International Journal of
Environmental Research and Public Health

MDPI

Article

Water Quality Prediction Based on Multi-Task Learning

Huan Wu [1,2], Shuiping Cheng [1,*], Kunlun Xin [1], Nian Ma [2,3], Jie Chen [2,4], Liang Tao [2] and Min Gao [5]

[1] College of Environmental Science and Engineering, Tongji University, Shanghai 200092, China
[2] T.Y.Lin International Engineering Consulting (China) Co., Ltd., Chongqing 401121, China
[3] Faculty of Natural Sciences, University of the Western Cape, Cape Town 7535, South Africa
[4] College of Environment and Ecology, Chongqing University, Chongqing 400030, China
[5] School of Big Data and Software Engineering, Chongqing University, Chongqing 401331, China
* Correspondence: shpcheng@tongji.edu.cn

Abstract: Water pollution seriously endangers people's lives and restricts the sustainable development of the economy. Water quality prediction is essential for early warning and prevention of water pollution. However, the nonlinear characteristics of water quality data make it challenging to accurately predicted by traditional methods. Recently, the methods based on deep learning can better deal with nonlinear characteristics, which improves the prediction performance. Still, they rarely consider the relationship between multiple prediction indicators of water quality. The relationship between multiple indicators is crucial for the prediction because they can provide more associated auxiliary information. To this end, we propose a prediction method based on exploring the correlation of water quality multi-indicator prediction tasks in this paper. We explore four sharing structures for the multi-indicator prediction to train the deep neural network models for constructing the highly complex nonlinear characteristics of water quality data. Experiments on the datasets of more than 120 water quality monitoring sites in China show that the proposed models outperform the state-of-the-art baselines.

Keywords: multi-task learning; water quality prediction; multiple indicator prediction

Citation: Wu, H.; Cheng, S.; Xin, K.; Ma, N.; Chen, J.; Tao, L.; Gao, M. Water Quality Prediction Based on Multi-Task Learning. *Int. J. Environ. Res. Public Health* **2022**, *19*, 9699. https://doi.org/10.3390/ijerph19159699

Received: 30 June 2022
Accepted: 3 August 2022
Published: 6 August 2022

1. Introduction

The excessive exploitation and utilization of water resources have caused a series of problems, such as deterioration of water quality, damage to water functional areas, and degradation of river ecosystem structures, which seriously endanger the social and economic development and the safety of people. Water quality prediction is essential for water pollution prevention and treatment, which can help fully understand the dynamic trend of the surface water ecological environment and warn of possible pollution incidents.

However, it is difficult to predict water quality because of the nonlinear characteristics of water-related data [1]. Traditional statistical analysis methods lack nonlinear approximation and self-learning abilities and cannot fully consider the complex impact of various environmental factors. With the rapid development of machine learning technology, scholars have begun to explore water quality prediction based on machine learning. They have achieved better water quality prediction performance by establishing nonlinear learning cognitive models from historical data, summarizing and discovering knowledge, and predicting system behavior. Olyaie et al. (2017) applied linear genetic programming and a support vector machine (SVM) to predict dissolved oxygen (DO) in the Delaware River in Trenton, USA [2]. Li et al. (2017) proposed a method that combines ensemble empirical mode decomposition (EEMD) [3–5] and least-squares SVR (support vector regression) to predict DO concentration [6]. Leong et al. (2021) applied SVM [7–9] and least squares support vector models to the Perak River in Malaysia [10]. The performance of these methods depends not only on the models but also on the features selected for training.

More and more researchers have recently applied deep learning methods to water quality prediction because deep learning (DL) [11,12] can efficiently train and abstract multi-level features of multi-dimensional training data. Banejad et al. (2011) applied a basic neural network to predict biochemical oxygen demand (BOD) and DO of the Morad River in Iran [13]. They verified that the deep learning technology can reliably, efficiently, and accurately extract the nonlinear characteristics of water quality data. Subsequently, Heddam et al. (2014, 2016) and Liu et al. (2020) successively proposed models based on GRNN (generalized regression neural network) and MLP (multilayer perceptron), which were applied to different rivers in the United States and lakes in China [14–16]. Zhou et al. (2019) proposed a deep cascade forest (DCF) that uses several random forests based on ensemble learning, performing well on large and even small-scale data [17]. Wang et al. (2019) proposed a hybrid CNN-LSTM (convolutional neural network- long short-term memory) deep learning algorithm for a dynamic chemical oxygen demand (COD) prediction model of urban sewage [18]. Zou et al. (2020) proposed a water quality prediction method based on the Bi-LSTM (Bidirectional LSTM) model with multiple time scales [19]. Niu et al. (2021) also developed a pixel-based deep neural network regression model and a patch-based deep neural network regression model, to estimate seven optically inactive water quality parameters [20]. Yang et al. (2021) proposed a mixed model named CNN-LSTM with Attention (CLA), combining CNN, LSTM, and Attention mechanisms to predict water quality [21]. Guo et al. (2022) use progressively decreasing deep neural network and multimodal deep learning (MDL) models without well-handled input features, to estimate long-term water indicators and explore the contribution of each feature by quantifying [22].

However, these models are constructed to optimize a single prediction indicator such as the potential of hydrogen (pH), dissolved oxygen (DO), chemical oxygen demand-Mn (COD_{Mn}), and Ammonia Nitrogen (NH_3-N, NHN for short), etc., which cannot guarantee the high efficiency and accuracy of the models in predicting other water quality indicators. The correlation between multiple prediction indicators can provide more correlation auxiliary information, which helps improve the prediction performance. To this end, we propose a water quality prediction model based on multi-task learning by learning the highly complex nonlinear characteristics of time series data and exploring the correlation of multi-indicator prediction.

The main contributions of this paper are as follows:

(1) We propose a multi-indicator prediction model of surface water quality based on deep learning, which excavates the highly complex nonlinear characteristics of surface water ecological environment water quality data and explores the correlation of multiple water quality prediction indicators.

(2) We propose four water quality prediction frameworks, named hard parameter sharing structure (Multi-Task-Hard), soft parameter sharing structure (Multi-Task-Soft), gated parameter sharing structure (Multi-Task-Gate), and gated hidden parameters sharing structure (Multi-Task-GH), based on different multi-task learning structures and combine the frameworks with various mainstream deep learning models to form different water quality prediction models.

(3) We conducted experiments to predict four water quality indicators, including pH, DO, COD_{Mn}, and NH_3-N, on real data from more than 120 water quality monitoring sites in seven river systems and lakes in China. The experimental results demonstrate that the proposed water quality multi-task learning prediction framework outperforms the state-of-the-art single-indicator prediction models.

2. Methodology

The existing deep learning-based water quality prediction models rarely consider the relationship between multiple indicators of water quality. The relationship between multiple indicators is crucial for the prediction because they can provide more associated auxiliary information. To this end, we propose a prediction method based on exploring the correlation of water quality multi-indicator prediction tasks in this section. We first define

the water quality prediction and explore four sharing structures for the multi-indictor prediction to train the deep neural network models for constructing the highly complex nonlinear characteristics of water quality data.

2.1. Definition of Water Quality Prediction

Following previous work [19,23,24], we choose four water quality indicators, including pH, DO, COD_{Mn}, and NH_3-N, as our prediction targets. Compared with other indicators, these indicators can predict that six water quality levels perform significantly better, reflecting the water quality better [23].

The water quality prediction is a time series prediction. We give the mathematical definitions for single-task prediction and multi-task prediction.

Single-task water quality prediction: $X \to Y$. Given water quality prediction indicators at known past times $(x_1, \ldots, x_i) \in X$, analyze the change patterns and predict the water quality indicator at the future time interval [11], denoted as $y_{i+1} \in Y$.

Multi-task water quality prediction: $(X_1, \ldots, X_N) \to (Y_1, \ldots, Y_N)$. Given N water quality prediction indicators of the past i times $\{(x_{11}, \ldots, x_{i1}), \ldots, (x_{1N}, \ldots, x_{iN})\} \in (X_1, \ldots, X_N)$, analyze the change patterns of N indicators at the same time, and predict multiple water quality indicators at the future time interval, denoted as $(y_1, \ldots, y_N) \in (Y_1, \ldots, Y_N)$.

For the four common water quality prediction indicators, pH, CODMn, DO, and NH3-N, the multi-task water quality prediction task can be defined as, given the numerical changes of pH, DO, COD_{Mn}, NH_3-N at the past i time intervals $\{(x_{pH1}, \ldots, x_{pHi}), (x_{DO1}, \ldots, x_{DOi}), (x_{COD1}, \ldots, x_{CODi}), (x_{NHN1}, \ldots, x_{NHNi})\} \in (X_{pH}, X_{CO}, X_{COD}, X_{NHN})$, analyze the change patterns and predict the corresponding water quality indicators at the future time interval, denoted as $(y_{pH}, y_{DO}, y_{COD}, y_{NHN}) \in (Y_{pH}, Y_{DO}, Y_{COD}, Y_{NHN})$.

2.2. Architecture of Water Quality Prediction Model Based on Multi-Task Learning

Frameworks for multi-task learning are often based on sharing the same bottom structure [25–27]. The model of multiple tasks can be transformed into a basic bottom model and multiple separate models. For single-task learning, the input and output of each task correspond to a separate model, and new models need to be built for new tasks, although the structure of the models is sometimes the same. For multi-task learning, the common structure of the model is unified into a basic model. Then, several separate models are introduced to realize the learning of multiple different tasks. Figure 1 is a basic framework for multi-task learning, in which the blue part represents the shared parameter layer, and the orange and yellow parts represent models for different tasks forming the tower layer. This framework structure saves the parameter space of multiple water quality prediction models and reduces the risk of over-fitting. We propose a multi-task learning framework for water quality prediction based on different structures [28,29]. The framework can be developed into four forms: hard parameter sharing structure (Multi-Task-Hard), soft parameter sharing structure (Multi-Task-Soft), gated parameter sharing structure (Multi-Task-Gate), and gated hidden parameters sharing structure (Multi-Task-GH). The differences between the four structures are described in detail in Section 2.3.

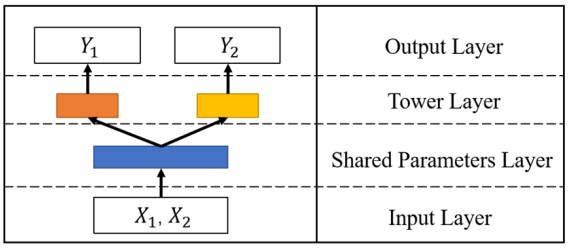

Figure 1. The basic framework of multi-task learning. The blue part represents the shared parameter layer, and the orange and yellow parts represent the models for different tasks forming the tower layer.

2.3. Multi-Task Learning Structures

2.3.1. Hard Parameter Sharing Structure of Multi-Indicator Water Quality Prediction (Multi-Task-Hard)

The hard parameter sharing structure is the basic structure of the shared bottom structure in multi-task learning. As shown in Figure 2, it is mainly divided into four parts.

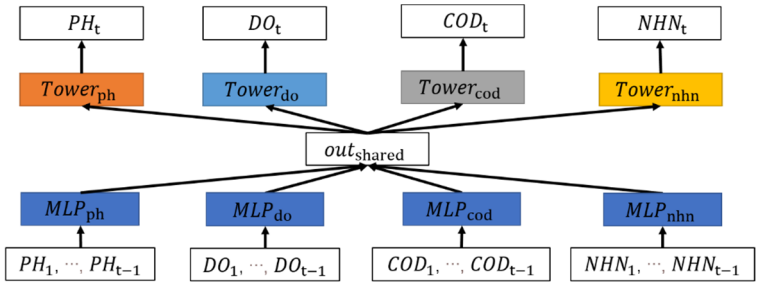

Figure 2. Hard parameter sharing structure of multi-indicator water quality prediction. The dark blue part represents the shared parameter layer, and the orange and yellow parts represent the models for different tasks forming the tower layer.

The first part is the input layer $(X_1, \ldots, X_N)$, which contains the time sequence information of each water quality indicator at the past time intervals.

The second part is the shared parameter layer, which is designed as a fully connected layer. This part takes the information transmitted by the input layer and extracts a shared implicit vector out_{shared}.

The third part is the tower layer, which is carefully designed for a task and will output the prediction results required by the corresponding task, which reflects the flexibility of the multi-task learning framework. The output of the second layer will be transmitted to the tower layer for different tasks simultaneously. Because of the differences between the indicators, it is necessary to design specific models for different water quality indicators in this layer. To put it simply, one task corresponds to one tower.

The fourth part is the output layer, which contains the outputs: $Y_{pH}, Y_{DO}, Y_{COD}, Y_{NHN}$ as the prediction.

The algorithm is shown in Algorithm 1, and the MLP is selected for processing in the shared layer.

Algorithm 1: Multi-indicator water quality prediction based on hard parameter sharing multi-task learning

Input: water quality prediction indicators at the past time intervals $(X_1, \ldots, X_N)$
1 : $out_{shared} \leftarrow \mathrm{MLP}([X_1, \ldots, X_N])$
2 : $(Y_1, \ldots, Y_N) \leftarrow \mathrm{Tower}_{1,\ldots,N}(out_{shared})$
Output: water quality indicators at the future time intervals $(Y_1, \ldots, Y_N)$

We introduce the specific structure of the hard parameter sharing structure with pH, DO, COD_{Mn}, and NH_3-N as the prediction target. As shown in Figure 2, all indicators from the input layer to the shared parameter layer have the same structure. Take the pH value part as an example. We input data $(pH_1, \ldots, pH_{t-1})$ in the input layer, which will be transmitted to the shared parameter layer and converted into the output vector by the fully connected neural network. Similarly, the inputs of DO, COD_{Mn}, and NH_3-N will also be converted to output vectors $out_{DO}, out_{COD}, out_{NHN}$ accordingly. The equation is as Equation (1). All input data will be dealt with by MLP and ReLU (rectified linear unit).

$$\begin{aligned} out_{pH} &= \mathrm{ReLU}(\mathrm{MLP}(pH_1, \ldots pH_{t-1})) \\ out_{DO} &= \mathrm{ReLU}(\mathrm{MLP}(DO_1, \ldots DO_{t-1})) \\ out_{COD} &= \mathrm{ReLU}(\mathrm{MLP}(COD\mathrm{Mn}_1, \ldots COD\mathrm{Mn}_{t-1})) \\ out_{NHN} &= \mathrm{ReLU}(\mathrm{MLP}(NHN_1, \ldots NHN_{t-1})) \end{aligned} \quad (1)$$

where ReLU is a nonlinear function used to add nonlinearity to the model. Compared with sigmoid, ReLU can effectively alleviate the problems of gradient disappearance and gradient explosion in deep neural networks. The formula of ReLU is shown in Equation (2):

$$\mathrm{ReLU}(x) = \begin{cases} x, & if\ x > 0 \\ 0, & if\ x \leq 0 \end{cases} \tag{2}$$

We use the multilayer perceptron (MLP) to extract the deeply hidden features of the water quality time series. MLP can simply and efficiently represent the global features of time series, which helps the subsequent tower layer extract the deep local features for different water quality indicators. The formulation of MLP is shown in Equations (3) and (4), where x denotes the input, W denotes the weight matrix w_i, b denotes the bias term, and y denotes the final output.

$$z = \sum_{i=1}^{n} w_i x_i + b \tag{3}$$

$$y = \mathrm{ReLU}(z) \tag{4}$$

The MLP model consists of three parts: input layer, hidden layer, and output layer. The number of hidden layers in the MLP can be adjusted as a hyperparameter. The number of neurons in the output layer is the number of the water quality prediction indicators. We train the MLP model with the BP (Back Propagation) algorithm, whose loss propagates back from the top layer to the bottom layer.

The last layer of the network is the output layer, and the loss function is defined as Equation (5), where L_n represents all neurons of the layer, $y_n^{(j)}$ represents the output of the j-th neuron, t denotes the predicted value corresponding to $(\hat{pH}_\mathrm{t}, \widehat{DO}_\mathrm{t}, \widehat{COD\mathrm{Mn}}_\mathrm{t}, \widehat{NHN}_\mathrm{t})$, and y denotes the real value corresponding to $(pH_\mathrm{t}, DO_\mathrm{t}, COD\mathrm{Mn}_\mathrm{t}, NHN_\mathrm{t})$.

$$Loss = \frac{1}{2} \sum_{j \in L_n} \left(t^{(j)} - y_n^{(j)}\right)^2 \tag{5}$$

The variables w and b are obtained by gradient descent to minimize the loss function, we show Equations (6)–(8) below to show the calculation of w and b's gradient:

$$\frac{\partial Loss}{\partial w_l^{(ji)}} = \frac{\partial Loss}{\partial y_l^{(j)}} \frac{\partial y_l^{(j)}}{\partial w_l^{(ji)}} = \frac{\partial Loss}{\partial y_l^{(j)}} \frac{\partial y_l^{(j)}}{\partial z_l^{(j)}} \frac{\partial z_l^{(j)}}{\partial w_l^{(ji)}} = \delta_l^{(j)} y_{l-1}^{(i)} \tag{6}$$

$$\frac{\partial Loss}{\partial b_l^{(j)}} = \frac{\partial Loss}{\partial y_l^{(j)}} \frac{\partial y_l^{(j)}}{\partial b_l^{(j)}} = \frac{\partial Loss}{\partial y_l^{(j)}} \frac{\partial y_l^{(j)}}{\partial z_l^{(j)}} \frac{\partial z_l^{(j)}}{\partial b_l^{(j)}} = \delta_l^{(j)} \tag{7}$$

$$\delta_l^{(j)} = \frac{\partial Loss}{\partial y_l^{(j)}} f'(z_l^{(j)}) = f'(z_l^{(j)}) \sum_{k \in L_{l+1}} \delta_{l+1}^k w_{l+1}^{(kj)} \tag{8}$$

$$W_l \leftarrow W_l - \eta \frac{\partial Loss}{\partial W_l} = W_l - \eta \delta_l y_{l-1}^T \tag{9}$$

$$b_l \leftarrow b_l - \eta \frac{\partial Loss}{\partial b} = b_l - \eta\ \delta_l \tag{10}$$

$$out_{shared} = concat\left(out_{pH}, out_{DO}, out_{COD}, out_{NHN}\right) \tag{11}$$

$$\begin{gathered} \widehat{pH}_\mathrm{t} = \mathrm{ReLU}(\mathrm{MLP}_{pH}(out_{shared})), \\ \widehat{DO}_\mathrm{t} = \mathrm{ReLU}(\mathrm{MLP}_{DO}(out_{shared})), \\ \widehat{CODMn}_\mathrm{t} = \mathrm{ReLU}(\mathrm{MLP}_{COD}(out_{shared})), \\ \widehat{NHN}_\mathrm{t} = \mathrm{ReLU}(\mathrm{MLP}_{NHN}(out_{shared})). \end{gathered} \tag{12}$$

$$Loss = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(pH_{ti} - \widehat{pH}_{ti}\right)^2 + \left(DO_{ti} - \widehat{DO}_{ti}\right)^2 + \left(COD_{ti} - \widehat{COD}_{ti}\right)^2 + \left(NHN_{ti} - \widehat{NHN}_{ti}\right)^2} \quad (13)$$

The parameters update formulas of each layer are expressed in matrix forms, as shown in Equations (9) and (10):

The well-trained model consists of updated w and b finally, and the output of Equation (1) can be obtained. Concatenating the four output vectors of Equation (1) to obtain, as shown in Equation (11), we can obtain the out_{shared}.

out_{shared} is the input of different tower layers (pH, DO, COD_{Mn}, and NH_3-N correspond to different towers) to generate corresponding prediction. Due to the different prediction targets, the tower layer structure can be different. Although the input out_{shared} is the same for all towers, the output of each tower layer is different. The formulas are shown in Equation (12):

Finally, the Root Mean Square Error (RMSE) between the predicted values $(\widehat{pH}_t, \widehat{DO}_t, \widehat{CODMn}_t, \widehat{NHN}_t)$ and the real values $(pH_t, DO_t, CODMn_t, NHN_t)$ is calculated as the loss (see Equation (13)), where N is the number of samples. The loss is backpropagated to update the model parameters until the model converges.

All tasks share a shared parameter layer in the hard parameter sharing structure, and different tower layers are built for different tasks. Such structure reduces the complexity of the model structure and parameters. It ensures the model's flexibility since the model is required to learn a general implicit embedding in the sharing layer to make each task perform better, thus reducing the risk of overfitting.

2.3.2. Soft Parameter Sharing Structure of Multi-Indicator Water Quality Prediction (Multi-Task-Soft)

The shared parameter layer of Multi-Task-Hard cannot reflect the relationship between different tasks well and cannot guarantee the stable performance of the model. Therefore, we propose a soft parameter sharing structure-based multi-indicator water quality prediction (Multi-Task-Soft), which is based on Multi-Task-Hard. In the Multi-Task-Soft, data will be input to modules of different tasks to extract different features. Different tasks jointly maintain an implicit vector to learn the correlation between different indicators.

The architecture of Multi-Task-Soft is similar to that of the Multi-Task-Hard, as shown in Figure 3, which is also composed of four parts. Their main difference is the design of the shared parameter layer. Different from the single parameter sharing layer of Multi-Task-Hard, Multi-Task-Soft inputs the data to modules of different tasks to obtain different outputs. The structure also maintains an implicit vector to learn the correlation between different indicators. The implicit vector is merged with the outputs corresponding to the underlying structures of each task, and the merged results are input to the tower layer. Finally, each tower model will output the prediction results required by the corresponding task. The model process is shown in Algorithm 2.

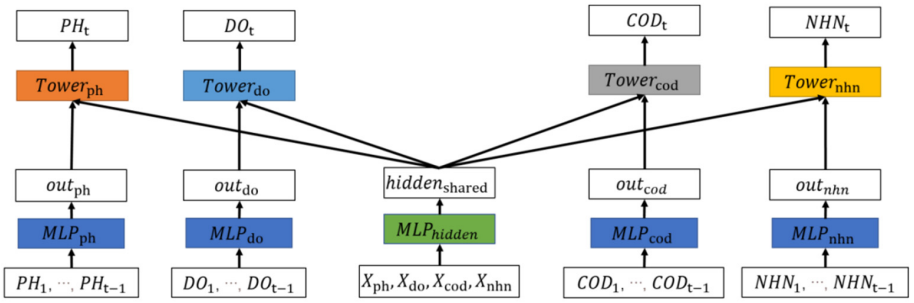

Figure 3. Soft parameter sharing structure of multi-indicator water quality prediction.

Algorithm 2: Multi-indicator water quality prediction based on soft parameter sharing multi-task learning

Input: water quality prediction indicators at the past time intervals $(X_1, \ldots, X_N)$
1 : $hidden_{shared} \leftarrow \mathrm{MLP}_{hidden}([X_1, \ldots, X_N])$
2 : $(\mathrm{out}_1, \ldots, \mathrm{out}_n) \leftarrow \mathrm{MLP}_{1,\ldots,n}(X_1, \ldots, X_N)$
3 : $(Y_1, \ldots, Y_n) \leftarrow \mathrm{Tow}_{1,\ldots,n}([\mathrm{out}_{1,\ldots,n}, \mathrm{out}_{shared}])$
Output: water quality indicators at the future time intervals $(Y_1, \ldots, Y_N)$

Input $(pH_1, \ldots, pH_{t-1}) \in X_{pH}$, $(DO_1, \ldots, DO_{t-1}) \in X_{DO}$, $(COD_{Mn_1}, \ldots, COD_{Mn_{t-1}}) \in X_{COD}$, and $(NHN_1, \ldots, HNH_{t-1}) \in X_{NHN}$ to the model. The data is passed through the fully connected neural network (as shown in Equation (1)) to obtain the output vectors $(out_{pH}, out_{DO}, out_{COD}, out_{NHN})$, respectively.

Meanwhile, (X_{pH}, X_{DO}, X_{COD}, X_{NHN}) is also used as the input of another fully connected neural network to obtain the output vector $hidden_{shared}$, as shown in Equation (14):

$$hidden_{shared} = \mathrm{ReLU}(\mathrm{MLP}(X_{pH}, X_{DO}, X_{COD}, X_{NHN})). \tag{14}$$

Concatenate output vectors and $hidden_{shared}$ to obtain corresponding vectors v_{pH}, v_{DO}, v_{COD}, v_{NHN}.

$$\begin{aligned} v_{pH} &= concat(out_{pH}, hidden_{shared}), \\ v_{DO} &= concat(out_{DO}, hidden_{shared}), \\ v_{COD} &= concat(out_{COD}, hidden_{shared}), \\ v_{NHN} &= concat(out_{NHN}, hidden_{shared}). \end{aligned} \tag{15}$$

Then, we input the vectors to the corresponding tower layer. Similar to Multi-Task-Hard, pH, DO, COD_{Mn}, and NH_3-N correspond to different towers, and the tower layer can be any neural network structure model. For different prediction indicators, the tower layer structure is different, which makes the corresponding output different. Taking MLP as an example, the predictions are shown as Equation (16):

$$\begin{aligned} \widehat{pH_t} &= \mathrm{ReLU}(\mathrm{MLP}_{pH}(v_{pH})), \\ \widehat{DO_t} &= \mathrm{ReLU}(\mathrm{MLP}_{DO}(v_{DO})), \\ \widehat{COD_t} &= \mathrm{ReLU}(\mathrm{MLP}_{COD}(v_{COD})), \\ \widehat{NHN_t} &= \mathrm{ReLU}(\mathrm{MLP}_{NHN}(v_{NHN})). \end{aligned} \tag{16}$$

Finally, the RMSE between the predicted and real values is calculated as the loss, and the model parameters are updated by the backpropagation method until the model converges.

In this structure, the association between different indicators is obtained by learning an implicit public vector, and each task has its unique learning module. Finally, the individual learning and joint learning results are merged to achieve better prediction results.

2.3.3. Gating Parameter Sharing Structure of Multi-Indicator Water Quality Prediction (Multi-Task-Gate)

To better learn the relative weight of different indicators for the task, we further add the gating module in the parameter sharing layer. As shown in Figure 4, the input is processed by different modules to obtain different implicit features. The implicit features obtain the weight of the current task through SoftMax. According to the weight, different implicit vectors are weighted and summed to obtain the tower layer input of each task. Finally, each tower model outputs the prediction results of the corresponding task. The model process is shown as Algorithm 3.

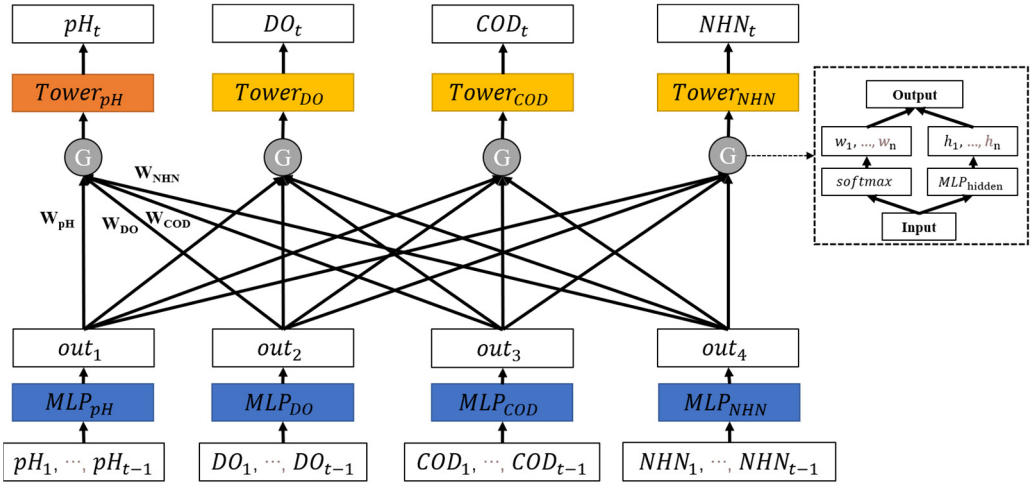

Figure 4. Gating parameter sharing structure of multi-indicator water quality prediction.

The design of the shared parameter layer is similar to Multi-Task-Hard, and the data $(pH_1, \ldots, pH_{t-1}) \in X_{pH}$, $(DO_1, \ldots, DO_{t-1}) \in X_{DO}$, $(COD_{Mn_1}, \ldots, COD_{Mn_{t-1}}) \in X_{COD}$, $(NHN_1, \ldots, HNH_{t-1}) \in X_{NHN}$ is input separately to the fully connected neural network of the shared parameter layer to obtain the output vectors $out_{pH}, out_{DO}, out_{COD}, out_{NHN}$, respectively. It is shown as Equation (1).

Unlike Multi-Task-Hard, the module calculates the importance of different output vectors to predict the pH instead of concatenating them and feeding them to the tower layer. Taking the prediction of pH as an example, we obtain the relative weights of different indicators in the prediction of pH through softmax. Softmax can map relative weights $(w_{pH}, w_{DO}, w_{COD}, w_{NHN})$ from 0 to 1. The relative weights show the corresponding results of different indicators, as shown in Equation (17):

$$(w_{pH}, w_{DO}, w_{COD}, w_{NHN}) = \text{softmax}\left(MLP_{ph}(out_{pH}, out_{DO}, out_{COD}, out_{NHN}\right). \quad (17)$$

Meanwhile, the output vectors $(out_{pH}, out_{DO}, out_{COD}, out_{NHN})$ are mapped through an MLP to $\left(hidden_{pH}, hidden_{DO}, hidden_{COD}, hidden_{NHN}\right)$, as shown in Equation (18):

$$hidden_{pH}, hidden_{DO}, hidden_{COD}, hidden_{NHN} = \text{MLP}_{\text{hidden}}\left(out_{pH}, out_{DO}, out_{COD}, out_{NHN}\right). \quad (18)$$

The vector input of the tower layer is obtained by weighted fusion, as shown in Equation (19):

$$v_{pH} = W_{pH} \times hidden_{pH} + W_{COD} \times hidden_{COD} + W_{DO} \times hidden_{DO} + W_{NHN} \times hidden_{NHN}. \quad (19)$$

Similarly, the tower layers of DO, COD_{Mn}, and NH_3-N also obtain the corresponding inputs, and the tower layers are designed as MLP. For different prediction indicators, the tower layer structure and output can be different. Taking MLP as an example, the formula is shown as Equation (16) in Section 2.3.2.

Finally, the RMSE between the predicted value and the real value is calculated as the loss, and the model parameters are updated by the backpropagation method until the model converges.

The gating parameter sharing structure does not learn the implicit vectors to extract the connection between tasks but learns the importance and connection of different indicators relative to a single task through the gating mechanism, which improves prediction performance.

Algorithm 3: Multi-task Learning of Gating Parameter Sharing Structure for Multi-indicator Water Quality Prediction

Input: water quality prediction indicators at the past time intervals $(X_1, \ldots, X_N)$
1 : $(hidden_1, \ldots, hidden_n) \leftarrow \mathrm{MLP}_{1,\ldots,n}([X_1, \ldots, X_N])$
2 : $(w_{i1}, \ldots, w_{in}) \in \mathrm{W_i} \leftarrow \mathrm{Softmax}(\mathrm{MLP}^{\mathrm{i}}_{shared}(hidden_1, \ldots, hidden_n))$
3 : $(Y_1, \ldots, Y_n) \leftarrow \mathrm{Tow_i}([\sum_i^n W_i \times hidden_i])$
Output: water quality indicators at the future time intervals $(Y_1, \ldots, Y_N)$

2.3.4. Gated Hidden Parameter Sharing Structure of Multi-Indicator Water Quality Prediction (Multi-Task-GH)

This section proposes a multi-task learning structure, which combines the advantages of the soft parameter sharing structure and the gated parameter sharing structure. As shown in Figure 5, the structure of the gated hidden parameter sharing structure (Multi-Task-GH) is similar to the Multi-Task-Gate, except that there is a model for learning an intermediate hidden vector in the parameter sharing layer. This intermediate implicit vector is similar to the Multi-Task-Soft design, which is combined with all other implicit vectors. The output results will be input to the tower layer through the gating mechanism. Finally, each tower model outputs the prediction results of the corresponding task. The model algorithm process is shown in Algorithm 4. Figure 6 shows an example of the kind of time series for each indicator.

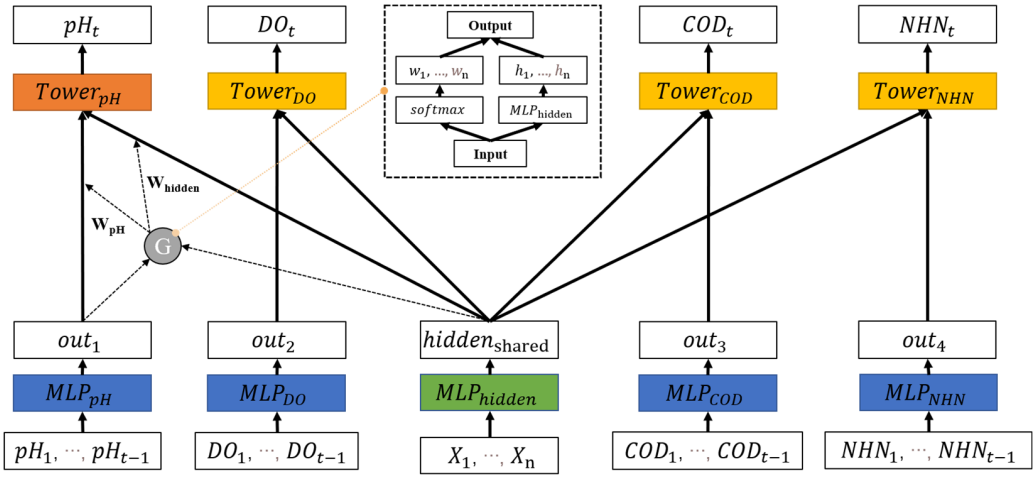

Figure 5. Gated hidden parameter sharing structure of multi-indicator water quality prediction.

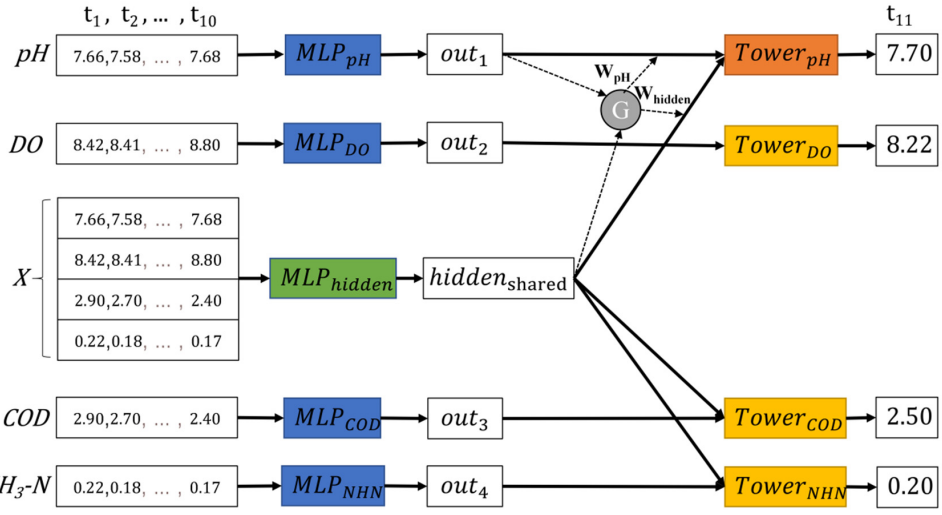

Figure 6. An example of the kind of time series for each indicator.

Algorithm 4: Gated Hidden Parameter Sharing Structure Multi-Task Learning for Multi-indicator Water Quality Prediction

Input: water quality prediction indicators at the past time intervals $(X_1, \ldots, X_N)$
1 : $hidden_{shared} \leftarrow \mathrm{MLP}_{hidden}([X_1, \ldots, X_N])$
2 : $(out_1, \ldots, out_n) \leftarrow \mathrm{MLP}_{1,\ldots,n}(X_1, \ldots, X_N)$
3 : $(w_{i1}, \ldots, \mathrm{w}_{in}) \in \mathrm{W}_i \leftarrow \mathrm{Softmax}(\mathrm{MLP}^{i}_{shared}([out_i, hidden_{shared}]))$
4 : $(Y_1, \ldots, Y_n) \leftarrow \mathrm{Tow}_i([\sum_i^n W_i * [out_i, hidden_{shared}]])$
Output: water quality indicators at the future time intervals $(Y_1, \ldots, Y_N)$

The design of the shared parameter layer is similar to Multi-Task-Soft, and data is input separately to the input layer: $(pH_1, \ldots, pH_{\mathrm{t}-1}) \in X_{pH}$, $(DO_1, \ldots, DO_{\mathrm{t}-1}) \in X_{DO}$, $(COD_{Mn_1}, \ldots, COD_{Mn_{\mathrm{t}-1}}) \in X_{COD}$, and $(NHN_1, \ldots, HNH_{\mathrm{t}-1}) \in X_{NHN}$. The data is passed to MLP of the shared parameter layer to obtain the output vectors, respectively. It is shown in Equation (1).

We then input $(X_{pH}, X_{DO}, X_{COD}, X_{NHN})$ to another implicit vector MLP to obtain the output vector $hidden_{shared}$. As shown in Equation (20):

$$hidden_{shared} = \mathrm{ReLU}\big(\mathrm{MLP}\big(X_{pH}, X_{DO}, X_{COD}, X_{NHN}\big)\big) \tag{20}$$

Unlike Multi-Task-Soft, the module calculates the importance of different vectors out_{pH}, out_{DO}, out_{cod} and out_{nhn} for the prediction target together with $hidden_{shared}$, respectively. The relative weight of the predicted target is obtained through Softmax, as shown in Equation (21):

$$\begin{aligned}
(\mathrm{w}_{pH}, \mathrm{w}_{hidden}) &= \mathrm{Softmax}\left(\mathrm{MLP}_{pH}\left(out_{pH}, hidden_{shared}\right)\right),\\
(\mathrm{w}_{DO}, \mathrm{w}_{hidden}) &= \mathrm{Softmax}\left(\mathrm{MLP}_{DO}\left(out_{DO}, hidden_{shared}\right)\right),\\
(\mathrm{w}_{COD}, \mathrm{w}_{hidden}) &= \mathrm{Softmax}\left(\mathrm{MLP}_{COD}\left(out_{COD}, hidden_{shared}\right)\right),\\
(\mathrm{w}_{NHN}, \mathrm{w}_{hidden}) &= \mathrm{Softmax}\left(\mathrm{MLP}_{NHN}\left(out_{NHN}, hidden_{shared}\right)\right).
\end{aligned} \tag{21}$$

Meanwhile, the output vectors are mapped through a fully connected neural network to $(hidden_{pH}, hidden_{DO}, hidden_{COD}, hidden_{NHN})$:

$$\begin{aligned}
hidden_{pH} &= \mathrm{ReLU}\big(\mathrm{MLP}_{hidden}\left(out_{pH}, hidden_{shared}\right)\big),\\
hidden_{DO} &= \mathrm{ReLU}\left(\mathrm{MLP}_{hidden}\left(out_{DO}, hidden_{shared}\right)\right),\\
hidden_{COD} &= \mathrm{ReLU}\left(\mathrm{MLP}_{hidden}\left(out_{COD}, hidden_{shared}\right)\right),\\
hidden_{NHN} &= \mathrm{ReLU}\left(\mathrm{MLP}_{hidden}\left(out_{NHN}, hidden_{shared}\right)\right).
\end{aligned} \tag{22}$$

The vector input of the tower layer is obtained by weighted fusion:

$$\begin{aligned}
v_{pH} &= W_{pH} \times hidden_{pH} + W_{hidden} \times hidden_{pH},\\
v_{DO} &= W_{DO} \times hidden_{DO} + W_{hidden} \times hidden_{DO},\\
v_{COD} &= W_{COD} \times hidden_{COD} + W_{hidden} \times hidden_{COD},\\
v_{NHN} &= W_{NHN} \times hidden_{NHN} + W_{hidden} \times hidden_{NHN}.
\end{aligned} \tag{23}$$

We then input v_{DO}, v_{COD} and v_{NHN} into the corresponding tower layers. For different prediction indicators, the tower layer structure and output are different. Taking MLP as an example, the formula is shown as Equation (16) in Section 2.3.2. Finally, the RMSE between the predicted and real values is calculated as the loss, and the model parameters are updated by the backpropagation method until the model converges.

2.3.5. Summary of Four Water Quality Prediction Models

The structure of the proposed four water quality prediction models is summarized in Table 1. The input layer and output are not listed in the table due to their similarity and simplicity. For more details, please refer to Appendix A.

Table 1. The structure of the proposed four water quality prediction models.

Name	Layer	Design
Mt-Hard	Shared parameter layer	1 × (MLP + Relu)
	Tower layer	pH: 3 × (MLP + ReLU) DO: 3 × (MLP + ReLU) COD_{Mn}: 2 × (MLP + ReLU) NH_3-N: 2 × (MLP + ReLU)
Mt-Soft	Shared parameter layer	pH, DO, COD_{Mn}, NH_3-N: 1 × (MLP + ReLU) Hidden: 2 × (MLP + ReLU)
	Tower layer	pH: 3 × (MLP + ReLU) DO: 3 × (MLP + ReLU) COD_{Mn}: 2 × (MLP + ReLU) NH_3-N: 2 × (MLP + ReLU)
Mt-Gate	Shared parameter layer	pH, DO, COD_{Mn}, NH_3-N: 1 × (MLP + ReLU)
	Tower layer	pH: Softmax + 3 × (MLP + ReLU) DO: Softmax + 3 × (MLP + ReLU) COD_{Mn}: Softmax + 2 × (MLP + ReLU) NH_3-N: Softmax + 2 × (MLP + ReLU)
Mt-GH	Shared parameter layer	pH, DO, COD_{Mn}, NH_3-N: 1 × (MLP + ReLU) Hidden: 2 × (MLP + ReLU)
	Tower layer	pH: Softmax + 3 × (MLP + ReLU) DO: Softmax + 3 × (MLP + ReLU) COD_{Mn}: Softmax + 2 × (MLP + ReLU) NH_3-N: Softmax + 2 × (MLP + ReLU)

3. Experiment Setup

This section introduces the datasets, evaluation metrics, baseline models, and model settings for the evaluation.

3.1. Datasets

The experiment datasets come from 147 water quality monitoring stations set up by China National Environmental Monitoring Station in China's seven river systems and lakes. Each station's monitoring water quality indicators include pH, DO, COD_{Mn}, and NH_3-N. We have two datasets: D-s (Dataset-short) from 2013 to 2015 and D-l (Dataset-long) from 2012 to 2018. We select 120 stations with relatively complete data as the experiment dataset. Among them, there are 7 monitoring stations in the Pearl River, 22 in the Yangtze River, 11 in the Songhua River, 7 in the Liaohe River, 12 in the Yellow River, 26 in the Huaihe River, 6 in the Haihe River, 6 in the Taihu Lake, 4 in Poyang Lake, and 18 in other large lakes and rivers. Detailed statistics of the dataset are shown in Table 2.

Table 2. Dataset statistics.

Name	Number of Sites	D-s	D-l
		Time	Time
Total data set	120	2013.1–2015.2	2012.6–2018.4
Pearl River	8	2013.1–2015.2	2012.6–2018.4
The Yangtze River	22	2013.1–2015.2	2012.6–2018.4
Songhua River	11	2013.1–2015.2	2012.6–2018.4
Liaohe River	7	2013.1–2015.2	2012.6–2018.4
The Yellow River	12	2013.1–2015.2	2012.6–2018.4
Huaihe River	26	2013.1–2015.2	2012.6–2018.4
Haihe River	6	2013.1–2015.2	2012.6–2018.4
Taihu Lake	6	2013.1–2015.2	2012.6–2018.4
Poyang Lake	4	2013.1–2015.2	2012.6–2018.4
Other	18	2013.1–2015.2	2012.6–2018.4

3.2. Evaluation Metrics

We select Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE), and Mean Absolute Error (MAE) as the evaluation metrics, which are widely used in time series prediction models [29]. Note that the lower the values of RMSE, MAPE, and MAE, the better the performance. The RMSE, MAE, and MAPE are calculated as follows:

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2} \tag{24}$$

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i| \tag{25}$$

$$\mathrm{MAPE} = \frac{100\%}{N}\sum_{i=1}^{N}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{26}$$

where y_i indicates the i-th real value, $\hat{y}_i$ indicates the i-th predicted value, and N is the number of data samples. These three metrics are used to measure the error between the predicted values and the real values. MAPE reflects the relative error between the predicted values and the real values, while MAE is a simple superposition of the absolute error. Therefore, MAPE can more accurately reflect the deviation degree of the predicted values. At the same time, RMSE first squares the error values. If the dispersion of errors is high, the RMSE is magnified. Therefore, RMSE is more affected by outliers than MAE and MAPE, but they are at the same data level [30].

3.3. Baselines

The baselines for comparison are as follows: Linear model [16], XGBoost model [31,32], MLP model [33], CNN model [34], LSTM model [19], GRU (Gated Recurrent Unit) model [35], and ATTENTION model (ATT for short) [36,37]. Our proposed models are Mt-Hard (Multi-Task-Hard), Mt-Soft (Multi-Task-Soft), Mt-Gate (Multi-Task-Gate), and Mt-GH (Multi-Task-GH).

3.4. Model Setting

For model learning, the input space node number is 120, the sequence length is 10, and the dimension of each time point is 4, representing four water quality indicators (pH, DO, COD_{Mn}, and NH_3-N). For prediction, the output space node number is also 120, the sequence length is set to 1, and the water quality indicators at each time point are also pH, DO, COD_{Mn}, and NH_3-N. The first 60% of the data is used for training, 20% is used for validation, and the last 20% is used for testing. The prediction time step is set to 1. In other words, the historical water quality values of 120 monitoring stations in the previous ten weeks are used to predict their values in the next week. We compare the proposed models with other models to verify the effectiveness of the proposed models.

For all deep learning models, Adam is used as the optimizer, which combines the advantages of AdaGrad (adaptive gradient) and RMSProp (root mean square propagation) to update the step size by comprehensively considering the first-moment estimation (i.e., the mean value of the gradient) and the second-moment estimation (i.e., the variance of the gradient). The learning rate can be automatically adjusted, and the fluctuation range of the adjustment is not too large [29]. The hyperparameters are highly interpretable and usually only need to be fine-tuned or even not need to be adjusted, which is suitable for large-scale data and parameter scenarios. We choose RMSE as the loss function. The learning rate is set to 0.001, and the epochs and batch sizes are set to 100 and 5, respectively.

4. Results and Discussion

In this section, we compare the proposed method with baselines on the prediction performance of the single-indicator and multi-indicator. We then compare the influence

of different tower layers on the model to verify the proposed methods' robustness and analyze the models' predictive performance for different rivers and lakes. We also show the training loss and validation loss of the best multi-task water quality prediction model.

4.1. Comparison of Prediction Performance for Single-Indicator

In this section, we compare the overall prediction performance of four multi-task learning models with seven baselines for single-indicator. The experimental results are shown in Table 3. In the table, the bold numbers are the best, and the numbers with asterisk are the second best.

Table 3. Comparison of the overall performance of prediction for single-indicator on D-s dataset.

Model	pH			DO			COD_{Mn}			NH_3-N		
	RMSE	MAE	MAPE	RMSE	MAE	MAPE	RMSE	MAE	MAPE	RMSE	MAE	MAPE
Linear [16]	0.560	0.413	0.054	2.657	1.978	0.299	2.793	1.236	0.354	1.366	0.423	0.948
XGB [31]	0.327	0.245	0.032	1.594	1.135	0.121	1.732	0.614	0.179	0.487	0.182	0.335
MLP [33]	0.299	0.211	0.028	1.218 *	0.828	0.910	**1.485**	**0.588**	0.178	0.474	0.164 *	**0.317**
CNN [34]	0.429	0.327	0.043	2.066	1.506	0.167	2.541	1.197	0.350	0.718	0.347	0.702
LSTM [19]	0.294	0.208	0.027	1.396	0.956	0.103	1.658	0.617	0.174	0.467	0.171	0.444
GRU [35]	0.282 *	0.206 *	0.027 *	1.230	0.821 *	0.087 *	1.742	0.610	0.169	0.457 *	0.172	0.434
ATT [37]	0.478	0.387	0.050	1.672	1.079	0.106	2.035	0.618	0.168 *	0.681	0.193	0.416
Mt-Hard	0.270	0.217	0.028	1.273	0.869	0.094	1.535	0.602	0.169	0.432	0.244	0.640
Mt-Soft	0.293	0.209	0.028	1.186	0.801	0.087	1.534	0.597	0.178	0.430	0.168	0.386
Mt-Gate	0.292	0.211	0.027	1.235	0.851	0.091	1.547	0.585	**0.166**	0.448	0.178	0.386
Mt-GH	**0.262**	**0.181**	**0.024**	**1.182**	**0.796**	**0.086**	1.515	0.592	0.173	**0.403**	**0.154**	0.331
Improv.	7.1%	12.1%	11.1%	3.0%	3.0%	1.1	-	-	1.6%	11.7%	6.3%	-

* In the table, the bold numbers are the best, and the numbers with asterisk are the second best.

(1) For pH, the hard parameter sharing structure (Multi-Task-Hard), soft parameter sharing structure (Multi-Task-Soft), gated parameter sharing structure (Multi-Task-Gate), and gated hidden parameter sharing structure (Multi-Task-GH) achieve better performance in all the metrics. Multi-Task-GH achieves the best performance, which means the pH predicted by Multi-Task-GH is closer to the real values.

(2) For DO, the four multi-task learning models also achieve better performance. Among the four multi-task learning models, the performance of Multi-Task-Hard is worse than other multi-task models (Multi-Task-Soft, Multi-Task-Gate, and Multi-Task-GH) and even worse than some traditional deep learning models (MLP and GRU). The Multi-Task-GH still achieves the best performance.

(3) For COD_{Mn}, the MLP model achieves the best performance in RMSE and MAE. Only Multi-Task-Gate achieves the best performance in MAPE among the four multi-task learning models. The prediction performance of the three soft parameter sharing models, Multi-Task-Soft, Multi-Task-Gate, and Multi-Task-GH, is almost the same as that of MLP, which means that the predicted COD_{Mn} of MLP is closer to the observed values. However, the multi-task learning model with three soft parameters shared can still achieve close results.

(4) For NH_3-N, MLP achieves the best performance in only MAPE, while Multi-Task-GH achieves the best results in both RMSE and MAE. This means that the NH_3-N predicted by the Multi-Task-GH model is closer to the real values in most cases.

As shown in Table 4, we further validate the proposed models on the dataset D-l. In the table, the bold numbers are the best. Similar to the results on D-s, the results also show that the multi-task learning models achieve better performance than other models in most cases and Multi-Task-GH achieves the best results in most.

Table 4. Comparison of the overall performance of prediction for single-indicator on D-l dataset.

Model	pH			DO			COD_{Mn}			NH_3-N		
	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE
CNN	0.456	0.047	0.357	2.049	0.192	1.547	1.487	0.346	1.051	0.323	0.946	0.188
LSTM	0.268	0.025	0.195	1.198	0.110	0.828	0.939	0.189	0.582	0.238	0.430	0.118
GRU	**0.242**	**0.022**	**0.168**	1.200	0.107	**0.813**	0.944	0.190	0.579	0.219	0.510	0.116
Mt-Soft	0.252	0.023	0.178	1.196	0.106	0.823	0.949	0.188	0.577	0.222	0.395	0.114
Mt-Gate	0.251	0.023	0.178	1.197	0.108	0.823	0.939	0.194	0.582	0.217	0.415	0.109
Mt-GH	0.256	0.023	0.181	**1.196**	**0.106**	0.824	**0.932**	**0.185**	**0.574**	**0.214**	**0.407**	**0.105**
Mt-GH	0.256	0.023	0.181	**1.196**	**0.106**	0.824	**0.932**	**0.185**	**0.574**	**0.214**	**0.407**	**0.105**

4.2. Comparison of Four Indicators and Three Indicators Multi-Task Learning Models

It is worth mentioning that we have conducted experiments on three indicators of multi-task learning models. The experimental results show a similar conclusion, but the whole performance is weaker than the four ones. The results are shown in Table 5, where 4-task means four indicator multi-task learning model, 3-tasks are three indicator multi-task learning models that include (pH, DO, COD_{Mn}), and (DO, COD_{Mn}, NH_3-N), (pH, DO, NH_3-N), and (pH, COD_{Mn}, NH_3-N) multi-task learning models. In the table, the bold numbers are the best. The results on D-s have similar trends.

Table 5. Comparison of four indicators and three indicators multi-task learning models on D-l.

	pH			DO			COD_{Mn}			NH_3-N		
	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE
4-task	**0.256**	**0.023**	**0.181**	**1.196**	**0.106**	**0.824**	**0.932**	**0.185**	**0.574**	**0.214**	**0.407**	**0.105**
3-task1	0.324	0.031	0.235	1.248	0.116	0.877	0.959	0.194	0.592	—	—	—
3-task2	—	—	—	1.277	0.116	0.892	0.965	0.193	0.594	0.234	0.514	0.136
3-task3	0.316	0.030	0.226	1.242	0.111	0.861	—	—	—	0.222	0.650	0.127
3-task4	0.342	0.033	0.255	—	—	—	0.977	0.189	0.596	0.221	0.440	0.107

4.3. Comparison of Prediction Performance for Multi-Indicators

Table 6 shows the average prediction performance of seven baselines and four multi-task learning models for multi-indicators on D-s. In the table, the bold numbers are the best, and the numbers with asterisk are the second best. For the space limitation, we only put the results on D-s in the following sections because the results on D-s and D-l have similar trends. Among the models, the Multi-Task-GH model achieves the best on all the metrics. Although the single-task learning models may achieve the best effect in predicting one target water quality indicator in some cases, the prediction accuracy will decrease when predicting other water quality indicators. Therefore, when the same model structure is used to simultaneously predict multiple target water quality indicators (pH, DO, COD_{Mn}, and NH_3-N), the Multi-Task-GH model can accomplish this task well and achieve the best performance in most indicators. This means that the Multi-Task-GH model can accurately predict multi-indicator.

4.4. Tower Layer Analysis

This paper also analyzes the impact of different tower types on the prediction performance of the Multi-task-GH model, as shown in Table 7. In the table, the bold numbers are the best. The five deep learning structures of LSTM, GRU, CNN, ATTENTION, and MLP are used as the tower layer of the Multi-Task-GH model to train the model and predict the water quality indicators. The results show that the Multi-Task-GH model with MLP as the tower layer achieves the best performance in most cases. The robustness of the model is the best, and there is no sharp drop in the prediction accuracy when predicting different water quality indicators. For example, when the ATTENTION-based deep learning structure is used as the tower layer of the Multi-Task-GH model, the prediction results of DO, COD_{Mn}, and NH_3-N are good, while the prediction results of pH are greatly reduced, which achieve the worst performance among the five structures. This shows that the ATTENTION-based

deep learning structure is not well compatible with the simultaneous prediction of four water quality indicators.

Table 6. Comparison of the overall performance of prediction for multi-indicator.

Model	RMSE	MAE	MAPE
Linear [16]	7.376	4.052	1.656
XGB [31]	4.139	2.177	0.667
MLP [33]	3.476 *	1.792 *	0.615 *
CNN [34]	5.753	3.377	1.262
LSTM [19]	3.814	1.952	0.748
GRU [35]	3.709	1.809	0.716
ATT [37]	4.866	2.277	0.730
Mt-Hard	3.511	1.932	0.930
Mt-Soft	3.443	1.775	0.679
Mt-Gate	3.523	1.823	0.670
Mt-GH	**3.362**	**1.723**	**0.614**
Improv.	3.3%	3.8%	1.6%

* In the table, the bold numbers are the best, and the numbers with asterisk are the second best.

Table 7. The performance comparison of tower structures in the Multi-Task-GH model.

Tower Type	pH RMSE	pH MAE	pH MAPE	DO RMSE	DO MAE	DO MAPE	COD_{Mn} RMSE	COD_{Mn} MAPE	COD_{Mn} MAE	NH_3-N RMSE	NH_3-N MAPE	NH_3-N MAE
LSTM	7.009	6.975	0.904	9.178	8.872	0.919	4.523	0.735	2.978	1.080	0.515	0.343
GRU	0.847	0.396	0.051	2.311	1.642	0.172	2.705	0.369	1.313	1.120	0.627	0.423
CNN	0.464	0.359	0.046	1.949	1.407	0.154	2.576	0.371	1.304	0.950	0.896	0.367
ATT	7.725	7.708	1.0	1.459	0.952	0.098	1.987	**0.162**	0.614	0.474	0.430	0.186
MLP	**0.262**	**0.181**	**0.024**	**1.182**	**0.796**	**0.086**	**1.515**	0.173	**0.592**	**0.403**	**0.331**	**0.154**

4.5. The Difference of Predictions and Real Data

To show the ability of the Multi-Task-GH model, we the difference between predictions and the real measured data, as shown in Figure 7. The curves of both present similar trends, which prove the Multi-Task-GH model can predict the indicator change.

Figure 7. The difference between predictions and real data.

4.6. Model Training Loss and Validation Loss

We train the baselines and the proposed four multi-task learning models with fixed hyperparameters. As shown in Figure 8, the two curves are the training loss and validation loss change curves of the Multi-Task-GH model. The two-loss curves converge after about ten epochs of training.

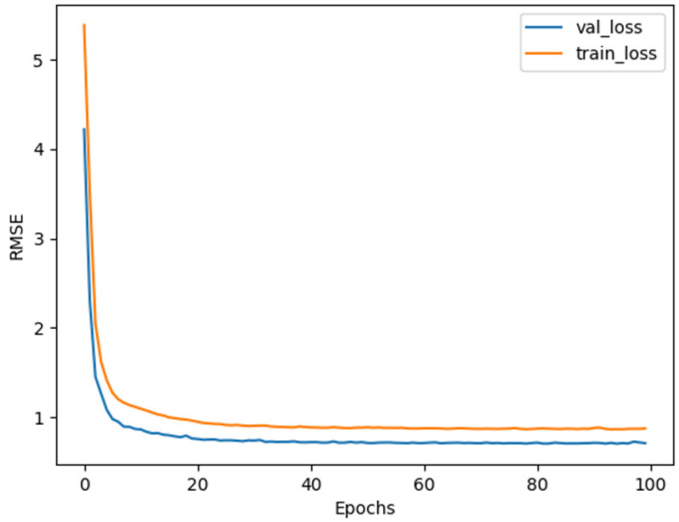

Figure 8. The learning curve of the Multi-Task-GH model.

4.7. Related Work Analysis

With the rapid development of machine learning, scholars have begun to explore water quality prediction methods based on machine learning, such as the support vector machine, genetic algorithm, and clustering algorithm. Recently, some researchers have proposed deep learning methods for water quality prediction, mainly aimed at predicting a single indicator. These models are based on single-task learning, and the representative models are as follows:

Avila et al. [16] adopted the ridge regression method to predict water quality. Lu et al. [31] used PCA to assess water quality. Chen et al. [32] used a machine learning algorithm with an integrated boosting method. Ahmed et al. [33] stacked multiple fully connected layers to predict water quality. Barzegar et al. [34] employed one convolution layer and LSTM layers for water quality parameter prediction. Yang et al. [21] incorporated one LSTM and two fully connected layers for prediction, which can extract short-term and long-term correlations of water quality and avoid gradient disappearance. Shrestha et al. [35] incorporated one GRU and two fully connected layers for water quality prediction.

Vaswani et al. and Jaderberg et al. [36,37] stacked three self-attention layers and two fully connected layers to mine the sequential relationship of water quality data. The input sequence is first converted to embedding through the first fully connected layer. The converted embedding then completes the information aggregation on the time step through the three-layer self-attention mechanism. Finally, it generates the water quality prediction through a fully connected layer.

Prediction methods based on deep learning can well extract the complex nonlinear characteristics and time-dependent relationship of water quality data, which achieves good prediction performance, but they still have some problems.

(1) Unable to predict multiple indicators with one model. The trained model often only performs well in the one prediction indicator. If the model is used to predict other indicators without changing the model's structure and parameters, the performance will be greatly reduced. Therefore, it is necessary to train different models to predict multiple indicators, which will lead to extended training and prediction time and large model storage space.

(2) Unable to consider the impact correlations between multiple indicators. When the water quality of the same water area becomes better or worse, there may be a certain correlation between different indicators. The single-indicator prediction model is difficult to deal with the correlations between multiple indicators.

This paper proposes water quality prediction models based on multi-task learning to solve the above problems. The model based on multi-task learning improves the prediction performance of each task, which learns the relevance between different tasks. The multi-task learning also saves parameter space and prediction time consumption by sharing part of the model [38,39].

5. Conclusions

This paper proposed a multi-task-learning-based prediction method to solve the shortcomings and challenges of the single-task learning model for water quality prediction. Four multi-task learning structures are proposed based on the idea of sharing bottom structures: hard parameter sharing structure, soft parameter sharing structure, gated parameter sharing structure, and gated hidden parameters sharing structure. Sufficient experiments are designed and implemented to demonstrate the effectiveness of the proposed method.

However, there is still room for improving the proposed method. The training gradient losses of different tasks in reverse gradient propagations show a magnitude gap, leading to unstable training. It is hard to train a large number of water quality indicators in multi-task learning because the balance of four indicators will be out of control.

In this paper, we did not take the data distributions and importance of different tasks into consideration explicitly because the Multi-task-GH can implicitly learn the unique and shared joint weights of each subtask through the gate network, and it can also implicitly reflect the different effects of data distributions. However, if the data distributions and importance of different tasks can be explicitly taken into consideration, e.g., as constraints of loss function or regularization, the models would have better performance. In the future, we will conduct more data analysis and design more reasonable losses for the tasks.

Author Contributions: Conceptualization, S.C. and K.X.; methodology, H.W. and N.M.; validation, J.C. and L.T.; formal analysis, M.G.; investigation, S.C.; data curation, L.T.; writing—original draft preparation, H.W.; writing—review and editing, M.G. and K.X.; visualization, N.M.; supervision, S.C.; project administration, H.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Key Research and Development Program of China, grant number 2020YFB1712901, and the Research Program of Chongqing Technology Innovation and Application Development, China, grant number cstc2020kqjscx-phxm1304.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Publicly available datasets were analyzed in this study. This data can be found here: http://envi.ckcest.cn/environment/special/special_list.jsp?specialId=108, accessed on 10 October 2021.

Conflicts of Interest: The authors declare no conflict of interest. The company "T.Y.Lin International Engineering Consulting (China) Co., Ltd." had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

Appendix A. Hyper-Parameters

In our proposed framework, there are many hyper-parameters. For MLP, we tune the number of hidden layers in a range of {1, 2, 3} and tune the number of hidden units in a range of {16, 32, 64}. Without specific mention, we set hidden units of the first MLP layer as 64, the second MLP layer as 32, and the third MLP layer as 16. We further tune the number of epochs for the training process in a range of {100, 200, 400}, the number of batch size is 8,

and the learning rate is 0.001. The input space node number is 120, the sequence length is 10, and the dimension of each time point is 4, representing four water quality indicators: pH, DO, COD_{Mn}, and NH_3-N. All the information is summarized in Table A1.

Table A1. Hyper-parameters.

Name	Experimental Set
MLP layer	1, 2, 3
	6.97
	0.904
MLP hidden unit units	16, 32, 64
	0.396
	0.051
Epoch	100, 200, 400
	0.359
	0.046
Batch size	8
	7.708
	1.0
Learning rate	0.001
	0.181
	0.024
node number	120
Sequence length	10
Prediction targets	4

Table A2 shows the details of input-output parameters.

Table A2. The details of input-output parameters.

Inputs	Outputs	Time Windows Size	Mean	Standard Variance	Maximum	Minimum
pH	pH	10	7.240	0.330	7.950	6.390
DO	DO	10	7.340	0.640	10.000	6.200
COD_{Mn}	COD_{Mn}	10	1.820	0.450	3.300	0.800
NH_3-N	NH_3-N	10	0.194	0.045	0.340	0.110

References

1. Votruba, L. *Analysis of Water Resource Systems*; Elsevier: Amsterdam, The Netherlands, 1988.
2. Olyaie, E.; Abyaneh, H.Z.; Mehr, A.D. A comparative analysis among computational intelligence techniques for dissolved oxygen prediction in Delaware River. *Geosci. Front.* **2017**, *8*, 517–527. [CrossRef]
3. Cortes, C.; Vapnik, V. Support-vector networks. *Mach. Learn.* **1995**, *20*, 273–297. [CrossRef]
4. Drucker, H. Improving regressors using boosting techniques. In Proceedings of the 14th International Conference on Machine Learning, San Francisco, CA, USA, 8–12 July 1997; pp. 107–115.
5. Vapnik, V.; Golowich, S.; Smola, A. Support Vector Method for Function Approximation, Regression Estimation and Signal Processing. In *Advances in Neural Information Processing Systems*; Mozer, M.C., Jordan, M., Petsche, T., Eds.; MIT Press: Cambridge, MA, USA, 1997; pp. 281–287.
6. Li, X.; Cheng, Z.; Yu, Q.; Bai, Y.; Li, C. Water-quality prediction using multimodal support vector regression: Case study of Jialing River, China. *J. Environ. Eng.* **2017**, *143*, 04017070. [CrossRef]
7. Huang, N.E.; Shen, Z.; Long, S.R. The empirical mode decomposition and the Hilbert spectrum for nonlinear and nonstationary time series analysis. *Process R. Soc. Lond.* **1998**, *454*, 903–995. [CrossRef]
8. Wu, Z.; Huang, N.E. Ensemble empirical mode decomposition: A noiseassisted data analysis method. *Adv. Adapt. Data Anal.* **2009**, *1*, 1–41. [CrossRef]
9. Yeh, J.R.; Shieh, J.S.; Huang, N.E. Complementary ensemble empirical mode decomposition: A novel noise enhanced data analysis method. *Adv. Adapt. Data Anal.* **2010**, *2*, 135–156. [CrossRef]
10. Leong, W.C.; Bahadori, A.; Zhang, J.; Ahmad, Z. Prediction of water quality index (WQI) using support vector machine (SVM) and least square-support vector machine (LS-SVM). *Int. J. River Basin Manag.* **2021**, *19*, 149–156. [CrossRef]
11. Rashed, E.A.; Hirata, A. Infectivity upsurge by COVID-19 viral variants in Japan: Evidence from Deep Learning Modeling. *Int. J. Environ. Res. Public Health* **2021**, *18*, 7799. [CrossRef]

12. Dildar, M.; Akram, S.; Irfan, M.; Khan, H.U.; Ramzan, M.; Mahmood, A.R.; Alsaiari, S.A.; Saeed, A.H.M.; Alraddadi, M.O.; Mahnashi, M.H. Skin cancer detection: A review using deep learning techniques. *Int. J. Environ. Res. Public Health* **2021**, *18*, 5479. [CrossRef]
13. Banejad, H.; Olyaie, E. Application of an artificial neural network model to rivers water quality indexes prediction—A case study. *J. Am. Sci.* **2011**, *7*, 60–65.
14. Heddam, S. Multilayer perceptron neural network-based approach for modeling pHycocyanin pigment concentrations: Case study from lower Charles River buoy, USA. *Environ. Sci. Pollut. Res.* **2016**, *23*, 17210–17225. [CrossRef] [PubMed]
15. Heddam, S. Generalized regression neural network-based approach for modeling hourly dissolved oxygen concentration in the Upper Klamath River, Oregon, USA. *Environ. Technol.* **2014**, *35*, 1650–1657. [CrossRef] [PubMed]
16. Avila, R.; Horn, B.; Moriarty, E.; Hodson, R.; Moltchanova, E. Evaluating statistical model performance in water quality prediction. *J. Environ. Manag.* **2018**, *206*, 910–919. [CrossRef]
17. Zhou, Z.H.; Feng, J. Deep Forest. *Natl. Sci. Rev.* **2019**, *6*, 74–86. [CrossRef]
18. Wang, Z.; Man, Y.; Hu, Y.; Li, J.; Hong, M.; Cui, P. A deep learning based dynamic COD prediction model for urban sewage. *Environ. Sci. Water Res. Technol.* **2019**, *5*, 2210–2218. [CrossRef]
19. Zou, Q.; Xiong, Q.; Li, Q.; Yi, H.; Yu, Y.; Wu, C. A water quality prediction method based on the multi-time scale bidirectional long short-term memory network. *Environ. Sci. Pollut. Res.* **2020**, *27*, 16853–16864. [CrossRef]
20. Niu, C.; Tan, K.; Jia, X.; Wang, X. Deep learning based regression for optically inactive inland water quality parameter estimation using airborne hyperspectral imagery. *Environ. Pollut.* **2021**, *286*, 117534. [CrossRef]
21. Yang, Y.; Xiong, Q.; Wu, C.; Zou, Q.; Yu, Y.; Yi, H.; Gao, M. A study on water quality prediction by a hybrid CNN-LSTM model with attention mechanism. *Environ. Sci. Pollut. Res.* **2021**, *28*, 55129–55139. [CrossRef] [PubMed]
22. Guo, H.; Tian, S.; Huang, J.J.; Zhu, X.; Wang, B.; Zhang, Z. Performance of deep learning in mapping water quality of Lake Simcoe with long-term Landsat archive. *ISPRS J. Photogramm. Remote Sens.* **2022**, *183*, 451–469. [CrossRef]
23. Chen, K.; Chen, H.; Zhou, C.; Huang, Y.; Qi, X.; Shen, R.; Liu, F.; Zuo, M.; Zou, X.; Wang, J.; et al. Comparative analysis of surface water quality prediction performance and identification of key water parameters using different machine learning models based on big data. *Water Res.* **2020**, *171*, 115454. [CrossRef]
24. Zhong, F.; Wu, J.; Dai, Y.; Deng, Z.; Cheng, S. Responses of water quality and phytoplankton assemblages to remediation projects in two hypereutrophic tributaries of Chaohu Lake. *J. Environ. Manag.* **2019**, *248*, 109276. [CrossRef] [PubMed]
25. Weinberger, K.; Dasgupta, A.; Langford, J.; Smola, A.; Attenberg, J. Feature hashing for large scale multi-task learning. In Proceedings of the 26th Annual International Conference on Machine Learning, Montreal, QC, Canada, 14–18 June 2009; pp. 1113–1120.
26. Huang, W.; Song, G.; Hong, H.; Xie, K. Deep architecture for traffic flow prediction: Deep belief networks with multi-task learning. *IEEE Trans. Intell. Transp. Syst.* **2014**, *15*, 2191–2201. [CrossRef]
27. Mao, C.; Gupta, A.; Nitin, V.; Ray, B.; Song, S.; Yang, J.; Vondrick, C. Multi-task learning strengthens adversarial robustness. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2020; pp. 158–174.
28. Yu, F.; Chen, H.; Wang, X.; Xian, W.; Chen, Y.; Liu, F.; Madhavan, V.; Darrell, T. Bdd100k: A diverse driving dataset for heterogeneous multi-task learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 2636–2645.
29. Willmott, C.J.; Matsuura, K. Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Clim. Res.* **2005**, *30*, 79–82. [CrossRef]
30. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
31. Lu, W.; Wu, J.; Li, Z.; Cui, N.; Cheng, S. Water quality assessment of an urban river receiving tail water using the single-factor index and principal component analysis. *Water Sci. Tech.* **2019**, *19*, 603–609. [CrossRef]
32. Chen, T.; Guestrin, C. Xgboost: A scalable tree boosting system. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, LA, USA, 13–17 August 2016; pp. 785–794.
33. Ahmed AA, M. Prediction of dissolved oxygen in Surma River by biochemical oxygen demand and chemical oxygen demand using the artificial neural networks (ANNs). *J. King Saud Univ.-Eng. Sci.* **2017**, *29*, 151–158. [CrossRef]
34. 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]
35. Jiang, Y.; Li, C.; Sun, L.; Guo, D.; Zhang, Y.; Wang, W. A deep learning algorithm for multi-source data fusion to predict water quality of urban sewer networks. *J. Clean. Prod.* **2021**, *318*, 128533. [CrossRef]
36. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Processing Syst.* **2017**, *30*, 5998–6008.
37. Liu, Y.; Zhang, Q.; Song, L.; Chen, Y. Attention-based recurrent neural networks for accurate short-term and long-term dissolved oxygen prediction. *Comput. Electron. Agric.* **2019**, *165*, 104964. [CrossRef]
38. Collobert, R.; Weston, J. A unified architecture for natural language processing: Deep neural networks with multi-task learning. In Proceedings of the 25th International Conference on Machine Learning, Helsinki, Finland, 5–9 July 2008; pp. 160–167.
39. Lindbeck, A.; Snower, D.J. Multitask learning and the reorganization of work: From Taylorism to holistic organization. *J. Labor Econ.* **2000**, *18*, 353–376. [CrossRef]