 0.0079 | 0.0889 | 0.0018 | 0.39 |
| | Att-GRU | 0.0284 | 0.0018 | 0.0429 | 0.0011 | 124.11 | 0.0443 | 0.0062 | 0.0789 | 0.0016 | 0.36 |

Hyper parameter settings: epochs 50, lookup size 30, batch size 32, learning rate 0.0008 and ADAM optimizer.

**TABLE 7. Evaluation of Prediction accuracy of ARIMA, LSTM, GRU, CNN, CNN-LSTM, CNN-GRU, Attention-based LSTM and Attention-based GRU for ADAK and MAC water quality datasets.**

| Water Quality Parameter | Model | ADAK | | | | MAC | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | MAE | MSE | RMSE | MAPE | MAE | MSE | RMSE | MAPE |
| Salinity (ppt) | ARIMA | 0.5669 | 0.4531 | 0.6731 | 0.0361 | 0.4715 | 0.4954 | 0.7038 | 0.0139 |
| | LSTM | 0.3942 | 0.2230 | 0.4722 | 0.0257 | 0.0535 | 0.0860 | 0.2933 | 0.0016 |
| | GRU | 0.3270 | 0.1614 | 0.4017 | 0.0212 | 0.0571 | 0.0975 | 0.3123 | 0.0018 |
| | CNN | 0.4800 | 0.4180 | 0.6466 | 0.0312 | 0.4783 | 0.7089 | 0.8420 | 0.0141 |
| | **CNN-LSTM** | **0.2509** | **0.1035** | **0.3217** | **0.0164** | **0.0381** | **0.0068** | **0.0827** | **0.0011** |
| | CNN-GRU | 0.2475 | 0.1100 | 0.3317 | 0.0161 | 0.1460 | 0.0736 | 0.2714 | 0.0043 |
| | Att-LSTM | 0.2376 | 0.0959 | 0.3097 | 0.0154 | 0.0536 | 0.0180 | 0.1341 | 0.0016 |
| | Att-GRU | 0.2227 | 0.0836 | 0.2891 | 0.0144 | 0.0706 | 0.0155 | 0.1246 | 0.0021 |
| pH | ARIMA | 0.0473 | 0.0029 | 0.0541 | 0.0064 | 0.1580 | 0.0512 | 0.2262 | 0.0184 |
| | LSTM | 0.0300 | 0.0015 | 0.0386 | 0.0041 | 0.0339 | 0.0047 | 0.0689 | 0.0039 |
| | GRU | 0.0353 | 0.0018 | 0.0429 | 0.0048 | 0.1358 | 0.1839 | 0.4288 | 0.0154 |
| | CNN | 0.0600 | 0.0058 | 0.0759 | 0.0082 | 0.0696 | 0.0055 | 0.0744 | 0.0083 |
| | **CNN-LSTM** | **0.0237** | **0.0010** | **0.0317** | **0.0032** | **0.0042** | **0.00004** | **0.0063** | **0.0005** |
| | CNN-GRU | 0.0269 | 0.0012 | 0.0349 | 0.0037 | 0.0673 | 0.0063 | 0.0792 | 0.0079 |
| | Att-LSTM | 0.0236 | 0.0010 | 0.0312 | 0.0032 | 0.0110 | 0.0003 | 0.0183 | 0.0013 |
| | Att-GRU | 0.0230 | 0.0009 | 0.0306 | 0.0031 | 0.0087 | 0.0003 | 0.0166 | 0.0010 |
| DO (ml/L) | ARIMA | 0.0556 | 0.0051 | 0.0717 | 0.0106 | 2.2085 | 8.0083 | 2.8299 | 0.6552 |
| | LSTM | 0.0295 | 0.0015 | 0.0384 | 0.0056 | 0.4767 | 1.3341 | 1.1550 | 0.0873 |
| | GRU | 0.0319 | 0.0017 | 0.0411 | 0.0060 | 0.6789 | 4.8710 | 2.2070 | 0.0785 |
| | CNN | 0.0974 | 0.0129 | 0.1135 | 0.0185 | 0.4492 | 0.6705 | 0.8188 | 0.0952 |
| | **CNN-LSTM** | **0.0250** | **0.0010** | **0.0321** | **0.0047** | **0.0566** | **0.0065** | **0.0804** | **0.0145** |
| | CNN-GRU | 0.0262 | 0.0011 | 0.0334 | 0.0050 | 0.2439 | 0.1961 | 0.4428 | 0.0509 |
| | Att-LSTM | 0.0242 | 0.0010 | 0.0313 | 0.0046 | 0.2219 | 0.1721 | 0.4148 | 0.0464 |
| | Att-GRU | 0.0237 | 0.0010 | 0.0309 | 0.0045 | 0.1463 | 0.0982 | 0.3133 | 0.0268 |
| Temperature(°C) | ARIMA | 0.4889 | 0.3901 | 0.6246 | 0.0191 | 1.8037 | 3.7638 | 1.9400 | 0.0654 |
| | LSTM | 0.6071 | 0.4633 | 0.6807 | 0.0240 | 0.0672 | 0.0067 | 0.0818 | 0.0025 |
| | GRU | 0.2951 | 0.1444 | 0.3800 | 0.0116 | 0.4373 | 0.1943 | 0.4408 | 0.0160 |
| | CNN | 0.5229 | 0.5295 | 0.7276 | 0.0207 | 0.1441 | 0.0349 | 0.1867 | 0.0052 |
| | **CNN-LSTM** | **0.2471** | **0.1036** | **0.3219** | **0.0098** | **0.0216** | **0.0009** | **0.0303** | **0.0008** |
| | CNN-GRU | 0.2410 | 0.0938 | 0.3063 | 0.0096 | 1.3416 | 1.8252 | 1.3510 | 0.0489 |
| | Att-LSTM | 0.2387 | 0.0897 | 0.2995 | 0.0095 | 0.0500 | 0.0079 | 0.0889 | 0.0018 |
| | Att-GRU | 0.2394 | 0.0918 | 0.3030 | 0.0095 | 0.0443 | 0.0062 | 0.0789 | 0.0016 |

Hyper parameter settings for ADAK dataset is epochs 100, lookup size 30, batch size 32, learning rate 0.0008 and ADAM optimizer.
Hyper parameter settings for MAC dataset is epochs 50, lookup size 30, batch size 64, learning rate 0.0008 and ADAM optimizer.

0.0216 °C, MSE = 0.0009 °C, RMSE = 0.0303 °C, and MAPE = 0.0008 °C) prediction as well. Also, the computation time required for CNN-LSTM is around 33% of the computation time needed by LSTM, GRU, attention-based LSTM and attention-based GRU models. CNN model has lesser computational time when compared to CNN-LSTM. The attention-based models have slightly better performance than the hybrid CNN-LSTM models in terms of prediction accuracy. The hybrid model outperforms them in training time. Hence, we can conclude that CNN-LSTM is the best DL model for MAC water quality data.

We have also compared the performance of classical model ARIMA with all the DL models for the MAC dataset in Table 7. The results show that all the DL models perform better than the classical model ARIMA for the MAC dataset.

## VI. CONCLUSION

In this research work, we have proposed hybrid DL models, CNN-LSTM and CNN-GRU, for aquaculture WQP. The developed hybrid prediction models were trained and tested on two distinct datasets. The water quality parameters data was collected from aquaculture ponds located in Kollam, Kerala, under ADAK. Another dataset used was the MAC dataset which was collected from the marine aquaculture base in Xincun Town, LingShui County, Hainan Province, China. We have also extensively analysed the impact of varying the $h_p$. For performance comparison and further analysis, optimal $h_p$ were used. We have compared the performance of these hybrid DL models (CNN-LSTM and CNN-GRU) with baseline DL models (LSTM, GRU, and CNN) and attention-based DL models (attention-based LSTM and attention-based GRU) in terms of MAE, MSE, RMSE, and MAPE. Results show that the CNN-LSTM hybrid model provides significant improvement in prediction accuracy as well as computation time compared to the baseline DL models. The hybrid models have a similar performance compared with the attention-based models. Still, they outperform the attention-based models in computation time, offering a realistic solution for predicting water quality parameters in smart aquaculture.

## ACKNOWLEDGMENT

The authors are thankful to the Agency for Development of Aquaculture Kerala (ADAK) for providing the data. They are also thankful to AI for Earth Microsoft Azure Compute Grants.

## REFERENCES

[1] *World Population Projected to Reach 9.8 Billion in 2050, and 11.2 Billion in 2100*, United Nations, New York, NY, USA, Jun. 2017. [Online]. Available: https://www.un.org/development/desa/en/news/population/world-population-prospects-2017.html

[2] S. Jennings, G. D. Stentiford, A. M. Leocadio, K. R. Jeffery, J. D. Metcalfe, I. Katsiadaki, N. A. Auchterlonie, S. C. Mangi, J. K. Pinnegar, T. Ellis, and E. J. Peeler, "Aquatic food security: Insights into challenges and solutions from an analysis of interactions between fisheries, aquaculture, food safety, human health, fish and human welfare, economy and environment," *Fish Fisheries*, vol. 17, no. 4, pp. 893–938, Feb. 2016.

[3] D. Li and C. Li, "Intelligent aquaculture," *J. World Aquaculture Soc.*, vol. 51, no. 4, pp. 808–814, Aug. 2020.

[4] Z. Hu, R. Li, X. Xia, C. Yu, X. Fan, and Y. Zhao, "A method overview in smart aquaculture," *Environ. Monitor. Assessment*, vol. 192, no. 8, pp. 1–25, Jul. 2020.

[5] S. S. Ahuja, "Monitoring water quality, pollution assessment, and remediation to assure sustainability," in *Monitoring Water Quality*, S. Ahuja, Ed. Amsterdam, The Netherlands: Elsevier, 2013, pp. 1–18.

[6] C. E. Boyd and C. S. Tucker, *Pond Aquaculture Water Quality Management*. New York, NY, USA: Springer, 1998.

[7] H. S. Ayele and M. Atlabachew, "Review of characterization, factors, impacts, and solutions of lake eutrophication: Lesson for lake tana, Ethiopia," *Environ. Sci. Pollut. Res.*, vol. 28, no. 12, pp. 14233–14252, Jan. 2021.

[8] J. A. Martos-Sitcha, J. M. Mancera, P. Prunet, and L. J. Magnoni, "Editorial: Welfare and stressors in fish: Challenges facing aquaculture," *Frontiers Physiol.*, vol. 11, p. 162, Feb. 2020.

[9] X. Li, H. Xie, R. Wang, Y. Cai, J. Cao, F. Wang, H. Min, and X. Deng, "Empirical analysis: Stock market prediction via extreme learning machine," *Neural Comput. Appl.*, vol. 27, no. 1, pp. 67–78, Jan. 2016.

[10] X. Zhang, X. Zhang, S. Qu, J. Huang, B. Fang, and P. Yu, "Stock market prediction via multi-source multiple instance learning," *IEEE Access*, vol. 6, pp. 50720–50728, 2018.

[11] G. Chen, C. Ding, Y. Li, X. Hu, X. Li, L. Ren, X. Ding, P. Tian, and W. Xue, "Prediction of chronic kidney disease using adaptive hybridized deep convolutional neural network on the Internet of Medical Things platform," *IEEE Access*, vol. 8, pp. 100497–100508, 2020.

[12] J. Contreras, R. Espinola, F. J. Nogales, and A. J. Conejo, "ARIMA models to predict next-day electricity prices," *IEEE Trans. Power Syst.*, vol. 18, no. 3, pp. 1014–1020, Aug. 2003.

[13] T. Arcomano, I. Szunyogh, J. Pathak, A. Wikner, B. R. Hunt, and E. Ott, "A machine learning-based global atmospheric forecast model," *Geophys. Res. Lett.*, vol. 47, no. 9, 2020, Art. no. e2020GL087776.

[14] Y. Yu, J. Cao, and J. Zhu, "An LSTM short-term solar irradiance forecasting under complicated weather conditions," *IEEE Access*, vol. 7, pp. 145651–145666, 2019.

[15] A. R. S. Parmezan, V. M. A. Souza, and G. E. A. P. A. Batista, "Evaluation of statistical and machine learning models for time series prediction: Identifying the state-of-the-art and the best conditions for the use of each model," *Inf. Sci.*, vol. 484, pp. 302–337, May 2019.

[16] J. Liu, C. Yu, Z. Hu, Y. Zhao, Y. Bai, M. Xie, and J. Luo, "Accurate prediction scheme of water quality in smart mariculture with deep Bi-S-SRU learning network," *IEEE Access*, vol. 8, pp. 24784–24798, 2020.

[17] J. Xie and Q. Wang, "Benchmarking machine learning algorithms on blood glucose prediction for type i diabetes in comparison with classical time-series models," *IEEE Trans. Biomed. Eng.*, vol. 67, no. 11, pp. 3101–3124, Nov. 2020.

[18] Y. Qin, D. Song, H. Chen, W. Cheng, G. Jiang, and G. Cottrell, "A dual-stage attention-based recurrent neural network for time series prediction," 2017, *arXiv:1704.02971*.

[19] Y. Yuan, M. Chao, and Y.-C. Lo, "Automatic skin lesion segmentation using deep fully convolutional networks with Jaccard distance," *IEEE Trans. Med. Imag.*, vol. 36, no. 9, pp. 1876–1886, Sep. 2017.

[20] T. Kourkounakis, A. Hajavi, and A. Etemad, "FluentNet: End-to-end detection of stuttered speech disfluencies with deep learning," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 29, pp. 2986–2999, 2021.

[21] Y. LeCun, Y. Bengio, and G. E. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, Dec. 2015.

[22] Z. Hu, Y. Zhang, Y. Zhao, M. Xie, J. Zhong, Z. Tu, and J. Liu, "A water quality prediction method based on the deep LSTM network considering correlation in smart mariculture," *Sensors*, vol. 19, no. 6, p. 1420, Mar. 2019.

[23] J. Zhao, F. Huang, J. Lv, Y. Duan, Z. Qin, G. Li, and G. Tian, "Do RNN and LSTM have long memory," in *Proc. 37th Int. Conf. Mach. Learn.*, vol. 119, H. D. III and A. Singh, Eds. Jul. 2020, pp. 11365–11375.

[24] W. Zheng and G. Chen, "An accurate GRU-based power time-series prediction approach with selective state updating and stochastic optimization," *IEEE Trans. Cybern.*, early access, Nov. 3, 2021, doi: 10.1109/TCYB.2021.3121312.

[25] W. Zheng, P. Zhao, K. Huang, and G. Chen, "Understanding the property of long term memory for the LSTM with attention mechanism," in *Proc. 30th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2021, pp. 2708–2717.

[26] T. Guo, T. Lin, and N. Antulov-Fantulin, "Exploring interpretable LSTM neural networks over multi-variable data," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 2494–2504.

[27] D. Song, H. Chen, G. Jiang, and Y. Qin, "Dual stage attention based recurrent neural network for time series prediction," U.S. Patent 10 929 674, Feb. 23, 2021.

[28] Y. Qin, D. Song, H. Chen, W. Cheng, G. Jiang, and G. Cottrell, "A dual-stage attention-based recurrent neural network for time series prediction," 2017, *arXiv:1704.02971*.

[29] Y. Liang, S. Ke, J. Zhang, X. Yi, and Y. Zheng, "GeoMAN: Multi-level attention networks for geo-sensory time series prediction," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Jul. 2018, pp. 3428–3434.

[30] C. J. Huang, Y. Shen, Y. H. Chen, and H. C. Chen, "A novel hybrid deep neural network model for short-term electricity price forecasting," *Int. J. Energy Res.*, vol. 45, no. 2, pp. 2511–2532, 2020.

[31] H. Gholamalinezhad and H. Khosravi, "Pooling methods in deep neural networks, a review," 2020, *arXiv:2009.07485*.

[32] J. Bradbury, S. Merity, C. Xiong, and R. Socher, "Quasi-recurrent neural networks," 2016, *arXiv:1611.01576*.

[33] D. L. Minh, A. Sadeghi-Niaraki, H. D. Huy, K. Min, and H. Moon, "Deep learning approach for short-term stock trends prediction based on two-stream gated recurrent unit network," *IEEE Access*, vol. 6, pp. 55392–55404, 2018.

[34] S. Tanwar, N. P. Patel, S. N. Patel, J. R. Patel, G. Sharma, and I. E. Davidson, "Deep learning-based cryptocurrency price prediction scheme with inter-dependent relations," *IEEE Access*, vol. 9, pp. 138633–138646, 2021.

[35] I. E. Livieris, E. Pintelas, and P. Pintelas, "A CNN–LSTM model for gold price time-series forecasting," *Neural Comput. Appl.*, vol. 32, no. 23, pp. 17351–17360, Apr. 2020.

[36] K. Li, J. Daniels, C. Liu, P. Herrero, and P. Georgiou, "Convolutional recurrent neural networks for glucose prediction," *IEEE J. Biomed. Health Informat.*, vol. 24, no. 2, pp. 603–613, Feb. 2020.

[37] S. Du, T. Li, Y. Yang, and S.-J. Horng, "Deep air quality forecasting using hybrid deep learning framework," *IEEE Trans. Knowl. Data Eng.*, vol. 33, no. 6, pp. 2412–2424, Jun. 2021.

[38] E. Sharma, R. C. Deo, R. Prasad, A. V. Parisi, and N. Raj, "Deep air quality forecasts: Suspended particulate matter modeling with convolutional neural and long short-term memory networks," *IEEE Access*, vol. 8, pp. 209503–209516, 2020.

[39] P.-W. Chiang and S.-J. Horng, "Hybrid time-series framework for daily-based $PM_{2.5}$ forecasting," *IEEE Access*, vol. 9, pp. 104162–104176, 2021.

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

[41] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[42] L.-Q. Pan, J.-Z. Li, and J.-Z. Luo, "A temporal and spatial correlation based missing values imputation algorithm in wireless sensor networks," *Chin. J. Comput.*, vol. 33, no. 1, pp. 1–11, Apr. 2010.

[43] P. Liu, J. Wang, A. Sangaiah, Y. Xie, and X. Yin, "Analysis and prediction of water quality using LSTM deep neural networks in IoT environment," *Sustainability*, vol. 11, no. 7, p. 2058, Apr. 2019.

[44] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[45] Y. Yu, X. Si, C. Hu, and Z. Jianxun, "A review of recurrent neural networks: LSTM cells and network architectures," *Neural Comput.*, vol. 31, no. 7, pp. 1235–1270, 2019.
[46] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder–decoder for statistical machine translation," 2014, *arXiv:1406.1078*.
[47] T. Zhang, S. Song, S. Li, L. Ma, S. Pan, and L. Han, "Research on gas concentration prediction models based on LSTM multidimensional time series," *Energies*, vol. 12, no. 1, p. 161, Jan. 2019.
[48] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, and V. Dubourg, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Oct. 2011.
[49] R. Barzegar, M. T. Aalami, and J. Adamowski, "Short-term water quality variable prediction using a hybrid CNN–LSTM deep learning model," *Stochastic Environ. Res. Risk Assessment*, vol. 34, no. 2, pp. 415–433, 2020.

**K. P. RASHEED ABDUL HAQ** (Student Member, IEEE) received the bachelor's degree in electronics and communication engineering from the School of Engineering, Cochin University of Science and Technology (CUSAT), and the master's degree in electronics design technology from the National Institute of Electronics and Information Technology (NIELIT), Calicut, India. He is currently pursuing the Ph.D. degree with the Department of Electronics and Communication Engineering, National Institute of Technology Puducherry, Karaikal, India. His research interests include aquaculture, deep learning, deep sequence modeling, machine learning, and time-series prediction.

**V. P. HARIGOVINDAN** (Senior Member, IEEE) received the Bachelor of Technology degree in electronics and communication engineering from the University of Calicut, the Master of Technology degree (Hons.) in digital electronics and communication systems from Visvesvaraya Technological University, and the Ph.D. degree from the National Institute of Technology Calicut, in 2013. He is currently the Head of the Department and an Associate Professor with the Department of Electronics and Communication Engineering, National Institute of Technology Puducherry (under the Ministry of Education, Government of India). He has more than 40 international publications to his credit. He serves as the Principal Investigator for three funded projects, each from the Department of Science and Technology, the Government of India, the Science and Engineering Research Board, and Microsoft AI for Earth. His research interests include wireless networks and wireless communications.