sustainability

MDPI

Article

Assessing the Performance of Deep Learning Algorithms for Short-Term Surface Water Quality Prediction

Heelak Choi [1], Sang-Ik Suh [2], Su-Hee Kim [2], Eun Jin Han [1] and Seo Jin Ki [2,*]

[1] Water Quality Assessment Research Division, Water Environment Research Department, National Institute of Environmental Research, Incheon 22689, Korea; hlchoi@korea.kr (H.C.); gene0921@korea.kr (E.J.H.)

[2] Department of Environmental Engineering, Gyeongsang National University, Jinju 52725, Korea; sangik.suh@gnu.ac.kr (S.-I.S.); kimsuhee99@naver.com (S.-H.K.)

* Correspondence: seojinki@gnu.ac.kr; Tel.: +82-55-772-3341

Abstract: This study aimed to investigate the applicability of deep learning algorithms to (monthly) surface water quality forecasting. A comparison was made between the performance of an autoregressive integrated moving average (ARIMA) model and four deep learning models. All prediction algorithms, except for the ARIMA model working on a single variable, were tested with univariate inputs consisting of one of two dependent variables as well as multivariate inputs containing both dependent and independent variables. We found that deep learning models (6.31–18.78%, in terms of the mean absolute percentage error) showed better performance than the ARIMA model (27.32–404.54%) in univariate data sets, regardless of dependent variables. However, the accuracy of prediction was not improved for all dependent variables in the presence of other associated water quality variables. In addition, changes in the number of input variables, sliding window size (i.e., input and output time steps), and relevant variables (e.g., meteorological and discharge parameters) resulted in wide variation of the predictive accuracy of deep learning models, reaching as high as 377.97%. Therefore, a refined search identifying the optimal values on such influencing factors is recommended to achieve the best performance of any deep learning model in given multivariate data sets.

Keywords: deep learning; ARIMA; surface water quality; univariate data set; multivariate data set

check for updates

Citation: Choi, H.; Suh, S.-I.; Kim, S.-H.; Han, E.J.; Ki, S.J. Assessing the Performance of Deep Learning Algorithms for Short-Term Surface Water Quality Prediction. *Sustainability* **2021**, *13*, 10690. https://doi.org/10.3390/su131910690

Academic Editor: Ozgur Kisi

Received: 19 August 2021
Accepted: 23 September 2021
Published: 26 September 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

1. Introduction

Interest in deep learning for predictive modeling is growing from scientific community in the fields of hydrology and water resources [1–3]. This is particularly true for those who take advantage of better performance from deep learning than its traditional counterparts such as machine learning and statistical models [1,4,5]. The research applied to hydrologic and water quality (time series) data ranged from flood and run-off forecasting through water quality and quantity modeling to modern chemical process, fisheries, and aquacultural engineering, just to name a few [6–10]. Despite its potential advantages, the performance of deep learning was found to be highly sensitive to the number, size, and type of layers, and to a less obvious extent, loss functions, optimization procedures, and so on [11,12]. Yet, there is less consensus about the design and configuration of more effective deep learning models for data compiled at different spatial and temporal resolutions from various water monitoring programs.

Recent evidence suggests that a hybrid deep learning model combining more than two algorithms (in series) outperforms any standalone model which is eligible to time series prediction [9,13,14]. Barzegar et al. [8] found that the model merging a convolutional neural network (CNN) and a long short-term memory (LSTM) was superior to single-task learning approaches in predicting short-term water quality variables at a particular lake, for example. The study of Yan et al. [15] showed that the predictive model based on three algorithms accurately described the cross-sectional water quality profiles, compared to single and multiple ones with two algorithms. Sha et al. [13] also reported substantial performance improvement of the proposed model integrating the hybrid CNN-LSTM and decomposition methods over existing ones when forecasting periodic and non-periodic water quality parameters observed in real time. It should be noted, however, that there are still many studies showing the effectiveness of a single deep learning model as well as the model implementing deep learning and other traditional approaches for specific prediction tasks of hydrologic and water quality variables [12,16–20].

In parallel, the usefulness of machine learning models in predicting hydrologic and water quality parameters, including other associated variables, was also studied widely in the latest research [21–26]. This is because the predictive methods such as gene expression programming, model tree, and evolutionary polynomial regression adopted in those studies are particularly useful for developing explicit formulas which provide physical insights into hydrologic and water quality processes unlike other typical black-box algorithms (i.e., common machine learning and deep learning models) [21,23,24]. In those studies, the performance of the proposed approaches, including other advanced variants connecting one machine learning model to another, was evaluated with various statistical measures (e.g., the correlation coefficient, scatter index, and bias), in addition to uncertainty, reliability, and resilience analyses [21–26]. There were also continuous attempts of reducing the number of input variables manually (by eliminating one variable at a time) or automatically (with the help of various statistical techniques such as the principal component analysis and improved grey relational analysis), regardless of the types of data-driven models [23,24,27]. Considerable efforts have been still devoted not only to compare the predictive accuracy between machine learning and deep learning algorithms, but also to improve their learning process [28]. However, there is an urgent need to establish universal scientific methods in terms of statistical measures, variable selection, and some additional analyses which assess the robustness and reliability of sequential and non-sequential (data-driven) models.

In the absence of strict guideline for selecting the best candidate (deep learning approach) among them, this study was motivated to address how input data (settings) affected the accuracy of four deep learning models used for short-term surface water quality prediction. More specifically, this study adopted three standalone and one hybrid deep learning algorithms specialized for time series prediction: to investigate their performance against (1) univariate data sets consisting of single dependent variable (i.e., one of two target parameters), as well as (2) multivariate data sets constructed with both one dependent and nine independent variables and (3) to identify other influencing factors (i.e., the number of (important) input variables, sliding window size, and relevant variables) leading to variation in their performance. We believe that the proposed methodology not only helps develop a strategic plan for short-term change in water quantity and quality in a timely manner, but also encourages the understanding of the complex dynamics of natural systems such as water resources and wetlands.

2. Materials and Methods

2.1. Monitoring Stations and Data Collection

We selected four water quality monitoring sites, Paldang (PD), Mulgeum (MG) Daecheong (DC), and Juam (JA), to assess the performance of four different types of deep learning models (Figure 1). Note that those monitoring stations are known as representative and major stations which provide a broad overview of water quality status at four major rivers (i.e., the Han, Nakdong, Geum, and Yeongsan/Seomjin Rivers) in Korea. In addition, the stations PD, DC, and JA were located at dams which regulated the tail water flow and elevation along the river networks, whereas the other was selected from the downstream channel to examine the difference in their prediction performance between stagnant and running waters.

In these monitoring stations, water quality data were compiled on a monthly basis from January 2009 to December 2018 through the Water Environment Information System which was maintained by the National Institute of Environmental Research, Korea. Out of a total of 43 water quality parameters observed, we used only 10 water quality variables with relatively few missing values for the given period (Table 1). These included water temperature (°C), pH (-), dissolved oxygen (mg/L), biochemical oxygen demand (BOD, mg/L), chemical oxygen demand (mg/L), suspended solids (mg/L), electrical conductivity (µS/cm), total nitrogen (mg/L), total phosphorus (T-P, mg/L), and total coliforms (cfu/100 mL). Discharge at the closest water quality monitoring stations was taken in the same period from the Water Management Information System operated in the Han River Flood Control Office, Korea. We also collected other relevant record, meteorological data, for the corresponding period, which were available publicly at Open MET Data Portal in the Korea Meteorological Administration. Note that meteorological data adjacent to each water quality monitoring station are aggregated by month using the average (operation) for air temperature, relative humidity, and wind speed as well as using the sum for precipitation and solar radiation. Similarly, the sum aggregate function was used to transform flow rate data from daily to monthly time resolution. Any missing value in discharge was also replaced by imputed values using a linear interpolation approach.

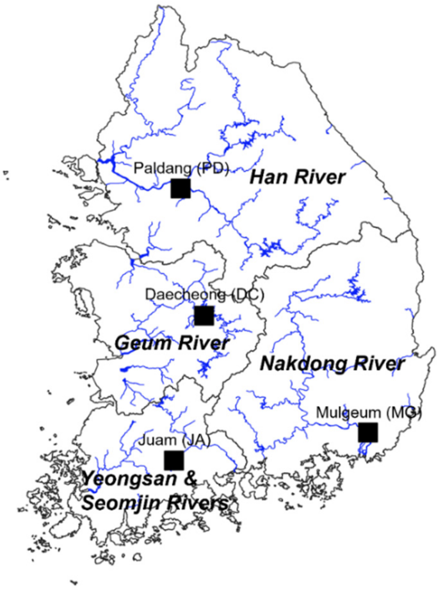

Figure 1. Water quality monitoring locations at four major rivers in Korea applied to time series prediction models.

Table 1. Descriptive statistics of water quality parameters monitored (monthly) at four different monitoring stations (PD, MG, DC, and JA) during the period of 2009 to 2018 (*n* = the number of data and CV = the coefficient of variation).

Variables	Units	*n*	PD		MG		DC		JA	
			Mean	CV	Mean	CV	Mean	CV	Mean	CV
Water temperature	°C	120	13.08	0.60	16.64	0.51	15.09	0.51	11.81	0.40
pH	–	120	7.77	0.10	8.03	0.05	7.85	0.05	6.93	0.05
Dissolved oxygen	mg/L	120	10.41	0.24	10.63	0.23	9.75	0.27	7.57	0.37
Biochemical oxygen demand	mg/L	120	1.16	0.31	2.17	0.36	0.95	0.25	0.84	0.24
Chemical oxygen demand	mg/L	120	3.69	0.16	6.33	0.19	4.12	0.17	2.97	0.12
Suspended solids	mg/L	120	6.42	1.20	16.35	1.12	2.70	0.56	1.99	0.54
Electrical conductivity	µS/cm	120	168.93	0.23	314.37	0.33	146.05	0.13	74.81	0.12
Total nitrogen	mg/L	120	2.08	0.18	2.81	0.24	1.45	0.23	0.75	0.14
Total phosphorus	mg/L	120	0.03	0.59	0.07	0.60	0.02	0.56	0.01	0.40
Total coliforms	cfu/100 mL	120	706.90	2.85	17,694	9.25	28.56	2.03	76.18	3.30

2.2. Input Data Preparation

Using all of the data (sources) listed above, various data sets were constructed to test the predictive accuracy of deep learning models. Firstly, we prepared two univariate data sets which included only one target variable. This was conducted because the baseline forecasting method, an autoregressive integrated moving average (ARIMA) model, to be compared to the adopted deep learning models, only accepted a single time series (see Section 2.3). For each performance test of these univariate data sets, either BOD or T-P was selected as the (target) dependent variable. Next, two multivariate data sets consisting of one of two target parameters (e.g., BOD or T-P) as well as the remaining 9 parameters out of 10 observed variables were built not only to examine the performance variation in deep learning models depending on dependent variables, but also to compare their accuracy to that of univariate data sets. Finally, several factors which were capable of affecting model performance were also evaluated by creating three different multivariate data sets. Those data sets were, in particular, developed by increasing the number of important independent variables (from 3 to 9), adjusting sliding window size in multiple input multi-step output (deep learning) models (from 9 through 12 to 15 months for multiple input and from 1 through 2 to 3 months for multi-step output), and incorporating additional variables such as discharge and meteorological data in the given multivariate data sets. Note that square root and log (with a base of 10) transformation are applied to BOD and T-P variables in univariate data sets, respectively, whereas the standardization method (namely, Z-score normalization) is used to make all independent variables, except for dependent variable, on the same scale in multivariate data sets. However, the ARIMA model was fitted to the raw time series data. All decisions of adopting different data preprocessing processes were made by trial and error to maximize the performance of all predictive models tested.

2.3. Applied Prediction Algorithms

We performed benchmark tests on different data sets using various (time series) prediction algorithms (Figure 2). In a series of tests, while the ARIMA played a role as the baseline model, three standalone and one hybrid deep learning algorithms were adopted for performance comparison with the reference ARIMA model. The prediction accuracy for all prediction algorithms is assessed in terms of the mean absolute percentage error (MAPE, in unit of %), which is one of the most common performance measures in time series forecasting. The deep learning algorithms we used were recurrent neural network (RNN), gated recurrent unit (GRU), LSTM, and those combined with CNN and GRU, all of which were widely applied to modern time series data. The chosen architecture of all applied deep learning models, except for the LSTM algorithm which included 2 layers of LSTM cells (namely, a stacked LSTM) as well as the hybrid algorithm which consisted of (1D) convolutional layer, max pooling layer, and GRU layer in series, had a single hidden layer. Note that we only apply a dropout rate of 0.2 and a recurrent dropout rate of 0.2 to the hidden layer in a single GRU algorithm. Moreover, while RNN, GRU and the initial layer of LSTM cells adopted the hyperbolic tangent (namely, tanh) function as non-linear activation, the rectified linear unit (namely, ReLU) function was used in the second layer of LSTM cells as well as convolutional layer in the hybrid algorithm. The root mean squared propagation (namely, RMSprop) was used to improve training speed and performance of all applied deep learning algorithms, where the learning rate and rho were set to 0.001 and 0.9, respectively. The total number of trainable model parameters were 708 for RNN, 415 for GRU, 29,761 for stacked LSTM, and 8,705 for hybrid one. More detailed information on those implemented models such as textual summary and graph plot is documented in the final project report [29].

The use of a free statistical software R (Ver. 4.0.4, The R Foundation, Vienna, Austria) as well as RStudio (Ver. 1.3.1073, RStudio, PBC., Boston, MA, USA) allowed us to evaluate the performance of all prediction algorithms, including the ARIMA model. More specifically, we used the forecast package (Ver. 8.13) in R to automatically search the best ARIMA model for the given univariate time series (from the auto.arima function). Different types of deep learning algorithms were also developed and assessed in R using the keras package (Ver. 2.4.0), a high-level deep learning library developed originally for Python, regardless of univariate and multivariate time series. Note that manipulation of time series data is conducted with the zoo package (Ver. 1.8-8). All developed deep learning algorithms were applied on partitioned data sets consisting of 70% of the data for training (n = 84) and the remaining 30% for testing (n = 36). The partitioned data sets were divided again into several equal segments (i.e., multiple samples) using an overlapping sliding (or moving) window with 1 month sliding interval to maximize the amount of data provided to deep learning algorithms. In other words, the windows where 9 and 3 (monthly) time steps were adopted as input and output (in multiple input multi-step output models), respectively, slid by 1 month. In this case, the following window overlapped with the preceding window by 8 (for input) and 2 months (for output). Those deep learning algorithms were trained for 100 epochs with a batch size of 12.

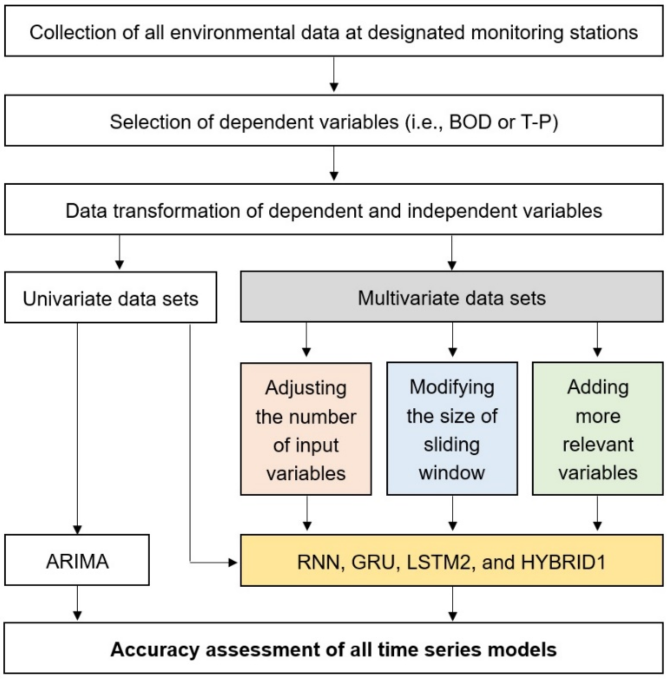
Collection of all environmental data at designated monitoring stations
Selection of dependent variables (i.e., BOD or T-P)
Data transformation of dependent and independent variables
Univariate data sets
Multivariate data sets
Adjusting the number of input variables
Modifying the size of sliding window
Adding more relevant variables
ARIMA
RNN, GRU, LSTM2, and HYBRID1
Accuracy assessment of all time series models

Figure 2. Schematic diagram illustrating a series of steps used to evaluate all applied time series models, including (input and output) data preparation.

2.4. Variable Selection

Removing irrelevant variables from the data sets assists in not only reducing the execution time of predictive models, but also enhancing their performance. For this study, the selection of important variables (namely, feature selection) was conducted easily with the help of the scikit-learn package, a popular library for machine learning in Python. The criterion for ranking all candidate independent variables we employed was the Pearson correlation coefficient (provided through the f_regression function). Note that the reticulate package (Ver. 1.20) in R enables us to use and run any Python code such as modules, classes, and functions immediately in a R environment. During the tests, the variables to be included in the model increased progressively from 3 to 9 (based on the variable importance determined from the Pearson correlation coefficient).

3. Results and Discussion

3.1. Performance Assessment on Univariate Data Sets

Table 2 presents the performance of all prediction algorithms for two dependent variables (i.e., BOD and TP) at four different monitoring locations (i.e., PD, MG, DC, and JA) using testing data in univariate data sets, in terms of MAPE (%). Note that the accuracy of prediction algorithms for testing data is almost equivalent to or lower than that of training data (not shown here for simplicity). Furthermore, the lower the MAPE is, the better the accuracy of prediction algorithm is. It was found from the table that the ARIMA model recorded higher error rates than four deep learning models (i.e., RNN, GRU, LSTM2, and HYBRID1), regardless of dependent variables as well as monitoring sites. In fact, the ARIMA model results achieved error rates as low as 27.32% for BOD and 27.61% for T-P. In contrast, deep learning models yielded MAPE values in the range of 6.51–18.78% for BOD and 7.98–18.66% for T-P. In addition, the predictive accuracy of the ARIMA model varied widely from station to station as well as from variable to variable. This inconsistent performance of the ARIMA model was similar to the results observed for deep learning models. In summary, even though we do not specify a universal model which works best in any situation, deep learning models lead to better performance the traditional method ARIMA. At this moment, we cannot clearly explain why both ARIMA and deep learning models show heterogeneous (prediction) performance according to stations and dependent variables.

Table 2. The predictive accuracy of five prediction algorithms (i.e., ARIMA plus four deep learning models) for two targe variables (i.e., BOD and T-P) at four different monitoring stations (PD, MG, DC, and JA) in univariate data sets in terms of MAPE (%).

Prediction Algorithms	**BOD**				**T-P**			
	PD	**MG**	**DC**	**JA**	**PD**	**MG**	**DC**	**JA**
ARIMA	109.64	404.54	27.32	43.97	27.61	69.14	40.70	36.16
RNN	7.91	18.78	6.51	10.90	13.06	8.82	11.32	8.08
GRU	9.50	18.54	8.84	10.37	17.47	9.63	18.26	7.98
LSTM2	7.46	15.60	11.14	10.59	13.54	9.96	13.91	9.65
HYBRID1	7.46	15.61	7.73	10.27	18.66	10.17	11.21	8.24

3.2. Performance Assessment on Multivariate Data Sets

The performance of all prediction algorithms was also assessed against testing data in multivariate data sets, with respect to MAPE (%) (Table 3). Please be aware that the predictive accuracy of four deep learning algorithms is compared to that of the ARIMA model obtained from single dependent variable. From the table, it was observed that deep learning models did not always show superior performance than the ARIMA model. More specifically, although deep learning models provided relatively accurate forecasting of BOD (time series) compared to that of the ARIMA model, their predictive accuracy for T-P significantly decreased depending on algorithms and stations. In fact, MAPE values were highest for HYBRID1 in PD (108.30%), RNN in MG (368.80%), and GRU in DC (121.80%) and JA (243.60%). Taken together, adding more independent (water quality) variables to deep learning models neither necessarily improves the accuracy of prediction nor maintains the performance steadily across stations.

Table 3. The predictive accuracy of five prediction algorithms (i.e., ARIMA plus four deep learning models) for two target variables (i.e., BOD and T-P) at four different monitoring stations (PD, MG, DC, and JA) in multivariate data sets in terms of MAPE (%).

Prediction Algorithms	**BOD**				**T-P**			
	PD	**MG**	**DC**	**JA**	**PD**	**MG**	**DC**	**JA**
ARIMA	109.64	404.54	27.32	43.97	27.61	69.14	40.70	36.16
RNN	40.40	48.91	27.10	27.69	57.68	368.80	40.76	22.22
GRU	36.06	29.03	25.20	26.16	96.20	183.00	121.80	243.60
LSTM2	39.08	32.91	24.54	25.02	54.84	71.91	54.33	42.26
HYBRID1	64.75	39.99	21.98	17.61	108.30	81.54	31.32	37.06

3.3. Influence of Other Factors on Performance

3.3.1. The Number of Input Variables

Figure 3a,b displays the variation in the performance of a particular deep learning model LSTM2 for BOD and T-P at four monitoring sties in response to the number of (input) variables, respectively. Note that independent variables are added sequentially to the model according to their importance provided by the proposed variable selection approach (see Section 2.4). As can be seen in the figures, MAPE values of the LSTM2 model changed significantly based on the number of input variables, regardless of target variables BOD and T-P. The accuracy of prediction was often improved by incorporating a few variables in the model at some stations (e.g., one more variable for BOD at PD and two more variables for T-P at PD), but its performance fluctuated remarkably among four stations. In some cases, the error rates were further reduced by incorporating the maximum number of variables (i.e., eight parameters), as compared to the minimum number of variables (i.e., three parameters). All these results reveal that determining the optimal number of input variables is a very complex task that inevitably requires an iterative process of searching for the minimal error for a given multivariate data set.

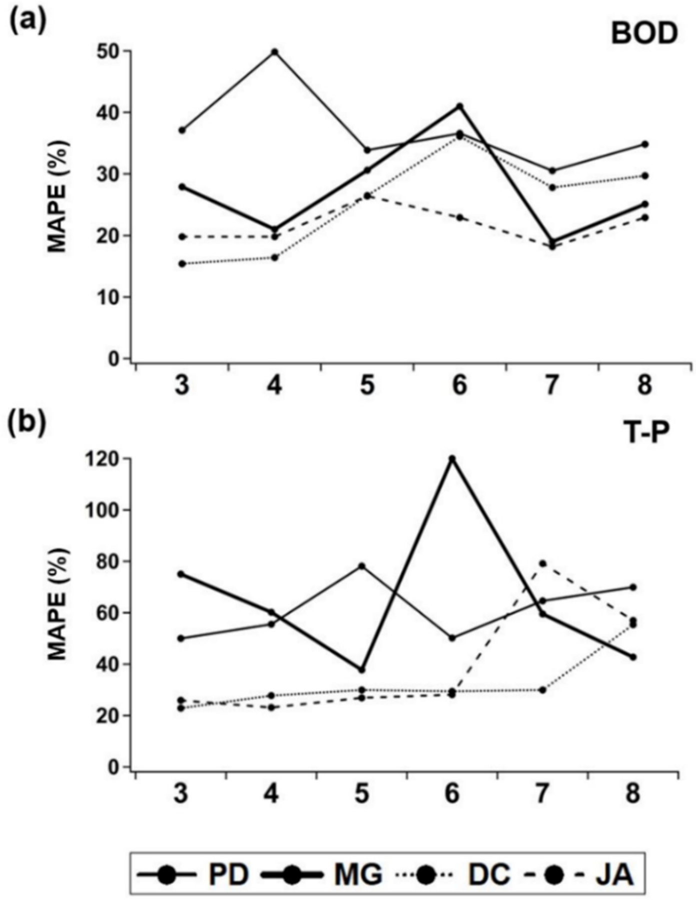

Figure 3. Changes in MAPE values of the LSTM2 model for (**a**) BOD and (**b**) T-P at four monitoring locations according to the number of input variables.

3.3.2. Sliding Window Size

The influence of sliding window size on the accuracy of prediction was also studied in the LSTM2 model at three stations (Table 4). In the table, MAPE values of the LSTM2 model at PD was specifically excluded due to extremely low performance ranging from 10^5 (ten to the power of five) to 10^7% (ten to the power of seven). It was determined from the table that either increasing (multiple) input steps from 9 through 12 to 15 months or decreasing multi-step output from 1 through 2 to 3 months did not simply result in the performance improvement of a particular deep learning model. In addition, the predictive accuracy did not increase gradually in accordance with dependent variables as well as stations. Consequently, altering time steps involved in input and output definitely causes the variation in the accuracy of prediction. However, the best performance of a given deep learning algorithm can be achieved through an iterative search for the optimal sliding window size at each station, as discussed in Section 3.3.1.

Table 4. The predictive accuracy of the LSTM2 algorithm for BOD and T-P based on different input (ranging from 9 through 12 to 15 months) and output steps (ranging from 1 through 2 to 3 months) in terms of MAPE (%).

Input and Output Steps	BOD			T-P		
	MG	DC	JA	MG	DC	JA
9 + 1	24.69	22.9	22.90	15.17	24.58	21.63
9 + 2	33.70	22.55	22.55	18.36	24.11	39.17
9 + 3	27.56	26.16	26.16	19.34	24.71	22.12
12 + 1	29.55	23.90	23.90	15.48	39.64	20.70
12 + 2	37.43	27.16	27.16	18.70	26.53	22.69
12 + 3	33.12	26.69	26.69	19.91	25.00	23.33
15 + 1	26.21	24.50	24.50	14.99	24.41	24.29
15 + 2	32.91	24.33	24.33	21.18	25.50	24.58
15 + 3	34.81	25.14	25.14	19.27	26.85	24.57

3.3.3. Relevant Variables

Figure 4a,b illustrates changes in the predictive accuracy of four deep learning models for BOD and T-P at one particular monitoring site DC when more independent variables associated with water quality are added. The relevant variables included additionally in the model were discharge and five different meteorological variables (see Section 2.1). In the figures, individual parameters belonged to the upper category are labeled as W for water quality variables, M for meteorological variables, and D for discharge variable only. The error bars indicate the standard deviation of MAPE values obtained from four deep learning models. It was confirmed from the figure that incorporating both discharge and meteorological variables into deep learning models helped elevate the predictive accuracy of BOD, whereas the reverse is true for T-P. Out of all possible combinations of variables examined, the contribution of discharge variable to reduction in error rates was relatively larger than those of meteorological variables only as well as meteorological plus discharge variables, regardless of the dependent variables.

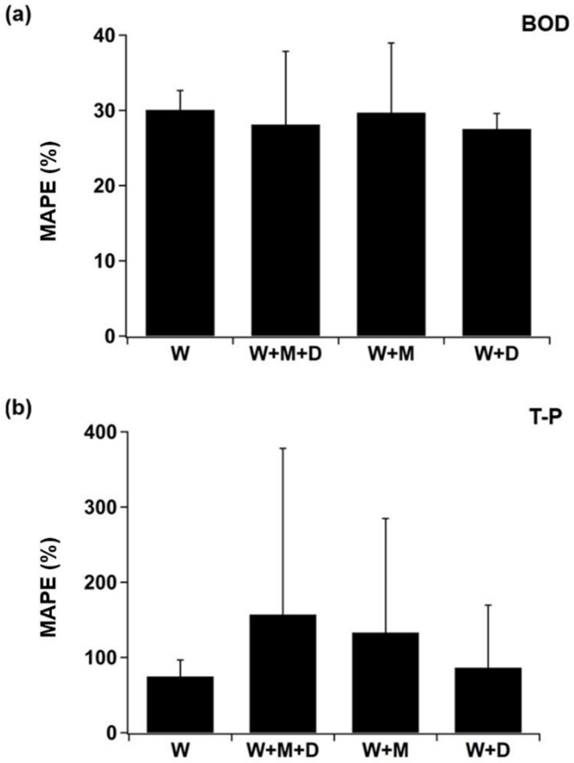

Figure 4. Changes in MAPE values of four deep learning models for (**a**) BOD and (**b**) T-P at one monitoring location DC in response to different multivariate data sets containing water quality variables only (indicated as W), water quality, meteorological, and discharge variables (indicated as W + M + D), water quality and meteorological variables (indicated as W + M), and water quality and discharge variables (indicated as W + D).

4. Conclusions

The intention of this study was to assess the predictive ability of deep learning algorithms for surface water quality in the short term. We constructed and employed three individual and one hybrid algorithms, which were widely adopted for time series prediction, to compare their performance to that of the traditional approach, ARIMA. By providing the modified data sets to all prediction models, the following conclusions were made.

- All deep learning algorithms applied to univariate data sets achieved more reliable forecasts than the ARIMA model whatever the dependent variables BOD and T-P. However, the performance of all prediction models, including ARIMA, was heavily dependent on monitoring stations.
- Using multivariate data sets, we observed noticeable improvement in the predictive accuracy of deep learning models for BOD rather than for T-P (in contrast to that of the ARIMA model derived from each dependent variable). This implied that additional water quality variables did not always enhance the accuracy of prediction for all target variables.
- The number of input variables and sliding window size (input and output steps in the models) were responsible for changes in the performance of deep learning models. The highest prediction accuracy of deep learning models was achieved with the addition of discharge variable (to existing multivariate data sets), instead of using other data sets merging water quality and relevant parameters such as meteorological variables or both meteorological and discharge variables. In our case, this assumption is, however, only valid for prediction of BOD (time series).
- As a preliminary study, this study did not examine the effectiveness of other advanced variants such as encoder-decoder model and attention mechanism, which evolved

from traditional deep learning approaches proposed for time series forecasting. More research is, therefore, needed to verify the superiority of those single algorithms, in addition to ensemble learning which combine predictions from multiple (deep learning) models to improve its prediction accuracy over a standalone model. Moreover, as the performance of deep learning algorithms was noticeably affected by the amount of data, model architectures, and dependent variables, these issues should be carefully addressed when developing short-term surface water quality prediction models, specifically using data sets updated monthly or weekly.

Author Contributions: Conceptualization, S.-I.S. and E.J.H.; methodology, S.-I.S. and E.J.H.; formal analysis, S.-I.S. and S.-H.K.; writing—original draft preparation, H.C. and S.J.K.; writing—review and editing, H.C. and S.J.K.; supervision, S.J.K. All authors have read and agreed to the published version of the manuscript.

Funding: This study was supported by a grant (NIER-2020-04-02-129) from the National Institute of Environmental Research (NIER), which was funded by the Ministry of Environment (MOE) of the Republic of Korea.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Than, N.H.; Ly, C.D.; Van Tat, P. The Performance of Classification and Forecasting Dong Nai River Water Quality for Sustainable Water Resources Management Using Neural Network Techniques. *J. Hydrol.* **2021**, *596*, 126099. [CrossRef]
2. Liu, P.; Wang, J.; Sangaiah, A.K.; Xie, Y.; Yin, X. Analysis and Prediction of Water Quality Using LSTM Deep Neural Networks in IoT Environment. *Sustainability* **2019**, *11*, 2058. [CrossRef]
3. Zhi, W.; Feng, D.; Tsai, W.P.; Sterle, G.; Harpold, A.; Shen, C.; Li, L. From Hydrometeorology to River Water Quality: Can a Deep Learning Model Predict Dissolved Oxygen at the Continental Scale? *Environ. Sci. Technol.* **2021**, *55*, 2357–2368. [CrossRef] [PubMed]
4. Lee, S.; Lee, D. Improved Prediction of Harmful Algal Blooms in Four Major South Korea's Rivers Using Deep Learning Models. *Int. J. Environ. Res. Public Health* **2018**, *15*, 1322. [CrossRef] [PubMed]
5. Peterson, K.T.; Sagan, V.; Sloan, J.J. Deep Learning-Based Water Quality Estimation and Anomaly Detection Using Landsat-8/Sentinel-2 Virtual Constellation and Cloud Computing. *GIScience Remote Sens.* **2020**, *57*, 510–525. [CrossRef]
6. Li, Z.; Ling, K.; Zhou, L.; Zhu, M. Deep Learning Framework with Time Series Analysis Methods for Runoff Prediction. *Water* **2021**, *13*, 575.
7. Udayakumar, K.; Subiramaniyam, N.P. Deep Learning-Based Production Assists Water Quality Warning System for Reverse Osmosis Plants. *H2Open J.* **2020**, *3*, 538–553.
8. Barzegar, R.; Aalami, M.T.; Adamowski, J. Short-Term Water Quality Variable Prediction Using a Hybrid CNN–LSTM Deep Learning Model. *Stoch. Environ. Res. Risk Assess.* **2020**, *34*, 415–433. [CrossRef]
9. Yan, J.; Liu, J.; Yu, Y.; Xu, H. Water Quality Prediction in the Luan River Based on 1-Drcnn and Bigru Hybrid Neural Network Model. *Water* **2021**, *13*, 1273.
10. Thai-Nghe, N.; Thanh-Hai, N.; Ngon, N.C. Deep Learning Approach for Forecasting Water Quality in IoT Systems. *Int. J. Adv. Comput. Sci. Appl.* **2020**, *11*, 686–693.
11. Wang, X.; Zhang, C. Water Quality Prediction of San Francisco Bay Based on Deep Learning. *J. Jilin Univ. (Earth Sci. Ed.)* **2021**, *51*, 222–230.
12. Chandra, R.; Goyal, S.; Gupta, R. Evaluation of Deep Learning Models for Multi-Step Ahead Time Series Prediction. *IEEE Access* **2021**, *9*, 83105–83123.
13. Sha, J.; Li, X.; Zhang, M.; Wang, Z.L. Comparison of Forecasting Models for Real-time Monitoring of Water Quality Parameters Based on Hybrid Deep Learning Neural Networks. *Water* **2021**, *13*, 1547. [CrossRef]
14. Baek, S.S.; Pyo, J.; Chun, J.A. Prediction of Water Level and Water Quality Using a CNN-LSTM Combined Deep Learning Approach. *Water* **2020**, *12*, 3399. [CrossRef]
15. Yan, J.; Gao, Y.; Yu, Y.; Xu, H.; Xu, Z. A Prediction Model Based on Deep Belief Network and Least Squares SVR Applied to Cross-Section Water Quality. *Water* **2020**, *12*, 1929.
16. Hu, Z.; Zhang, Y.; Zhao, Y.; Xie, M.; Zhong, J.; Tu, Z.; Liu, J. A water quality prediction method based on the deep LSTM network considering correlation in smart mariculture. *Sensors* **2019**, *19*, 1420. [CrossRef]
17. Faruk, D.Ö. A Hybrid Neural Network and ARIMA Model for Water Quality Time Series Prediction. *Eng. Appl. Artif. Intell.* **2010**, *23*, 586–594.
18. Li, Z.; Peng, F.; Niu, B.; Li, G.; Wu, J.; Miao, Z. Water quality prediction model combining sparse auto-encoder and LSTM network. *IFAC-Pap.* **2018**, *51*, 831–836. [CrossRef]

19. Zhou, Y. Real-Time Probabilistic Forecasting of River Water Quality under Data Missing Situation: Deep Learning plus Post-Processing Techniques. *J. Hydrol.* **2020**, *589*, 125164.
20. Loc, H.H.; Do, Q.H.; Cokro, A.A.; Irvine, K.N. Deep Neural Network Analyses of Water Quality Time Series Associated with Water Sensitive Urban Design (WSUD) Features. *J. Appl. Water Eng. Res.* **2020**, *8*, 313–332.
21. Najafzadeh, M. Evaluation of conjugate depths of hydraulic jump in circular pipes using evolutionary computing. *Soft Comput.* **2019**, *23*, 13375–13391. [CrossRef]
22. Saberi-Movahed, F.; Najafzadeh, M.; Mehrpooya, A. Receiving more accurate predictions for longitudinal dispersion coefficients in water pipelines: Training group method of data handling using extreme learning machine conceptions. *Water Resour. Manag.* **2020**, *34*, 529–561. [CrossRef]
23. Najafzadeh, M.; Ghaemi, A.; Emamgholizadeh, S. Prediction of water quality parameters using evolutionary computing-based formulations. *Int. J. Environ. Sci. Technol.* **2019**, *16*, 6377–6396. [CrossRef]
24. Najafzadeh, M.; Homaei, F.; Farhadi, H. Reliability assessment of water quality index based on guidelines of national sanitation foundation in natural streams: Integration of remote sensing and data-driven models. *Artif. Intell. Rev.* **2021**, *54*, 4619–4651. [CrossRef]
25. Heddam, S.; Kisi, O. Modelling daily dissolved oxygen concentration using least square support vector machine, multivariate adaptive regression splines and M5 model tree. *J. Hydrol.* **2018**, *559*, 499–509. [CrossRef]
26. Najafzadeh, M.; Niazmardi, S. A Novel Multiple-Kernel Support Vector Regression Algorithm for Estimation of Water Quality Parameters. *Nat. Resour. Res.* **2021**, *30*, 3761–3775. [CrossRef]
27. Zhou, J.; Wang, Y.; Xiao, F.; Wang, Y.; Sun, L. Water quality prediction method based on IGRA and LSTM. *Water* **2018**, *10*, 1148. [CrossRef]
28. Khullar, S.; Singh, N. Water quality assessment of a river using deep learning Bi-LSTM methodology: Forecasting and validation. *Environ. Sci. Pollut. Res.* **2021**. [CrossRef]
29. Korea Water Resources Association. *A Study on Water Quality Assessment with Data-Driven Models and Its Short-Term Prediction Methods*; National Institute of Environmental Research: Incheon, Korea, 2021.