# Advance prediction of coastal groundwater levels with temporal convolutional and long short-term memory networks

Xiaoying Zhang[a,b], Fan Dong[a,b], Guangquan Chen[c], Zhenxue Dai[a,b]*

[a] Institute of Intelligent Simulation and Early Warning for Subsurface Environment, Jilin University, Changchun, China
[b] College of Construction Engineering, Jilin University, Changchun, China
[c] Key Laboratory of Marine Sedimentology and Environmental Geology, First Institute of Oceanography, State Oceanic Administration, Qingdao, China
* Correspondence: Zhenxue Dai, dzx@jlu.edu.cn

## Highlights

- TCN- and LSTM-based models were proposed to predict groundwater levels in a coastal aquifer
- Tidal, precipitation, and groundwater levels were utilized as input data in the networks
- In advance 1- day, 3-, 7-, and 15-days groundwater levels were predicted with the highest accuracy of 1-day-lead prediction
- The TCN-based model slightly outperformed the LSTM-based model in accuracy but less efficiency

## Abstract

Prediction of groundwater level is of immense importance and challenges for the coastal aquifer management with rapidly increasing climatic change. With the development of artificial intelligence, the data-driven models have been widely adopted in hydrological processes management. However, due to the limitation of network framework and construction, they are mostly adopted to produce only a one-time step in advance. Here, the temporal convolutional network (TCN) and long short-

term memory (LSTM) based models were developed to predict groundwater levels with different leading periods in a coastal aquifer. The beginning hourly-monitored ten-month data in two monitoring wells were used for model training and testing, and the data of the following three months were used as prediction with 24, 72, 180, and 360 time-steps (1-day, 3-, 7-, and 15-days) in advance. The historical precipitation and tidal level data were incorporated as input data. For the one-step prediction of the two wells, the calculated $R^2$ of the TCN-based models' values were higher and the RMSE values were lower than that of the LSTM-based model in the prediction stage with shorter running times. For the advanced prediction, the model accuracy decreased with the increase of the advancing period from 1-day to 3-, 7-, and 15-days. By comparing the simulation accuracy and efficiency, the TCN-based model slightly outperformed the LSTM-based model but was less efficient in training time. Both models showed great ability to learn complex patterns in advance using historical data with different leading periods and had been proven to be valid localized groundwater level prediction tools in the subsurface environment.

**Keywords:** Groundwater level prediction; Coastal aquifer; Temporal convolutional networks; Long Short-Term Memory

## 1 Introduction

As the economic development and population escalate in the coastal area, the fresh groundwater needs continue to mount, and seawater intrusion has ~~post~~ posed a great threat to the availability of portable water resources globally (Baena-Ruiz et al., 2018). In the United States, Mexico, Canada, Australia, China, India, South Korea, Italy, and Greece, with dense population, many coastal aquifers have experienced salinization caused by seawater intrusion (Barlow and Reichard, 2009; Park et al., 2011; Pratheepa et al., 2015; Zhang et al., 2017; Lu et al., 2013). Protection projects such as aquifer replenishment can be constructed to alleviate seawater intrusion by artificially increasing groundwater recharge in ~~the~~ the aquifer than what occurs naturally (Abdalla and Al-Rawahi, 2012; Lu et al., 2019). The replenishment programs have been operated in the developed areas such as Perth, Western Australia, and California, USA (Garza-Díaz et al., 2019). The infrastructures tend to be costly and out of reach for many developing countries. A reliable seawater intrusion monitoring and the predicting system is still essential and is recognized as the most effective way of keeping fresh~~water~~ water from seawater contamination ~~of seawater~~ (Xu and Hu, 2017).

In the past several decades, conventional numerical models have been widely utilized to simulate and predict ~~the~~ groundwater fluctuation dynamics and chemical variations (Batelaan et al., 2003; Dai et al., 2020; Huang et al., 2015; Li et al., 2002). However, the difficulty of acquiring extensive hydrological and geological data and setting reasonable boundaries limits its application ~~on~~ to seawater intrusion management. Meanwhile, the method is not suitable ~~to~~ for simultaneously adopting

updated monitoring data and ~~produce~~ producing real-time prediction. Under such circumstances, where the data source is scarce, artificial intelligence technology has been proposed in groundwater dynamic predictions. Artificial neutral network (ANN) has been greatly improved and has ~~became~~ become a robust tool for dealing with groundwater problems~~,~~ where the flow is nonlinear and highly dynamic in nature (Maier and Dandy, 2000). The conventional network model generally has defects such as high computational complexity, slow training speed, and failure in retaining historical information, thus is hard~~ly~~ to be enrolled in the long-term time-series prediction (Cannas et al., 2006; Mei et al., 2017). To solve this problem, researchers upgraded the conventional networks by integrating them with methods like a genetic algorithm (Danandeh Mehr and Nourani, 2017; Ketabchi and Ataie-Ashtiani, 2015), singular spectrum (Sahoo et al., 2017), and wavelet transform (Gorgij et al., 2017; Seo et al., 2015; Zhang et al., 2019). Singular spectrum analysis and wavelet transform can help to preprocess the time-series data before they are put into the neural networks to improve prediction accuracy and efficiency.

With ~~the~~ computing capacity development, deep learning (DL) has emerged as a very powerful time-series prediction method. DL models are particularly suitable for big data time-series~~,~~ because they can automatically extract complex patterns without feature extraction preprocessing steps (Torres et al., 2019). However, the general~~ly~~ fully connected networks are not effective ~~to~~ in capturing~~e~~ the temporal dependence of time ~~-~~series (Senthil Kumar et al., 2005). Therefore, more specialized DL models, such as recurrent neural networks (RNN) (Rumelhart et al., 1986) and convolutional neural

networks (CNN) (Lecun et al., 1998) have been adopted in the field of time-series prediction (Feng et al., 2020). Different from the back-propagation (BP) neural network that the information flows from the input to the output layer in one direction, the RNN preserves the information from the previous step as input to the current step with loops (Coulibaly et al., 2001). This allows the RNN to handle time-series and other sequential data but generally is not straightforward for a long-term calculation in practice (Bengio et al., 1994). Therefore, the enhanced RNN model, long short-term memory (LSTM) is proposed and capable ~~to~~ of processing high variable-length sequences even with millions of data points (Fischer and Krauss, 2018; Kratzert et al., 2019) . As one of the best deep neural network model in time-series predicting, the LSTM has been widely used in the prediction of temporal variations such as stock market predictions (Fischer and Krauss, 2018), rainfall-runoff (Kumar Dubey et al., 2021), and groundwater level (Solgi et al., 2021). Despite ~~of~~ substantial progress~~es~~ in hydrology ~~predicting~~prediction, these networks still have issues of low training efficiency and low accuracy (Zhan et al., 2022).

More recently, a variant of the CNN architecture known as temporal convolutional networks (TCN) has acquired popularity (Bai et al., 2018). The prominent characteristic of TCN is its ability to capture long-term dependencies without information loss (Cao et al., 2021). Meanwhile, it joints a residual block structure to fix the disappearance of the gradient in the deep network structure (Chen et al., 2020). With proper modifications, the TCN is quite genetic and easily to be used to build a very deep and extensive network in sequence modeling. In earth science, the TCN has been

116 successfully applied to time-series prediction tasks, including multivariate time-series
117 predicting for meteorological data (Wan et al., 2019), probabilistic predicting (Chen et
118 al., 2020) and wind speed predicting (Gan et al., 2021). Researchers suggest that the
119 TCN convincingly has the advantage in popular ~~deep~~ deep-learning models across a
120 broad range of sequence modeling tasks (Borovykh et al., 2019; Chen et al., 2020; Wan
121 et al., 2019). Another important subject is that these networks are mostly used to predict
122 variables in only one step, which is not enough for the prediction of hydrology
123 information in management. Researches have ~~been~~ adopted the method to predict the
124 trends of ENSO and sea temperature (Yan et al.,2020; Jian et al., 2021). However, the
125 potential of TCN has not been investigated in the sequencing model of the
126 hydrogeology field. Therefore, it is worthy to explore their prediction abilities in
127 leading periods.

128 The objective of this study is to develop real-time advance prediction climate-
129 ~~dydro~~ hydro hybrid data-driven models of groundwater level in the coastal aquifer
130 based on TCN and LSTM. The hourly processed tidal, precipitation with groundwater
131 level data in monitoring wells of Laizhou Bay are utilized to train the model and predict
132 the groundwater level in a period of 1-day, 3-, 7-, and 15-days. The two models were
133 further compared in ~~the~~ view of accuracy and efficiency. The rest of the paper is
134 organized as follows. Sec~~t~~. 2 introduces the study area and observational data. Sec~~t~~. 3
135 illustrates the detailed concepts of TCN and LSTM, the experimental model settings
136 and model evaluation criteria. Sec~~t~~. 4 presents the ~~predicting~~ predicted results and
137 discussions. Finally, the paper is concluded in Sec~~t~~. 5.

## 2 Study area and data processing

### 2.1 Site description

The study area is located ~~in~~ on the south coast of Laizhou Bay, along the Yangzi to Weifang section in the Shandong province of China (Fig. 1). ~~The~~ The Laizhou Bay ~~is~~ has been one of the earliest and most seriously affected areas by seawater intrusion since the 1970s in China (Han et al., 2014; Zeng et al., 2016). The area is a coastal plain, which contains a series of Cretaceous to modern sediments that cover~~ing~~ the Paleozoic basement. The sedimentary facies of coastal aquifer are alluvium, p~~ro~~luvial, and marine sediments from south to north (Han et al., 2011). According to the research of (Xue et al., 2000), there ~~were~~ have been three seawater intrusion and regression events in the sea area of Laizhou Bay since the upper Pleistocene. The transgression in the early upper Pleistocene formed the third marine aquifer containing sedimentary water. ~~These brine~~This brine ~~were~~was formed by evaporation and concentration of ancient seawater and re-dissolution and mixing of salt (Dai and Samper, 2006; Zhang et al., 2017). The monitoring wells BH01-BH05 are distributed in the study area along a ~~eross~~ cross-section perpendicular to the coastline. Among the wells, ~~the~~ wells BH01 and BH05 have relatively integrated data in time and are distributed ~~in~~ on the two sides of the cross profile with distinguished annual variation patterns, which are selected as examples for the developed models.

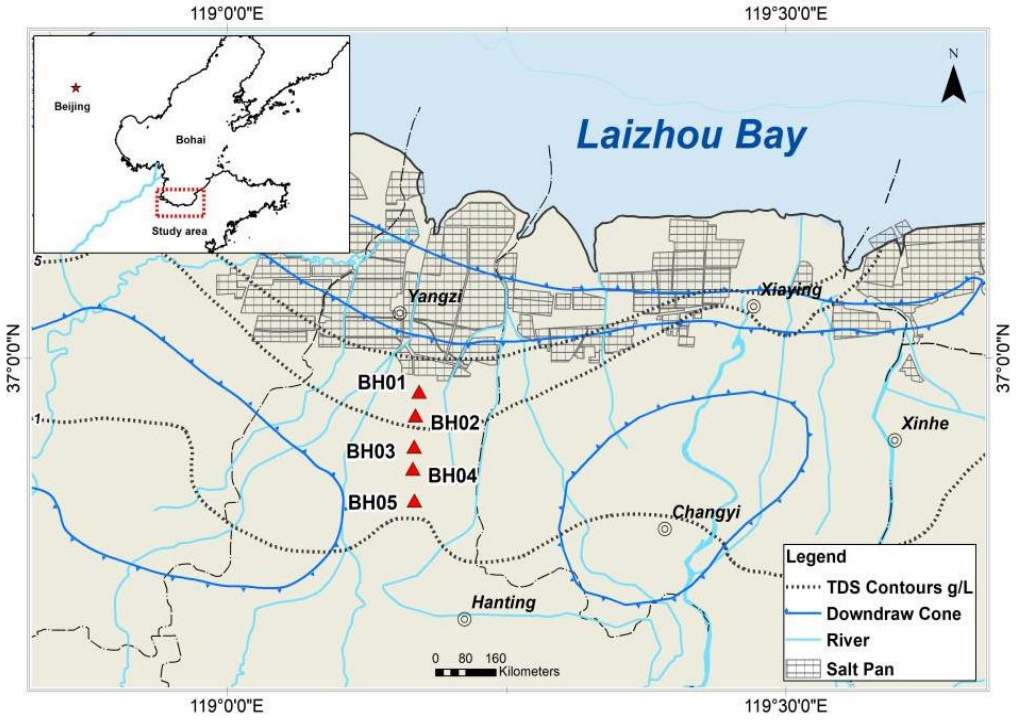


Figure 1. Schematic figure of the study area with monitoring wells BH01-BH05.

## 2.2 Data collection and pre-processing

The precipitation and tidal level are selected as the primary factors to affect the groundwater dynamics in the coastal area. The data in the period of 2011 to 2012 with groundwater level observations of two wells are combined as the input of the deep learning models. A total of 28,836 data items are collected for monitoring wells. ~~and~~ ~~t~~The variations of groundwater level~~,~~ and tidal level with precipitation are shown in Figure 2. The rainfall is concentrated from June to September and in shortage from December to April. The tide in the study area is irregular, mixed with a semi-diurnal variation. In the experiments, ten months of data from October 2011 to July 2012 is first extracted for model training and testing. The rest of the data from August 2012 to October 2012 is used to test model prediction accuracy.

In addition, the magnitudes of meteorological and hydrological variables have

obvious temporal variations. To reduce the negative impact on the model learning ability, especially on the speed of gradient descent, all variables are normalized to ensure that they remain at the same scale (Kratzert et al., 2019). This pre-processing method ensures the stable convergence of parameters in the developed TCN- and LSTM-based models and improve the simulation accuracy of the model. The normalization formula is as follows:

$$x'_{i\,\sout{i}} = \frac{x_i - x_{min}}{x_{max} - x_{min}} \quad (1)$$

where $x_i$ represents the original data in time $i$ ~~and $x_i$ represents the original data in time $i$~~. $x_{max}$ and $x_{min}$ are the maximum and minimum variable values. The output of the network is retransformed to obtain the final groundwater level prediction, which is an inverse data scaling process.

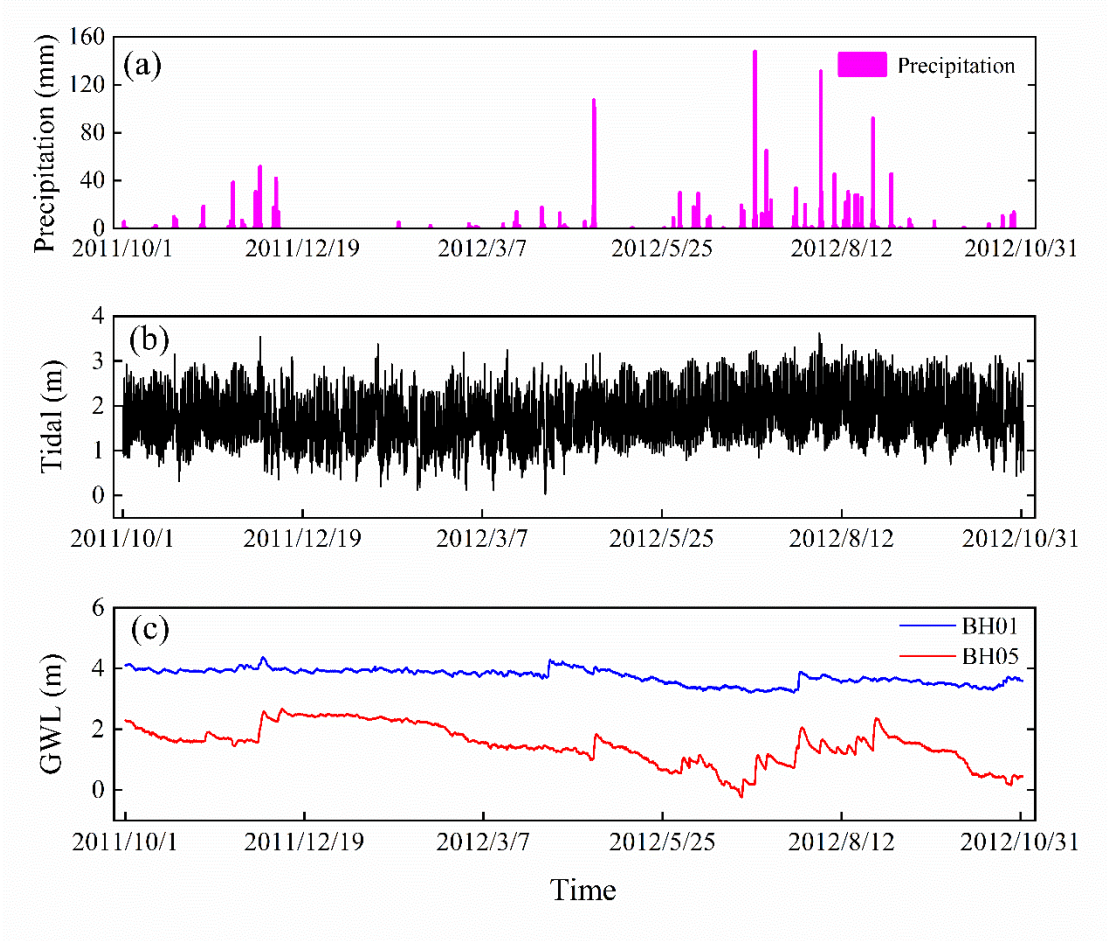


Figure 2. Time-series of the variables in the study, including (a) precipitation, (b) tide, (c) groundwater level (GWL).

# 3 Methodology

## 3.1 Temporal Convolutional Network

The TCN is first proposed for video action segmentation and detection by hierarchically capturing intermediate feature presentations. Then the term is extended for sequential data for a wide family of architectures with generic convolution (Bai et al., 2018; Lea et al., 2017). Suppose that we have an input hydro-climate sequence at different times $x_0$, ..., $x_T$, the goal of the modeling is to predict the corresponding groundwater level as outputs $y_0$, ..., $y_T$ at each time. The problem could transfer to build a network $f$ that minimizes the function loss between observations and actual network

outputs $L[(y_0, ..., y_T),(\hat{y}_0,...,\hat{y}_T)]$, where $\hat{y}_0,...,\hat{y}_T = f(x_0, ..., x_T)$. Currently, a typical TCN consists of dilated, causal 1D full-convolutional layers with the same input and output lengths. With TCN, the prediction $y_t$ depends only on the data from $x_0$ and $x_t$ and does not include the future data from $x_t$ and $x_T$ (Yan et al., 2020). With the three key components of TCN, it has two distinguishing characteristics: (1) the TCN is able to map the same length of output as the input sequence as in RNN; (2) the convolution involved in TCN is causal, eliminating the influence of future information on the output.

3.1.1 Causal Dilated Convolutions

In the TCN, the first advantage is accomplished by a 1D full-convolutional network (FCN) architecture. Different from the traditional CNN, the FCN transforms the fully connected layers into the convolutional layers for the last layers, preserving the same length of output as that of the input (Long et al., 2015). As shown in Fig. 3a, the lengths of the input, the hidden and the output layers are the same in the FCN. Some zero padding is needed in this step by adding additional zero-valued entries with a length of kernel size-1 in each layer. The kernel size is the number of successive elements that are used to produce one element in the next layer.

To avoid ~~the~~ information leakage from the future (after time $t$), the TCN uses causal convolution instead of standard convolution, where only the elements at or before time $t$ in the previous layer are adopted into the mapping of the output at time $t$. Further, the dilated convolution is employed to capture long-term historical information by skipping a given step size (dilation factor $d$) in each layer. For example, the dilation factor $d$ increases from 1 to 4 with the evolution of the network depth ($n$) in an

exponentially increasing pattern. In this way, a very large receiving domain is created and all the historical records in the input can be involved in the prediction model with a deep network.

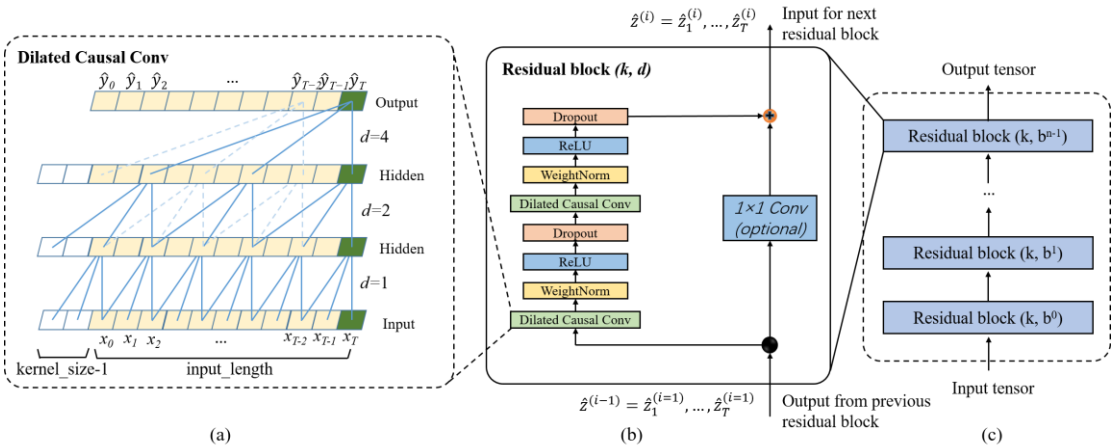


Figure 3. Architectural elements in the proposed TCN. (a) the structure of causal dilated convolution; (b) the TCN residual block. A~~n~~ 1x1 convolution is added when residual input and output have different dimensions; (c) framework of residual connection in the TCN.

3.2.2 Residual Connections

In a high dimensional and long-term sequence, the network structure could be very deep with increasing complicity and cause a vanishing gradient. To solve this issue, a residual block structure is introduced to replace the simple 1D causal convolution layer~~,~~ so that the designed TCN structure is more generic (He et al., 2016). The residual block in a TCN is represented in Fig. 3b. It has two convolutional layers with the same kernel size and dilation factor and non-linearity. To solve non-linear models, the rectified linear unit (ReLU) is added to the top of the convolutional layer (Nair and Hinton, 2010). The weight normalization is applied between the input of hidden layers (Salimans and

Kingma, 2016). Meanwhile, a dropout is added after each dilated convolution for regularization (Srivastava et al., 2014). For all connected inner residual blocks, the channel widths of input and output are consistent. But the width may be different between the input of the first convolutional layer of the first residual block and the output of the second convolutional layer of the last residual block. Therefore, a 1×1 convolution is added in the first and last residual block to adjust the dimensions of the residual tensor ~~in~~to the same. The output of the residual block is represented by $\hat{Z}^{(i)}$ for the $i^{\text{th}}$ block.

3.2.3 Structure of TCN

A complete structure of TCN is illustrated in Fig.3c. It contains a series of proceeding residual blocks. The structural characteristics make TCN a deep learning network model very suitable for complex time-series prediction problems (Lara-Benítez et al., 2020). The main advantage of TCN is that, similar to RNN, they have flexible receptive fields and can deal with various length inputs by sliding one-dimensional causal convolution kernel. Furthermore, because TCN shares a convolution kernel and has parallelism, it can process long sequences in parallel instead of sequential processing like RNN, so it has lower memory usage and shorter computing time than a cyclic network. Moreover, RNN often has the problems of gradient disappearance and gradient explosion, which are mainly caused by sharing parameters in different periods, while TCN uses a standard backpropagation-through-time algorithm (BPTT) for training, so there is little gradient disappearance and explosion problem (Pascanu et al., 2012). The detailed mathematical calculation and

associated information of the TCN architecture are referred to (Bai et al., 2018).

### 3.2 Long Short-Term Memory network

LSTM is a special RNN model explicitly designed for long-term dependence problems. As shown in Fig. 4a, the RNN has a series of repeating modules that are recursively connected in the evolution direction of the sequence. The chain-like structure permits the RNN to retain important information in a "tanh" layer and produce the same length of output $\hat{y}_t$ as input $x_t$. However, the short-length "remember time" is not enough for ~~the~~ groundwater prediction. Especially for our hourly recorded data, a maximum step of about ten reported by Bengio et al. (1994) is unable to count the effect of annual~~ly~~, seasonal~~ly~~, and even daily groundwater variation. Different from the simple layer in the RNN, the LSTM has a more complicated repeating module with four interacting layers.

The core idea of LSTM is the special structure to control the cell state in the module as shown in Fig. 4b. It includes a cell and an input gate $i_t$, a forget gate $f_t$, and an output gate $o_t$. The information can directly flow down along cells $C$ without critical changes, therefore, preserving long-term history messages (Zhang et al., 2018b). The three gates control which data in a sequence is important to keep or throw away, and protect the relevant information passed down in the cell to make predictions. The forget gate $f_t$ has a sigmoid layer to determine which information is discarded with a value between 0 and 1. The lower the value, the less ~~the~~ information is added to the cell state (Ergen and Kozat, 2018). Opposite the forget gate, the input gate $i_t$ decides what information to retain in the cell state. It is composed of two parts: a sigmoid layer and

a tanh layer. The two layers are combined to govern which values will be updated by generating a new candidate value $\tilde{C}_t$. The old cell state $C_{t-1}$ can then ~~can~~ be updated into the new cell state $C_t$ with a weighted function. Finally, the output gate $o_t$ determines what parts of the cell state should be passed on to the next hidden state. The detailed calculation of the LTSM can be referenced to (Lea et al., 2016).

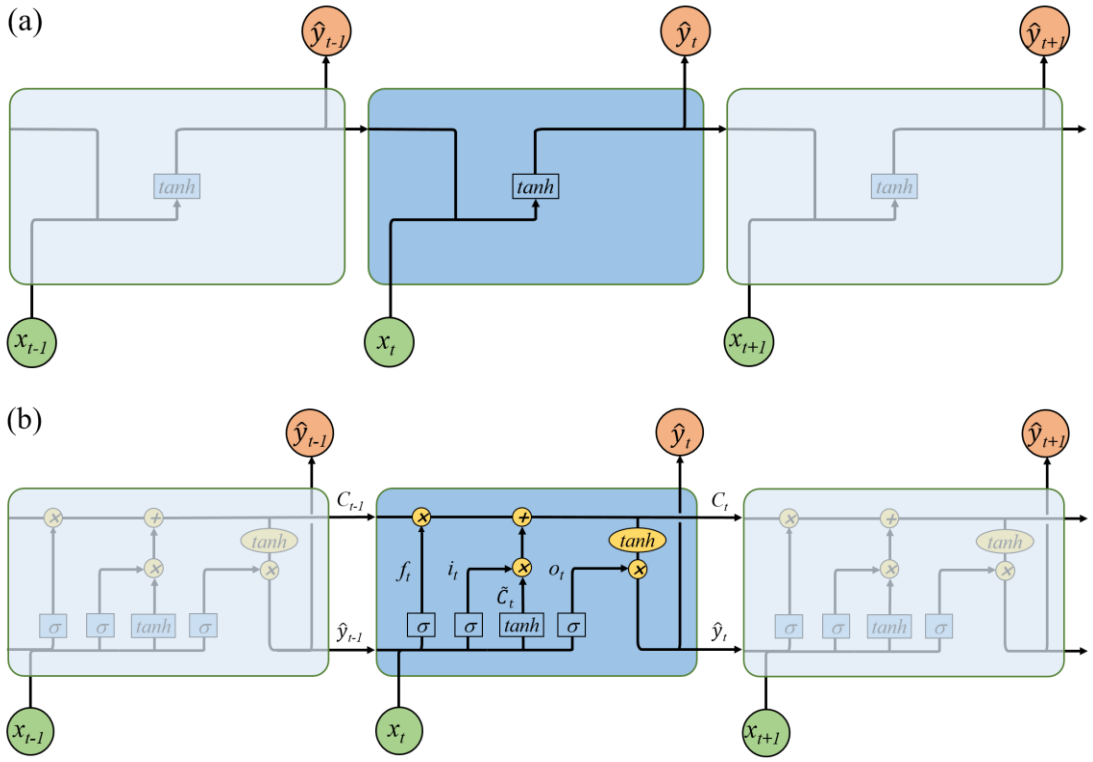


Figure 4. Graphical representation for (a) ~~chain~~ chain-like structure of the RNN by assigning $x_t$ and $\hat{y}_t$ as input and output. The self-connected hidden units allow information to be passed from one step to the next; (b) LSTM's memory block based on RNN. The hidden block includes three gates (input $i_t$, forget $f_t$, output $o_t$) and a cell state to select and pass the historical information.

### 3.3 Experimental study

The TCN- and LSTM-based models were developed separately for monitoring wells BH01 and BH05. Due to the high complexity of the DL models, setting

appropriate hyper-parameters for the developed networks is very important. Here, the impact of the size of the input window, the epoch number, and the batch size ~~were~~ was tested with different convolutional architectures over the monitoring data (Lara-Benítez et al., 2020). The learning dataset is first divided into two parts: 80% of the time-series data is used as the training set, and 20% of the data is utilized as the testing set. The effect of different splitting strategies is further tested in section 4. With the increase of the epoch numbers, the curve gradually approaches ~~to~~ the optimal fitting state from the initial non-fitting state, but too many epochs frequently lead to over-fitting of the neural network (Afaq and Rao, 2020). Meanwhile, the number of iterations generally increases for updating weights in the neural network. Therefore, the number of epochs from 0 to 300 is evaluated. Batch size represents the number of samples between model weight updates (Kreyenberg et al., 2019). The value of the batch size often is set between 1 and hundreds. A ~~Larger~~ larger batch size often leads to faster convergence of the model~~,~~ but may lead to less ideal of the final weight set. To find the best balance between memory efficiency and capacity, the batch size should be carefully set to optimize the performance of the network model. Besides these parameters, the number of filters in the TCN-based and the hidden nodes in the LSTM-based model ~~were~~ was as well tested within reasonable ranges.

The 1-day, 3-, 7-, and 15-days leading prediction experiments were further conducted to test the capacity of DL methods in predicting long-term groundwater levels in the coastal aquifer. To eliminate the randomness of model training, all experiments were repeated 5 times and the average values of each index were compared.

In all experiments, the average absolute error (MAE) has been used as the loss function of networks (Lara-Benítez et al., 2020). The Adam optimizer has an adaptive learning rate, which can improve the convergence speed of deep networks, which has been used to train the models (Kingma and Ba, 2015).

### 3.4 Evaluation of model performance

Two evaluation metrics, coefficient of determination ($R^2$) and root mean square error (RMSE) are selected to quantify the goodness-of-fit between model outputs and observations (Zhang et al., 2020). The two criteria are calculated using the following equations:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(h_i - y_i)^2} \quad (1)$$

$$R^2 = \frac{\sum_{i=1}^{N}(h_i-\bar{h})^2-\sum_{i=1}^{N}(h_i-y_i)^2}{\sum_{i=1}^{N}(h_i-\bar{h})^2} \quad (2)$$

where $h_i$ is the observed groundwater level at time $i$, $y_i$ is the network prediction values at time $i$, $\bar{h}$ is the mean of the observed groundwater levels, and $n$ is the number of observations. RMSE measures the prediction precision which creates a positive value by squaring the errors. The RMSE score is between [0, ∞]. If the RMSE approaches ~~to~~ 0, the model prediction is ideal. $R^2$ measures the degree of model replication results, ranging between [−∞, 1]. For the optimal model prediction, the score of $R^2$ is close to 1.

## 4 Results and discussions

### 4.1 Hyper-parameter trial experiments

4.2.1 Experiments of the TCN-based model

–The TCN-based model was built on the Keras platform using TensorFlow of

python as the backend. ~~Taken~~ Taking the groundwater level prediction data set in well BH01 as an example, the trials were set up with a ~~variety combination of different hyper-parameters~~variety combination of different hyper-parameters in the TCN-based model as illustrated in Table 1. With the fixed number of ~~epoch~~epochs, the simulation results of 32 filters were better than that of 16 and 64 filters. Meanwhile, under the condition of 32 filters, the accuracy of the model decreased with the increasing ~~of~~ batch size. The results of the 16 batch size were~~–~~ better than that of 32 and 64 batches. Based on the above experimental results, the influence of different numbers of epochs on the simulation was further explored with the filters was 32 and the batch size was 16, as shown in Fig. 5. The overall results of the model were improved when the number of ~~epoch~~epochs increased from 100 to 190 though the variation was not strictly linear, and the variations became stable with minor fluctuations when the number of epoch exceeded 200.

Table 1. The RMSE and $R^2$ values between the observed and predicted groundwater levels in well BH01 with different numbers of epochs, different numbers of filters, and different batch sizes. The bold values represent the optimal hyper-parameters with the smallest RMSE and the highest $R^2$ scores in the TCN-based model.

| Epoch | Filters | Batch size | RMSE(m) | $R^2$ | Time(min) |
|---|---|---|---|---|---|
| 100 | 32 | 16 | 0.0182 | 0.9904 | 1.29 |
| | | 32 | 0.0117 | 0.9876 | 1.05 |
| | | 64 | 0.0117 | 0.9875 | 0.78 |
| 200 | 16 | 16 | 0.0078 | 0.9946 | 2.41 |
| | | 32 | 0.0068 | 0.9959 | 1.75 |

| | | | | | |
|---|---|---|---|---|---|
| | | 64 | 0.0090 | 0.9942 | 1.19 |
| 200 | 32 | 16 | 0.0059 | 0.9970 | 2.58 |
| | | 32 | 0.0075 | 0.9948 | 2.01 |
| | | 64 | 0.0082 | 0.9938 | 1.51 |
| 200 | 64 | 16 | 0.0125 | 0.9906 | 3.68 |
| | | 32 | 0.0101 | 0.9907 | 3.21 |
| | | 64 | 0.0157 | 0.9775 | 2.76 |
| 300 | 32 | 16 | 0.0065 | 0.9955 | 3.8 |
| | | 32 | 0.0076 | 0.9946 | 3.01 |
| | | 64 | 0.0099 | 0.9904 | 2.22 |

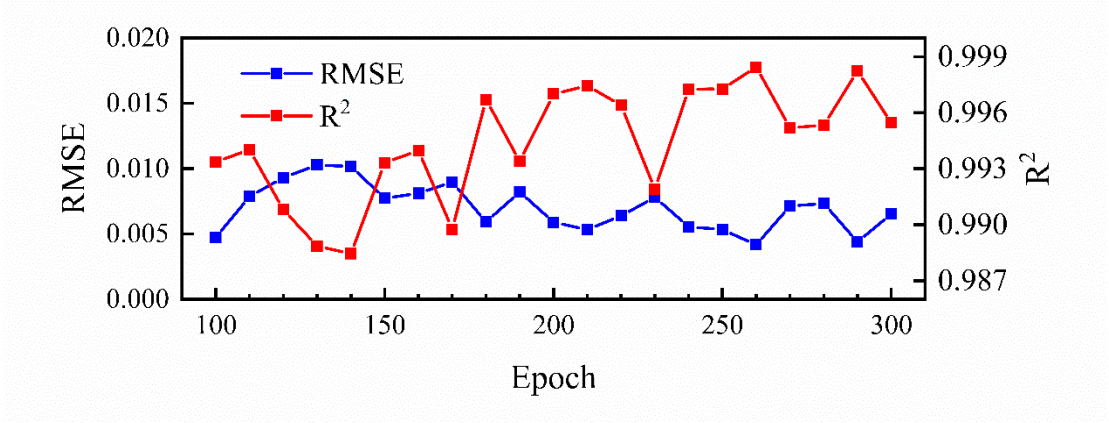


Figure 5. The variation of RMSE and $R^2$ values between the observed and predicted groundwater levels of well BH01 with the increasing number of the epoch when the number of filters is 32 and the batch size is 16.

4.2.2 Experiments of the LSTM-based model

The number of epochs and hidden nodes are two key parameters affecting the simulation accuracy of LSTMs (Zhang et al., 2018a). Different hyper-parameter combinations were tested as well as in the proposed TCN-based model with groundwater levels in well BH01. The RMSE, $R^2$ and running time are shown in Table 2. With a fixed number of hidden nodes, the results of 100 and 200 epochs were better

than that in the 300 epochs experiment. A detailed variation of RMSE and $R^2$ values with the increasing number of hidden nodes and epochs is further illustrated in Fig. 6. The figure shows that the RMSE and $R^2$ have a decreasing and increasing trend separately when the number of epochs is greater than 150, but is reversed when it is larger than 240. The variations of RMSE and $R^2$ with increasing hidden nodes have similar changes as shown in Table 2. Though an insufficient number of neurons may decrease the learning ability of the network, the results indicate that ~~an~~ increasing training hyper-parameters may not be necessary to ensure better prediction.

Table 2. The RMSE and $R^2$ values between the observed and predicted groundwater levels in well BH01 with different numbers of epochs and hidden nodes. The bold values represent the optimal hyper-parameters used in the proposed LSTM-based model.

| Epoch | Hidden nodes | RMSE | $R^2$ | Time(min) |
|---|---|---|---|---|
| 100 | 50 | 0.0104 | 0.9902 | 1.01 |
| | 60 | 0.0098 | 0.9916 | 1.38 |
| | 70 | 0.0095 | 0.9922 | 1.53 |
| | 80 | 0.01 | 0.9913 | 1.75 |
| **200** | 50 | 0.0094 | 0.9922 | 1.91 |
| | 60 | 0.0089 | 0.9931 | 2.59 |
| | **70** | **0.0088** | **0.9932** | **2.96** |
| | 80 | 0.0092 | 0.9925 | 3.28 |
| 300 | 50 | 0.0101 | 0.9903 | 2.86 |
| | 60 | 0.0105 | 0.9901 | 3.85 |
| | 70 | 0.0103 | 0.9907 | 4.29 |
| | 80 | 0.0120 | 0.9872 | 4.92 |

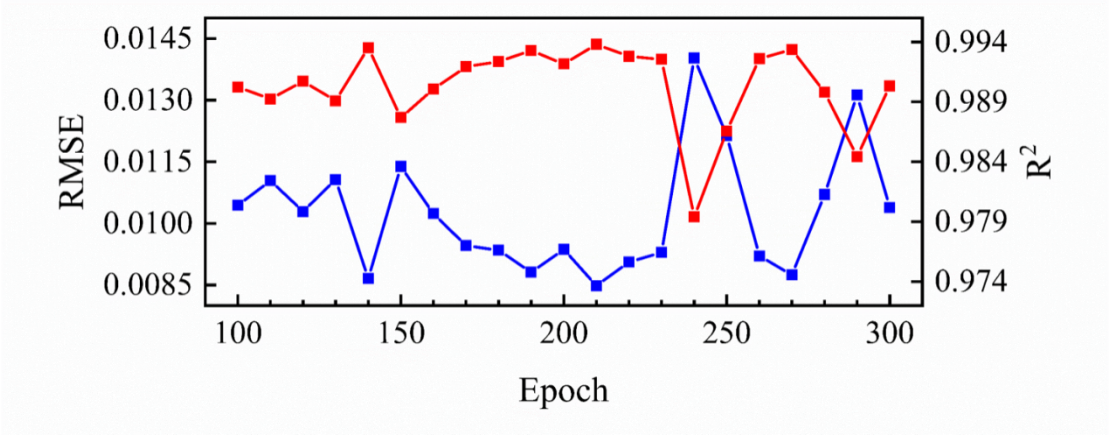


Figure 6. The variation of RMSE and $R^2$ values between the observed and predicted groundwater levels of well BH01 with the increasing of the number of epochs when the hidden node is 50.

The trial experimental results present a similar fitting pattern shared by the two kinds of networks. The growing value of parameters dramatically increases the computational cost in the network. For example, the time cost from 50 to 80 hidden nodes has increased about 1.7 times in each iteration trial in the LSTM-based model. Finally, 200 epochs, 32 filters, and ~~the~~ 16 batch size were chosen as the optimal parameters in the TCN network. For the LSTM network, the number of epochs and hidden nodes were chosen as 200 and 70.

### 4.3 Model performance and evaluation

The optimal hyper-parameters of the proposed TCN-based model for groundwater level predicting are shown in Table 1 (epoch = 200, filters = 32 and batch size = 16). Besides that, the kernel size in each convolutional layer is set as 6, and the dilations are [1,2,4,8]. For the LSTM-based model, the batch size is set to 148 with epoch=200 and nodes=70. The same hyper-parameters are then utilized to construct TCN and LSTM architectures for the prediction of groundwater level in different

monitoring wells.

The ~~one~~ one-~~step~~ step-ahead simulated groundwater level in the training and testing~~,~~ and prediction stages by the two models are shown in Fig. 7. For both models, the simulated values completely capture the variation of groundwater levels in monitoring wells with the overlapped plots. The $R^2$ and RMSE values of simulation results are listed in Table 3. In the prediction stage, the values of RMSE are 0.0019 and 0.0166 for BH01 and BH05, and the values of $R^2$ are larger than 0.999 in the prediction for the TCN-based model. For the LSTM-based model, the RMSE values are 0.0074 and 0.0588, and the $R^2$ values are 0.9957 and 0.9980. These metrics indicate that both models can "remember" the historical records and produce true observations. The simulation accuracy of TCN-based models is slightly higher than the LSTM-based models. In addition, the running time of the TCN-based model is 2.6 minutes, which is faster than that of the TCN-based model.

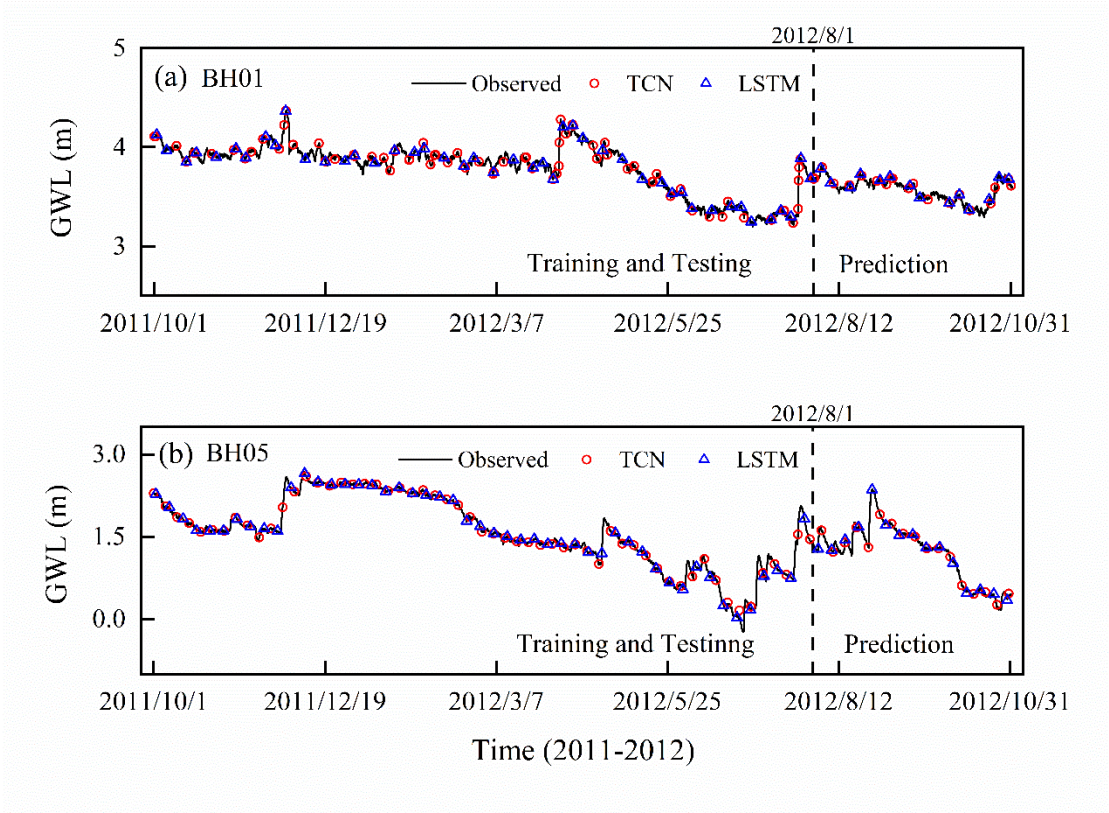


Figure 7. The simulation results of groundwater level of monitoring wells BH01 and BH05 by TCN-based model. The black dash line divides the data into two groups: the training and testing dataset, and the prediction dataset.

Table 3. The model results for groundwater level in the training and testing and prediction stage

| Well | Model | Training and Testing | | | Prediction | | |
|---|---|---|---|---|---|---|---|
| | | MAE | RMSE | $R^2$ | MAE | RMSE | $R^2$ |
| BH01 | TCN | 0.0017 | 0.0068 | 0.9992 | 0.0009 | 0.0019 | 0.9997 |
| | LSTM | 0.0053 | 0.0077 | 0.9990 | 0.0050 | 0.0074 | 0.9957 |
| BH05 | TCN | 0.0070 | 0.0279 | 0.9981 | 0.0061 | 0.0166 | 0.9990 |
| | LSTM | 0.0082 | 0.0116 | 0.9997 | 0.0168 | 0.0558 | 0.9980 |

### 4.4 Long term leading time prediction

The TCN- and LSTM-based models were further adjusted to predict the groundwater levels over three months ahead with different leading periods. Prediction results with 1-day, 3-, 7-, and 15-days leading time with TCN- and LSTM-based models are illustrated in Fig. 8 and Fig. 9 for wells BH01 and BH05, respectively. The results show that the predicted groundwater values have the same change trend as the actual groundwater level in monitoring wells. Both of the models are able to capture the variation trend of groundwater levels with longer leading periods of more than ~~one time~~one-time step in the two monitoring wells.

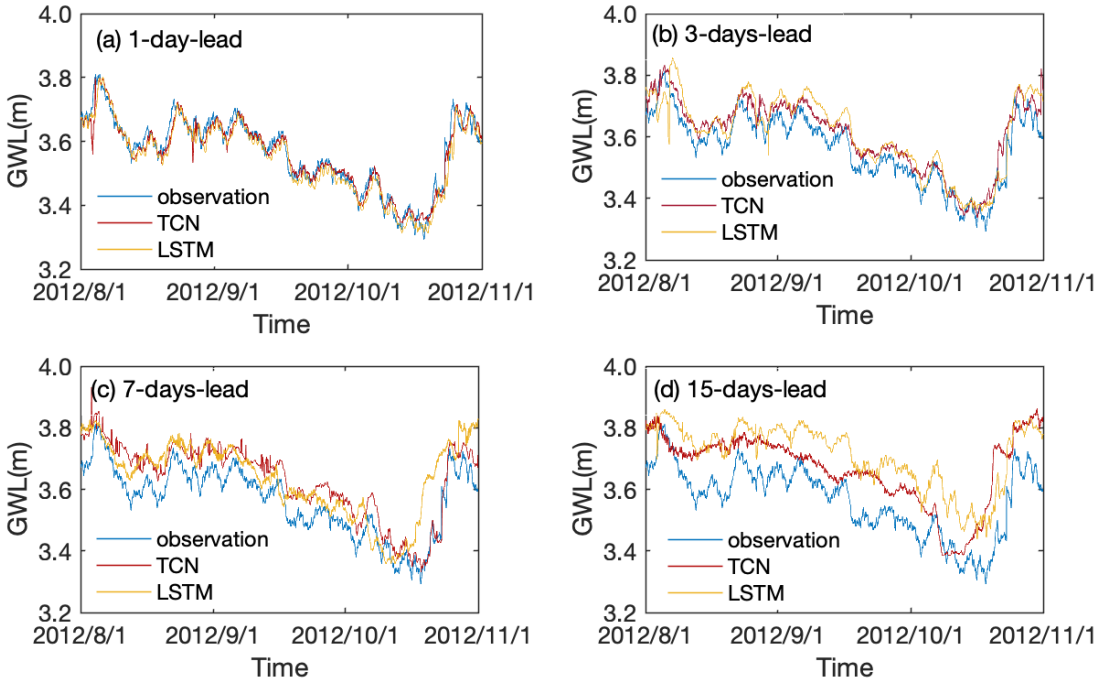


Figure 8. The observed and predicted values of the groundwater level with TCN- and LSTM-based models for 1-day, 3-, 7- and 15-days lead period in monitoring well BH01.

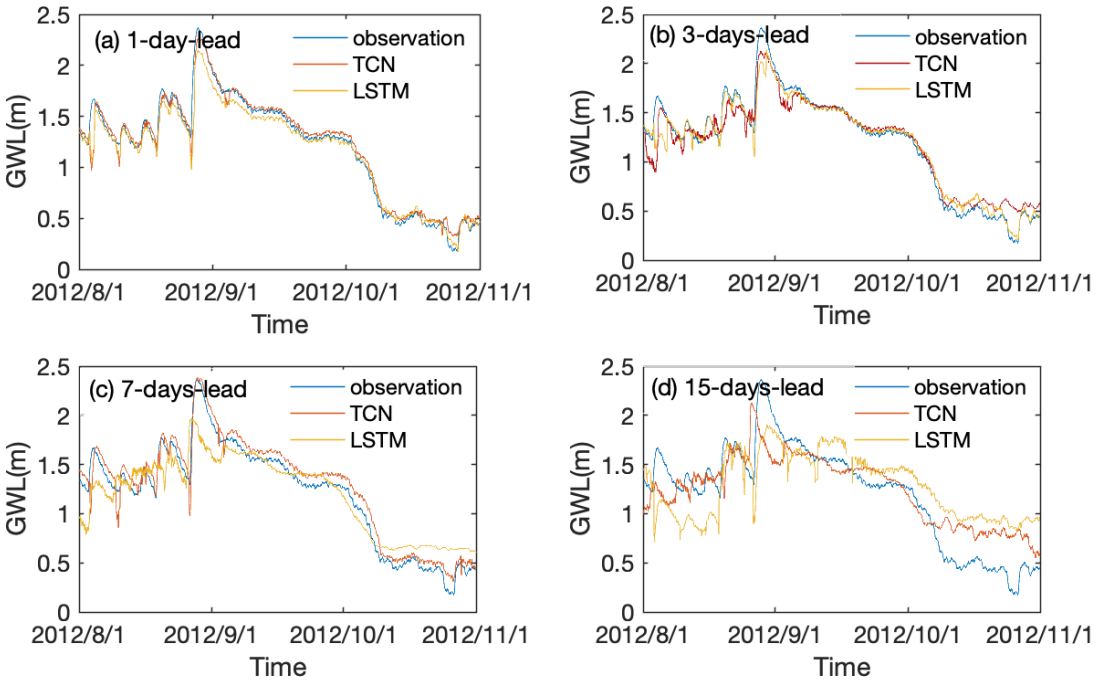


Figure 9. The observed and predicted values of the groundwater level with TCN- and LSTM-based models for 1-day, 3-, 7-, and 15-days leading period in monitoring well BH05.

To quantitatively compare the prediction accuracy of the proposed TCN- and LSTM-based models, the results of two evaluation metrics with the model running time are summarized in Table 4. It can be learned that the $R^2$ value of TCN-based models decreased from 0.9386 to -0.1407 for well BH01 and from 0.9670 to 0.7271 for well BH05. Correspondingly, an increase of RMSE values from 0.028 to 0.1209 and 0.0934 to 0.206 are observed for BH01 and BH05, separately. A similar variation pattern is recognized for the LSTM-based model with smaller $R^2$ and higher RMSE than that of the TCN-based model. Notably, the running time of advance prediction is much longer than that of ~~single~~ single-step prediction. Meanwhile, with the increasing of the leading period, the time had been raised nonlinearly. Further, in this process, the TCN-based model costs longer time than that of the LSTM-based model.

Table 4. The model results for groundwater level in the ~~long~~ long-term prediction

| Well | Model | Prediction | | Mins | Model | Prediction | | Mins |
|---|---|---|---|---|---|---|---|---|
| | | RMSE | $R^2$ | | | RMSE | $R^2$ | |
| BH01 | TCN-1 | 0.0280 | 0.9386 | 5.38 | LSTM-1 | 0.0349 | 0.9047 | 3.76 |
| | TCN-3 | 0.0550 | 0.7638 | 16.1 | LSTM-3 | 0.0640 | 0.6802 | 11.01 |
| | TCN-7 | 0.0741 | 0.5713 | 34.3 | LSTM-7 | 0.0956 | 0.2874 | 26.27 |
| | TCN-15 | 0.1209 | -0.1407 | 94.95 | LSTM-15 | 0.1486 | -0.7227 | 85.13 |
| BH05 | TCN-1 | 0.0934 | 0.9670 | 5.19 | LSTM-1 | 0.1012 | 0.9613 | 3.78 |
| | TCN-3 | 0.1375 | 0.9285 | 16.18 | LSTM-3 | 0.1086 | 0.9554 | 11.4 |
| | TCN-7 | 0.1084 | 0.9296 | 35.44 | LSTM-7 | 0.2050 | 0.8406 | 26.2 |
| | TCN-15 | 0.2060 | 0.7271 | 80.46 | LSTM-15 | 0.3515 | 0.5330 | 73.45 |

The performance of the two networks was further evaluated with Taylor diagrams by taking different criteria aspects, ~~which~~ including standard deviation (SD), correlation coefficient (COR), and root mean square deviation (RMSD) into account (Taylor, 2001). The comparisons of the TCN- and LSTM-based models are shown in Fig. 10. As the metrics are distributed away from the reference point (Ref), the deviation of prediction from observation is generally increased with extending of the leading period. ~~Taken~~ Taking well BH05 for example, the prediction with 1-day (24 hours prediction window) in advance ~~are~~ is the highest in agreement with the actual situation in the two models. The 1-day leading prediction results have the lowest RMSD values and highest $R^2$ values for both models. The prediction precision gradually decreases with the ~~extending~~ extension of leading time to 3-days, 7-days, and 15-days. For well BH01, an ~~out~~ out-~~of~~ of-trending point is observed. The 15-day~~s~~ prediction results of the LSTM-based model ~~is~~ are closer to the Ref point compared with the TCN-based model. The reason is that the simulation data is highly correlated with observations as shown in Fig. 8.

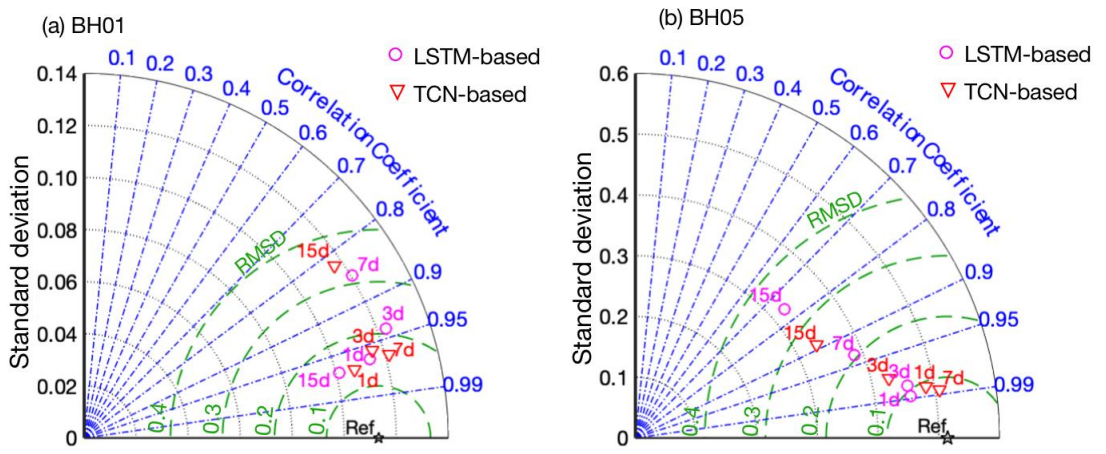


Figure 10. Taylor diagrams with statistical (SD, COR, RMSD) comparison results of the TCN-based and LSTM-based models for well (a) BH01, (b) BH05.

Overall, the TCN- and LSTM-based models both have strong prediction ~~ability~~ abilities in ~~long~~ long-term hydrological time series data. Both models are able to provide accurate predictions once they are trained. The simulation accuracy of the TCN-based model is slightly better than that of the LSTM-based model in the three months prediction but the difference is not significant, with $p>0.05$ in the t-test. The causal dilated convolutions used by TCNs are ~~proved~~ proven to be good at capturing long-term dependencies of time series data. Meanwhile, the model precision decreases and the running time increases with the raising leading period. The processing speed of parallel convolution TCN-based models for long input sequences is slower than that of recurrent networks. This seems to be a shortage in real-time monitoring and early warning. A leading period shorter than 7 days is recommended to ensure both ~~of~~ the accuracy and efficiency of the models in real-time monitoring and early warning.

From the groundwater level variation, significant groundwater decreasing trends can be observed in the irrigation season from March to June. Therefore, ~~the~~ human

activities, such as groundwater pumping are potential reasons for groundwater level change in the study area. Here, the groundwater levels were predicted based on the available data of precipitation and tide. If the pumping data is available and considered in the models, the prediction precision would be enhanced in the models.

### 4.5 Influence of training set percentage

The data-driven methods are supported by data; however, how much data is needed to build an effective model is still a challenging problem (Reichstein et al., 2019). This is because specific problems depend on application cases, data features, and model features (Wunsch et al., 2021). Here we discuss the effect of the training set percentage on the TCN- and LSTM-based models. In our study, the data is the hourly-monitored data from 2011 to 2012. From 2011, we set 20%, 30% to 90% training sets in turn, so as to gradually expand the length of the training set.

Fig. 11 shows the effect of the increased percentage of the training set on the performance of the model. All experiments were repeated five times, and the average values of each index were compared. It can be seen that the performance of the TCN-based model improved with the increase of the percentage of the training set. When the training set reached 80%, the performance was relatively optimal, and then the performance began to deteriorate with the increase of the percentage of the training set. At the same time, it can be seen that the performance of the LSTM-based model tends to be stable when the training set reaches 70%, and then decreases slightly with the increase of the training set. Therefore, a training set evaluation is recommended before the training and testing. We should carefully evaluate and shorten the training data set

as much as possible when necessary. Finally, we set 80% of the training set length to simulate the coastal aquifer time-series data.

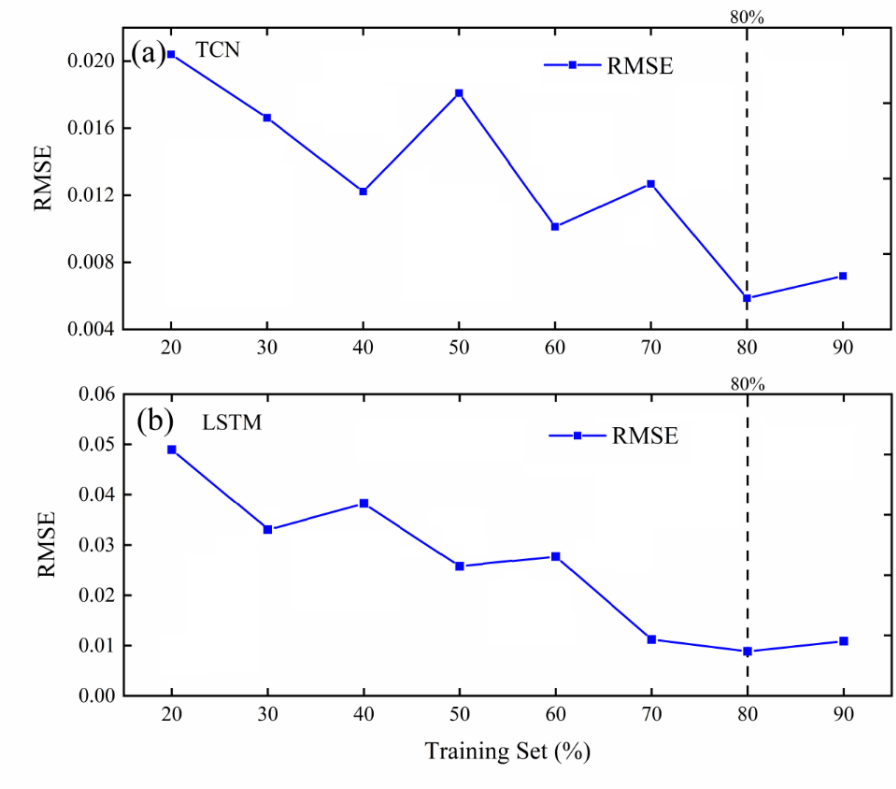


Figure 11. Influence of training set percentage on the performance of the model for (a) BH01 and (b) BH05.

## 5 Conclusions

The TCN- and LSTM-based deep learning models were proposed in this paper to predict groundwater levels in a coastal aquifer. Hyper-parameter searches ~~was~~ were first conducted to obtain good architecture configurations. The results indicated that a deeper, broader model does not necessarily guarantee better predictions. The optimal configurations were then ~~were~~ adopted for the networks of all monitoring data. Both ~~of~~ the TCN- and LSTM- based models well captured the fluctuation of groundwater levels and achieved satisfactory performance on the prediction. Meanwhile, a decreasing

precision is revealed when the leading time increases in advance prediction. In view of accuracy, the TCN-based model outperforms the LSTM-based model but is less ~~efficiency~~ efficient in long-term simulation. Thus, both models can ~~bu~~ be used as a promising method for time-series prediction of hydrogeological data, especially when the regional data is difficult to collect in a complex system.

**Acknowledgements**

This work was jointly supported by National Natural Science Foundation of China (No: 41702244), the Program for Jilin University (JLU) Science and Technology Innovative Research Team (No. 2019TD-35).

**Code availability**

The pieces of code that were used for all analyses are available from the authors upon request.

**Data availability**

The data sets that have been analyzed in this paper are available from the authors upon request.

**Author contribution**

XZ drafted the manuscript and revised the manuscript. GC designed the experiments and collected all the data. DF developed the model code and performed the simulations. ZD was responsible for the project design, oversaw the analysis, and conducted manuscript revision as the project leader and the senior scientist.

**Competing interests**

The authors declare that they have no conflict of interest.

## Reference


Abdalla, O. A. and Al-Rawahi, A. S.: Groundwater recharge dams in arid areas as tools for aquifer replenishment and mitigating seawater intrusion: example of AlKhod, Oman, Environ. Earth Sci., 69, 1951-1962, 2013.

Adam, D. K. J. B.: A method for stochastic optimization in: 3rd International Conference on Learning Representations, 2015.

Afaq, S. and Rao, S.: Significance of epochs on training a neural network, International Journal of Scientific & Technology Research, 9, 485-488, 2020.

Baena-Ruiz, L., Pulido-Velazquez, D., Collados-Lara, A.-J., Renau-Pruñonosa, A., and Morell, I.: Global assessment of seawater intrusion problems (status and vulnerability), Water Resour. Manag., 32, 2681-2700, 2018.

Bai, S., Kolter, J. Z., and Koltun, V.: An empirical evaluation of generic convolutional and recurrent networks for sequence modeling, arXiv preprint arXiv:1803.01271, 2018.

Barlow, P. M. and Reichard, E. G.: Saltwater intrusion in coastal regions of North America, Hydrogeol. J., 18, 247-260, 2010.

Batelaan, O., De Smedt, F., and Triest, L.: Regional groundwater discharge: phreatophyte mapping, groundwater modelling and impact analysis of land-use change, J. Hydrol., 275, 86-108, 2003.

Bengio, Y., Simard, P., and Frasconi, P.: Learning long-term dependencies with gradient descent is difficult, IEEE Trans. Neural Netw., 5, 157-166, 1994.

Borovykh, A., Bohte, S., and Oosterlee, C. W.: Dilated convolutional neural networks for time series forecasting, J. Comput. Financ., 2018.

Cannas, B., Fanni, A., See, L., and Sias, G.: Data preprocessing for river flow forecasting using neural networks: wavelet transforms and data partitioning, Phys. Chem. Earth, 31, 1164-1171, 2006.

Cao, Y., Ding, Y., Jia, M., and Tian, R.: A novel temporal convolutional network with residual self-attention mechanism for remaining useful life prediction of rolling bearings, Reliab. Eng. Syst. Saf., 215, 107813, 2021.

Chen, Y., Kang, Y., Chen, Y., and Wang, Z.: Probabilistic forecasting with temporal convolutional neural network, Neurocomputing, 399, 491-501, 2020.

Coulibaly, P., Anctil, F., Aravena, R., and Bobée, B.: Artificial neural network modeling of water table depth fluctuations, Water Resour. Res., 37, 885-896, 2001.

Dai, Z. and Samper, J.: Inverse modeling of water flow and multicomponent reactive transport in coastal aquifer systems, J. Hydrol., 327, 447-461, 2006.

Dai, Z., Xu, L., Xiao, T., McPherson, B., Zhang, X., Zheng, L., Dong, S., Yang, Z., Soltanian, M. R., and Yang, C.: Reactive chemical transport simulations of geologic carbon sequestration: Methods and applications, Earth-Sci. Rev., 208, 103265, 2020.

Dubey, A. K., Kumar, A., García-Díaz, V., Sharma, A. K., and Kanhaiya, K.: Study and analysis of SARIMA and LSTM in forecasting time series data, Sustain. Energy Technol. Assess., 47, 101474, 2021.

Ergen, T. and Kozat, S. S.: Efficient online learning algorithms based on LSTM neural networks, IEEE Trans. Neural Netw. Learn. Syst., 29, 3772-3783, 2017.

579 Feng, N., Geng, X., and Qin, L.: Study on MRI medical image segmentation technology
580 based on CNN-CRF model, IEEE Access, 8, 60505-60514, 2020.
581 Fischer, T. and Krauss, C.: Deep learning with long short-term memory networks for
582 financial market predictions, Eur. J. Oper. Res., 270, 654-669, 2018.
583 Gan, Z., Li, C., Zhou, J., and Tang, G.: Temporal convolutional networks interval
584 prediction model for wind speed forecasting, Electr. Power Syst. Res., 191, 106865,
585 2021.
586 Garza-Díaz, L. E., DeVincentis, A. J., Sandoval-Solis, S., Azizipour, M., Ortiz-Partida,
587 J. P., Mahlknecht, J., Cahn, M., Medellín-Azuara, J., Zaccaria, D., and Kisekka, I.:
588 Land-use optimization for sustainable agricultural water management in Pajaro
589 Valley, California, J. Water Resour. Plan. Manage.-ASCE, 145, 05019018-
590 05019018, 2019.
591 Gorgij, A. D., Kisi, O., and Moghaddam, A. A.: Groundwater budget forecasting, using
592 hybrid wavelet-ANN-GP modelling: a case study of Azarshahr Plain, East
593 Azerbaijan, Iran, Hydrol. Res., 48, 455-467, 2017.
594 Han, D., Kohfahl, C., Song, X., Xiao, G., and Yang, J.: Geochemical and isotopic
595 evidence for palaeo-seawater intrusion into the south coast aquifer of Laizhou Bay,
596 China, Appl. Geochem., 26, 863-883, 2011.
597 Han, D., Song, X., Currell, M. J., Yang, J., and Xiao, G.: Chemical and isotopic
598 constraints on evolution of groundwater salinization in the coastal plain aquifer of
599 Laizhou Bay, China, J. Hydrol., 508, 12-27, 2014.
600 He, K., Zhang, X., Ren, S., and Sun, J.: Deep residual learning for image recognition,
601 Proc. IEEE, 770-778,
602 Huang, F.-K., Chuang, M.-H., Wang, G. S., and Yeh, H.-D.: Tide-induced groundwater
603 level fluctuation in a U-shaped coastal aquifer, J. Hydrol., 530, 291-305, 2015.
604 Jiang, Y., Zhao, M., Zhao, W., Qin, H., Qi, H., Wang, K., and Wang, C.: Prediction of
605 sea temperature using temporal convolutional network and LSTM-GRU network,
606 Complex Engineering Systems, 1, -, 2021.
607 Ketabchi, H. and Ataie-Ashtiani, B.: Evolutionary algorithms for the optimal
608 management of coastal groundwater: A comparative study toward future challenges,
609 J. Hydrol., 520, 193-213, 2015.
610 Kratzert, F., Herrnegger, M., Klotz, D., Hochreiter, S., and Klambauer, G.:
611 NeuralHydrology–interpreting LSTMs in hydrology, in: Explainable AI:
612 Interpreting, explaining and visualizing deep learning, Springer, 347-362, 2019.
613 Kreyenberg, P. J., Bauser, H. H., and Roth, K.: Velocity field estimation on density-
614 driven solute transport with a convolutional neural network, Water Resour. Res., 55,
615 7275-7293, 2019.
616 Lara-Benítez, P., Carranza-García, M., Luna-Romera, J. M., and Riquelme, J. C.:
617 Temporal convolutional networks applied to energy-related time series forecasting,
618 applied sciences, 10, 2322, 2020.
619 Lea, C., Vidal, R., Reiter, A., and Hager, G. D.: Temporal convolutional networks: A
620 unified approach to action segmentation, European conference on computer vision,
621 47-54,
622 Lea, C., Flynn, M. D., Vidal, R., Reiter, A., and Hager, G. D.: Temporal convolutional

networks for action segmentation and detection, Proc. IEEE, 156-165,
LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P.: Gradient-based learning applied to document recognition, Proc. IEEE, 86, 2278-2324, 1998.
Li, H., Jiao, J. J., Luk, M., and Cheung, K.: Tide-induced groundwater level fluctuation in coastal aquifers bounded by L-shaped coastlines, Water Resour. Res., 38, 6-1-6-8, 2002.
Long, J., Shelhamer, E., and Darrell, T.: Fully convolutional networks for semantic segmentation, Proc. IEEE, 3431-3440,
Lu, C., Werner, A. D., and Simmons, C. T.: Threats to coastal aquifers, Nature Climate Change, 3, 605-605, 2013.
Lu, C., Cao, H., Ma, J., Shi, W., Rathore, S. S., Wu, J., and Luo, J.: A proof-of-concept study of using a less permeable slice along the shoreline to increase fresh groundwater storage of oceanic islands: Analytical and experimental validation, Water Resour. Res., 55, 6450-6463, 2019.
Maier, H. R. and Dandy, G. C.: Neural networks for the prediction and forecasting of water resources variables: a review of modelling issues and applications, Environ. Modell. Softw., 15, 101-124, 2000.
Mehr, A. D. and Nourani, V.: A Pareto-optimal moving average-multigene genetic programming model for rainfall-runoff modelling, Environ. Modell. Softw., 92, 239-251, 2017.
Mei, Y., Tan, G., and Liu, Z.: An improved brain-inspired emotional learning algorithm for fast classification, Algorithms, 10, 70, 2017.
Nair, V. and Hinton, G. E.: Rectified linear units improve restricted boltzmann machines, Icml,
Park, Y., Lee, J.-Y., Kim, J.-H., and Song, S.-H.: National scale evaluation of groundwater chemistry in Korea coastal aquifers: evidences of seawater intrusion, Environ. Earth Sci., 66, 707-718, 2012.
Pascanu, R., Mikolov, T., and Bengio, Y.: On the difficulty of training recurrent neural networks, International conference on machine learning, 1310-1318, 2013.
Pratheepa, V., Ramesh, S., Sukumaran, N., and Murugesan, A.: Identification of the sources for groundwater salinization in the coastal aquifers of Southern Tamil Nadu, India, Environ. Earth Sci., 74, 2819-2829, 2015.
Reichstein, M., Camps-Valls, G., Stevens, B., Jung, M., Denzler, J., and Carvalhais, N.: Deep learning and process understanding for data-driven Earth system science, Nature, 566, 195-204, 2019.
Rumelhart, D. E., Hinton, G. E., and Williams, R. J.: Learning representations by back-propagating errors, Nature, 323, 533-536, 1986.
Sahoo, S., Russo, T., Elliott, J., and Foster, I.: Machine learning algorithms for modeling groundwater level changes in agricultural regions of the US, Water Resour. Res., 53, 3878-3895, 2017.
Salimans, T. and Kingma, D. P.: Weight normalization: A simple reparameterization to accelerate training of deep neural networks, Advances in neural information processing systems, 29, 2016.
Senthil Kumar, A., Sudheer, K., Jain, S., and Agarwal, P.: Rainfall-runoff modelling

using artificial neural networks: comparison of network types, Hydrological Processes: An International Journal, 19, 1277-1291, 2005.
Seo, Y., Kim, S., Kisi, O., and Singh, V. P.: Daily water level forecasting using wavelet decomposition and artificial intelligence techniques, J. Hydrol., 520, 224-243, 2015.
Solgi, R., Loáiciga, H. A., and Kram, M.: Long short-term memory neural network (LSTM-NN) for aquifer level time series forecasting using in-situ piezometric observations, J. Hydrol., 601, 126800, 2021.
Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting, J. Mach. Learn. Res., 15, 1929-1958, 2014.
Taylor, K. E.: Summarizing multiple aspects of model performance in a single diagram, J. Geophys. Res.-Atmos., 106, 7183-7192, 2001.
Torres, J. F., Troncoso, A., Koprinska, I., Wang, Z., and Martínez-Álvarez, F.: Deep learning for big data time series forecasting applied to solar power, The 13th International Conference on Soft Computing Models in Industrial and Environmental Applications, 123-133, 2018.
Wan, R., Mei, S., Wang, J., Liu, M., and Yang, F.: Multivariate temporal convolutional network: A deep neural networks approach for multivariate time series forecasting, Electronics, 8, 876, 2019.
Wunsch, A., Liesch, T., and Broda, S.: Groundwater level forecasting with artificial neural networks: a comparison of long short-term memory (LSTM), convolutional neural networks (CNNs), and non-linear autoregressive networks with exogenous input (NARX), Hydrol. Earth Syst. Sci., 25, 1671-1687, 2021.
Xu, Z. and Hu, B. X.: Development of a discrete-continuum VDFST-CFP numerical model for simulating seawater intrusion to a coastal karst aquifer with a conduit system, Water Resour. Res., 53, 688-711, 2017.
Xue, Y., Wu, J., Ye, S., and Zhang, Y.: Hydrogeological and hydrogeochemical studies for salt water intrusion on the south coast of Laizhou Bay, China, Groundwater, 38, 38-45, 2000.
Yan, J., Mu, L., Wang, L., Ranjan, R., and Zomaya, A. Y.: Temporal convolutional networks for the advance prediction of ENSO, Sci Rep, 10, 1-15, 2020.
Zeng, X., Wu, J., Wang, D., and Zhu, X.: Assessing the pollution risk of a groundwater source field at western Laizhou Bay under seawater intrusion, Environ. Res., 148, 586-594, 2016.
Zhan, C., Dai, Z., Soltanian, M. R., and Zhang, X.: Stage-wise stochastic deep learning inversion framework for subsurface sedimentary structure identification, Geophys. Res. Lett., 49, e2021GL095823, 2022.
Zhang, D., Lin, J., Peng, Q., Wang, D., Yang, T., Sorooshian, S., Liu, X., and Zhuang, J.: Modeling and simulating of reservoir operation using the artificial neural network, support vector regression, deep learning algorithm, J. Hydrol., 565, 720-736, 2018a.
Zhang, J., Zhu, Y., Zhang, X., Ye, M., and Yang, J.: Developing a Long Short-Term Memory (LSTM) based model for predicting water table depth in agricultural areas, J. Hydrol., 561, 918-929, 2018b.

711 Zhang, J., Zhang, X., Niu, J., Hu, B. X., Soltanian, M. R., Qiu, H., and Yang, L.:
712 Prediction of groundwater level in seashore reclaimed land using wavelet and
713 artificial neural network-based hybrid model, J. Hydrol., 577, 123948, 2019.
714 Zhang, X., Miao, J., Hu, B. X., Liu, H., Zhang, H., and Ma, Z.: Hydrogeochemical
715 characterization and groundwater quality assessment in intruded coastal brine
716 aquifers (Laizhou Bay, China), Environ. Sci. Pollut. Res., 24, 21073-21090, 2017.
717 Zhang, X., Dong, F., Dai, H., Hu, B. X., Qin, G., Li, D., Lv, X., Dai, Z., and Soltanian,
718 M. R.: Influence of lunar semidiurnal tides on groundwater dynamics in estuarine
719 aquifers, Hydrogeol. J., 28, 1419-1429, 2020.
720