Journal of Hydrology 561 (2018) 918–929

Contents lists available at ScienceDirect

Journal of Hydrology

journal homepage: www.elsevier.com/locate/jhydrol

Research papers

Developing a Long Short-Term Memory (LSTM) based model for predicting water table depth in agricultural areas

Jianfeng Zhang[a], Yan Zhu[b], Xiaoping Zhang[a,*], Ming Ye[c], Jinzhong Yang[b]

[a] School of Mathematics and Statistics, Wuhan University, Wuhan, Hubei 430072, PR China
[b] State Key Laboratory of Water Resources and Hydropower Engineering Science, Wuhan University, Wuhan, Hubei 430072, PR China
[c] Department of Earth, Ocean, and Atmospheric Science, Florida State University, Tallahassee, FL 32306, USA

ARTICLE INFO

This manuscript was handled by Corrado Corradini, Editor-in-Chief, with the assistance of Vahid Nourani, Associate Editor

Keywords:
Machine-learning model
Water table depth
Deep learning
LSTM
Hetao Irrigation District

ABSTRACT

Predicting water table depth over the long-term in agricultural areas presents great challenges because these areas have complex and heterogeneous hydrogeological characteristics, boundary conditions, and human activities; also, nonlinear interactions occur among these factors. Therefore, a new time series model based on Long Short-Term Memory (LSTM), was developed in this study as an alternative to computationally expensive physical models. The proposed model is composed of an LSTM layer with another fully connected layer on top of it, with a dropout method applied in the first LSTM layer. In this study, the proposed model was applied and evaluated in five sub-areas of Hetao Irrigation District in arid northwestern China using data of 14 years (2000–2013). The proposed model uses monthly water diversion, evaporation, precipitation, temperature, and time as input data to predict water table depth. A simple but effective standardization method was employed to pre-process data to ensure data on the same scale. 14 years of data are separated into two sets: training set (2000–2011) and validation set (2012–2013) in the experiment. As expected, the proposed model achieves higher R^2 scores (0.789–0.952) in water table depth prediction, when compared with the results of traditional feed-forward neural network (FFNN), which only reaches relatively low R^2 scores (0.004–0.495), proving that the proposed model can preserve and learn previous information well. Furthermore, the validity of the dropout method and the proposed model's architecture are discussed. Through experimentation, the results show that the dropout method can prevent overfitting significantly. In addition, comparisons between the R^2 scores of the proposed model and Double-LSTM model (R^2 scores range from 0.170 to 0.864), further prove that the proposed model's architecture is reasonable and can contribute to a strong learning ability on time series data. Thus, one can conclude that the proposed model can serve as an alternative approach predicting water table depth, especially in areas where hydrogeological data are difficult to obtain.

1. Introduction

Groundwater provides an important source of water for domestic, agricultural, and industrial use. However, groundwater resources are vulnerable to over-exploitation, climate change, and biochemical pollution (Bouwer, 2000; Sophocleous, 2005; Evans and Sadler, 2008; White and Falkland, 2010; Karandish et al., 2015). As a result, many areas over the world face groundwater shortages. An example of such areas is the Hetao Irrigation District, one of the largest irrigation districts in China that is located in the arid area of the Yellow River watershed. The Yellow River serves as the main source of irrigation water in this district. However, the availability of irrigation water from the Yellow River has decreased dramatically, with intensified water resource use in the Yellow River watershed (Yang et al., 2003). Therefore, groundwater has become an important source of supplementary irrigation water in Hetao Irrigation District. The effective management of groundwater resources, especially in the context of increased groundwater demands for agriculture use, is necessary to provide sustainable use of water resources in Hetao Irrigation District. Sustainable groundwater planning and management requires accurate forecasting of water table depth (Wang et al., 2014). An accurate and reliable assessment of water table depth can help engineers and decision-makers to: (1) develop optimal water resource allocation strategies, (2) adjust crop patterns in different sub-irrigation areas, and (3) develop optimal irrigation schedules while controlling the effects of salinity related to intensive irrigation (Xu et al., 2010). The objective of this study is to develop an effective and accurate method for predicting water table depth that can be used to help engineers and decision-makers manage

* Corresponding author.
E-mail address: xpzhang.math@whu.edu.cn (X. Zhang).

https://doi.org/10.1016/j.jhydrol.2018.04.065
Received 11 January 2018; Received in revised form 23 April 2018; Accepted 25 April 2018
Available online 26 April 2018

groundwater resources and make management decisions.

In the last two decades, predictions of water table depth have usually been made by using physically based models such as MODFLOW and HYDRUS (e.g., Pang et al., 2000; Batelaan et al., 2003; Zume and Tarhule, 2008; Faulkner et al., 2009; Xu et al., 2012). However, those models have many practical limitations because they are always data demanding and time consuming during model development and calibration. Using physically based models to predict water table depth in Hetao Irrigation District is particularly challenging because the district covers a large area, lacks abundant hydrogeological data, and has strong spatial and seasonal variability in freeze-thaw periods.

In the last decade, many studies have investigated the advantages and disadvantages of various physically based models and evaluated their prediction performance with that of emerging data-driven, machine learning methods. The machine learning methods include Multiple Linear Regression (MLR) (Sahoo and Jha, 2013), Support Vector Regression (SVR) (Yu et al., 2006; Yoon et al., 2011; Belayneh et al., 2014; Mirzavand and Ghazavi, 2015), and Artificial Neural Networks (ANN) (Raman and Sunilkumar, 1995; Daliakopoulos et al., 2005; Sarangi et al., 2006; Napolitano et al., 2011; Parchami-Araghi et al., 2013; Seo et al., 2015; Chang et al., 2016). These statistical methods explore the spatial and temporal patterns hidden in historical data without using a physical model because the latter always requires a large number of physical parameters and a deep understanding of the physical processes of the modeling domain. In many cases, machine learning methods can achieve a better predictive performance than physically based models (Parkin et al., 2007; Chu and Chang, 2009; Mohanty et al., 2013; Karandish and Šimunek, 2016). For example, Mohanty et al., (2013) found that an ANN model provided a better prediction of water table depth than MODFLOW for short-term predictions. Karandish and Šimunek (2016) found that both Adaptive Neuro-Fuzzy Inference System and SVM models performed well when compared with HYDRUS-2D for water stressed conditions. Therefore, machine learning methods provide promising tools for predicting water table depth.

ANN methods are used in this study for predicting water table depth, because ANN has a strong self-learning ability, which is suitable for complicated problems. However, traditional FFNN models do not have the ability to learn time series data because they cannot preserve previous information, resulting in limited prediction capability for long-term time series data, such as data related to water table depth (Cannas et al., 2006). This problem can be resolved by using more advanced FFNN models. Recently, researchers have integrated FFNN with other methods such as genetic algorithms (Ketabchi and Ataie-Ashtiani, 2015; Bahrami et al., 2016; Mehr and Kahya, 2017), wavelet transform (Adamowski and Chan; 2011, Kisi and Shiri, 2012; Nourani et al., 2015), and singular spectrum analysis (Sahoo et al., 2017; Polomčićet al., 2017). Genetic algorithm can be applied to optimize neural networks. Wavelet transform and singular spectrum analysis can pre-process time series data, then add processed time series data into neural networks and thus allow FFNN models to learn time series data very well. However, these advanced FFNN models require complicated procedures related to data pre-processing. For example, Sahoo et al. (2017) first employed singular spectrum analysis to decompose time series data into separable and interpretable reconstructed components, and then applied genetic algorithm to select the best reconstructed components as inputs of FFNN. While this data pre-processing can strengthen the ability of a FFNN model to learn time series data, subjective user intervention is needed, e.g., choosing the number of different reconstructed components. Furthermore, the pre-processing is time consuming because of the calculation of many reconstructed components.

The present study focuses on a time series prediction task, so recurrent neural network (RNN) (Rumelhart et al., 1986) is a suitable choice. A RNN model has internal self-looped cells, allowing the RNN to "remember" time series information and making it adept at performing time series tasks. In addition, in this study, LSTM, a special kind of RNN, that works well in processing long term time series data, was chosen due to its sophisticated network structure. Compared with aforementioned advanced FFNN model, the proposed LSTM-based model only applied a very simple data pre-processing method. LSTM is a famous deep learning model. It is recurrent, where connections between units form a directed cycle allowing data to flow both forwards and backwards within the network; then the previous information can be preserved for future use. LSTM model have already been used as a very advanced model in the field of Artificial Intelligence and Deep Learning, such as in nature language processing (Mikolov et al., 2010; Sundermeyer et al., 2012), speech recognition (Graves and Jaitly, 2014), machine translation (Sutskever et al., 2014) and automatic image captioning (Wang et al., 2016). However, only a few studies have applied RNNs or LSTMs to handle time series data in the field of hydrology (Silva et al., 2013).

The goal of this study was to develop a two-layer LSTM-based model for predicting water table depth in Hetao Irrigation District. The model contained one layer of LSTM and a fully connected layer atop of the LSTM layer. The model employed monthly water diversion, evaporation, precipitation, temperature and time as input variables for predicting water table depth in the district. The rest of the paper is organized as follows. In Section 2, the study area and observational data are introduced. Section 3 illustrates RNN, LSTM, dropout regularization, the proposed model's architecture, and model evaluation criteria. Section 4 presents the experiment results and discussion. Finally, Section 5 concludes the paper.

2. Study area and observational data

2.1. Study area

Hetao Irrigation District lies within Bayannaoer City, Inner Mongolia, China (106°20′–109°29′ N, 40°14′–41°18′ E; elevation 1020–1050 m a.s.l.). The Yellow River and Wolf Mountain form the northern and southern boundaries, respectively. Characterized by narrow strips of flat terrain and fertile land, the entire Hetao Irrigation District has been divided into five sub-areas from west to east as follows: Ulanbuh, Jiefangzha, Yongji, Yichang and Urad (Fig. 1) based on management practices. Mean annual precipitation is 169.4 mm, about 70% of which falls during June–September; the maximum precipitation occurs in August. The mean, minimum and maximum temperatures were 3.9 °C, −14.6 °C, and 28.4 °C, respectively (China Meteorological Data Service Center, 1954–2013). Hetao Irrigation District covers 11,073 km^2; the central part of the district is about 180 km long and 60 km wide. About 52% (5740 km^2) of the entire terrestrial area is irrigated. A total of 227 monitoring wells are used in the district (Fig. 1) with the water table depth measured every 5 days. Because monthly data are used in the model, the measured water table depth is averaged by month.

2.2. Data and statistical analysis

Our experiments used 14 years (2000–2013) of time series data from the five sub-areas in Hetao Irrigation District. Since people in this district use surface water, pumping volume is very small, so it was omitted in this study. Therefore, at the beginning, the data included water diversion, precipitation, evaporation, temperature, water table depth, area of the region, water consumption for industry and domestic use. Then Lasso regression (Yuan and Lin, 2006; Tibshirani, 1996), a statistical method that can be used to select variables, was applied to choose important variables from the original data, based on regression weights. Finally, only five variables were chosen for this study: water diversion, precipitation, evaporation volume, temperature, and water table depth. Time series of these variables are presented for the five sub-

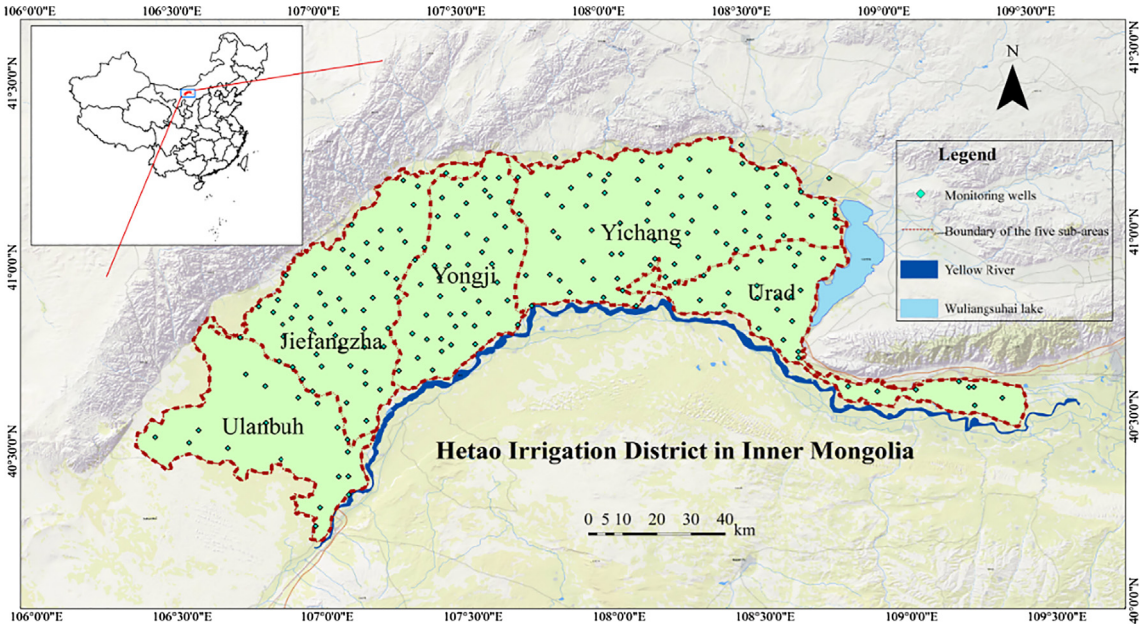

Fig. 1. Location of Hetao irrigation district, the boundary of the five sub-areas, and the observation wells.

areas of the district (Fig. 2). From Fig. 2, these data were found to have periodic characteristics. Therefore, time (at the monthly scale) has also been used as an input variable to improve the model's ability to generalize, creating a sixth variable.

In the present study, the first 12 years of time series data were used as a training set, and the next two years of data was used as a validation set. A statistic description of the water table depth of different sub-areas is shown in Table 1. Table 1 shows that Ulanbuh had the smallest standard deviation (0.331 m) and range of water table depth (1.297 m), meaning the water table depth of Ulanbuh is more stable than that of other areas.

2.3. Data pre-processing

As shown in Fig. 2, the input data varied widely. These differences will have a negative effect on the model's ability to learn. Therefore, all six of the variables were standardized to ensure they remain on the same scale. This pre-processing can guarantee a stable convergence of parameters in the model developed in the present study. The standardization formula is as follows:

$$x_{ij}^{(new)} = \frac{x_{ij} - \overline{x}_i}{\sigma_i} \quad (1)$$

where x_{ij} represents data in ith year, jth month; $\overline{x}_i$ and σ_i are the average and standard deviation of data in the ith year, respectively.

3. Methodology

3.1. Long Short-Term Memory network (LSTM)

Before introducing LSTM, we would like to first introduce RNN because LSTM is a special kind of RNN. RNNs were first developed in the 1980s; these networks have connections between neurons and form a directed cycle. This type of structure creates an internal self-looped cell, which allows it to display dynamic temporal behavior. RNNs have chain-like structures of repeating modules (Fig. 3(a)). These structures can help RNNs to "remember" previous information, which allows the RNNs to process arbitrary (long time) sequences. Therefore, RNNs have been successful in learning sequences.

Forward propagation begins with a specific initial state $h_0 = 0$. For each time step from t = 1 to t = τ, the following update equations were applied:

$$h_t = \tanh(b_h + Wh_{t-1} + Ux_t), \quad (2)$$

$$o_t = b_o + Vh_t, \quad (3)$$

where x_t is the input vector at time t and h_{t-1} is the hidden cell state at time t − 1. The parameters are the bias vector b_h and b_o, as well as the weight matrices U, W and V for input-to-hidden, hidden-to-hidden and hidden-to-output connections, respectively.

The gradients of the RNNs can be computed via Back-Propagation Through Time (Rumelhart et al., 1986; Werbos, 1990). However, Back-Propagation Through Time is not sufficiently efficient to learn a pattern from long term dependency because of a gradient vanishing problem (Hochreiter, 1998). This problem can be solved by the structure of LSTMs (Hochreiter and Schmidhuber, 1997; Graves, 2012; Jozefowicz et al., 2015). Like RNNs, LSTMs also have chain like modules, but the repeating modules have more complicated structures. Each repeating module of LSTMs contain a memory block. This memory block is specifically designed to store information over long time periods.

The memory block contains four parts: a CEC (the Constant Error Carousel) cell in addition to three special multiplicative units called gates. The CEC cell runs straight down the entire chain without any activation function and thus the gradient does not vanish when Back-Propagation Through Time is applied to train a LSTM. Therefore, LSTMs have been shown to learn long-term dependencies more easily than RNNs because information can easily flow along the cells unchanged. Furthermore, the input, forget (Gers et al., 2000) and output gates, in each memory block can control the flow of information inside memory block. The input, forget, and output gates control the extent to which new input flows into a CEC cell, information is stored in a cell, and output flows of the cell into the rest of the networks, respectively.

A schematic of memory block is shown in Fig. 3(b). It includes block input, three gates (input, forget, output), and a CEC cell.

Similar to RNNs, LSTMs computes a mapping from an input sequence x to an output sequence y by calculating the network unit activations using the following equations iteratively from t = 1 to t = τ with initial values $C_0 = 0$ and $h_0 = 0$:

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i), \quad (4)$$

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f), \quad (5)$$

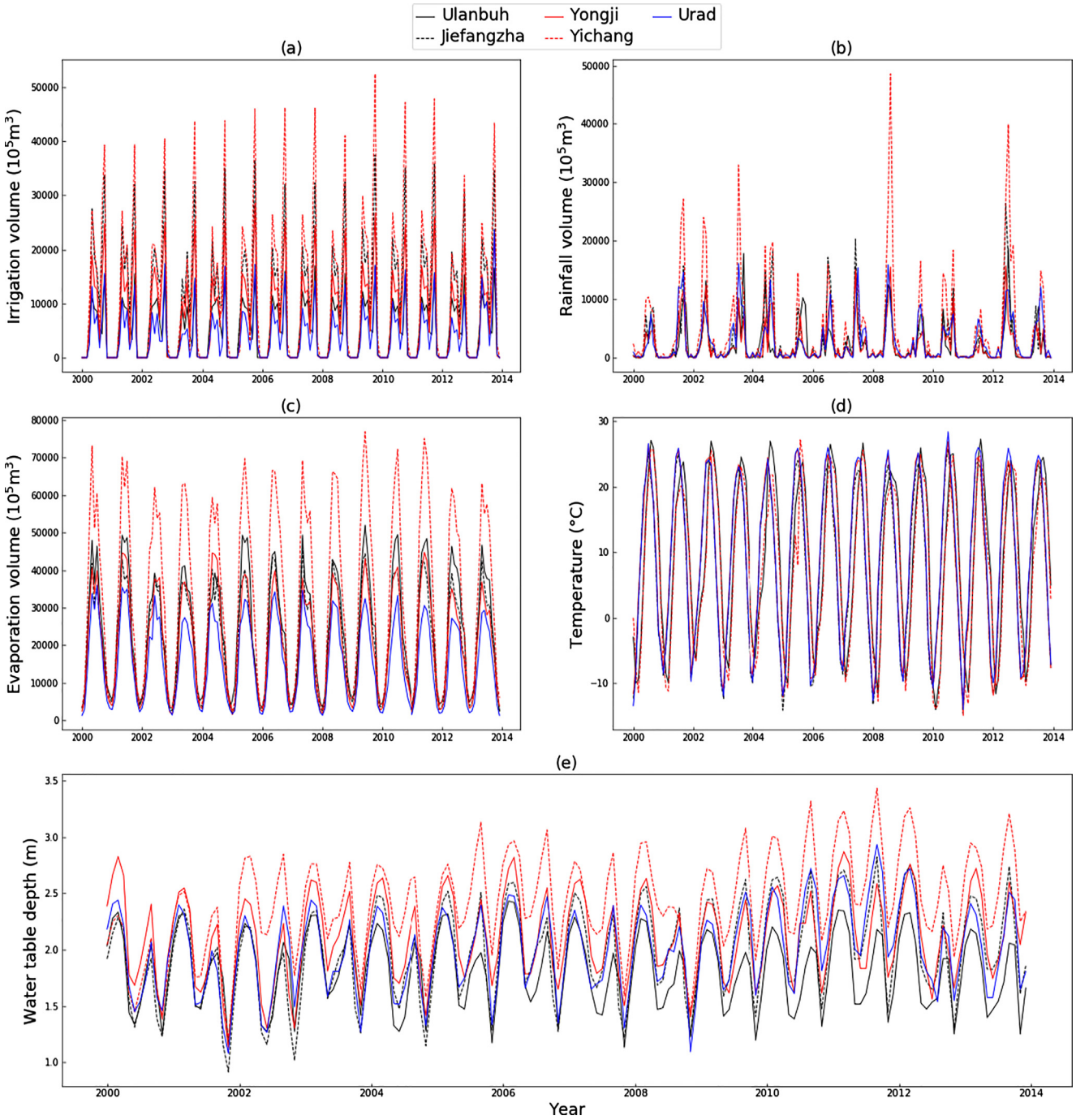

Fig. 2. Variables for different sub-areas, including (a) water diversion, (b) precipitation (c) evaporation volume, (d) temperature and (e) water table depth.

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o), \quad (6)$$

$$\widetilde{C}_t = \tanh(W_c x_t + U_c h_{t-1} + b_c), \quad (7)$$

$$C_t = f_t \otimes C_{t-1} + i_t \otimes \widetilde{C}_t, \quad (8)$$

$$h_t = o_t \otimes \tanh(C_t), \quad (9)$$

Table 1
Statistic description of water table depth of five sub-areas.

Areas	Average (m)	Maximum (m)	Minimum (m)	Standard Deviation (m)	Skewness
Ulanbuh	1.817	2.430	1.133	0.331	−0.167
Jiefangzha	1.970	2.821	0.914	0.422	−0.140
Yongji	2.111	2.870	1.129	0.379	−0.108
Yichang	2.421	3.430	1.202	0.429	−0.222
Urad	2.002	2.932	1.080	0.375	−0.103

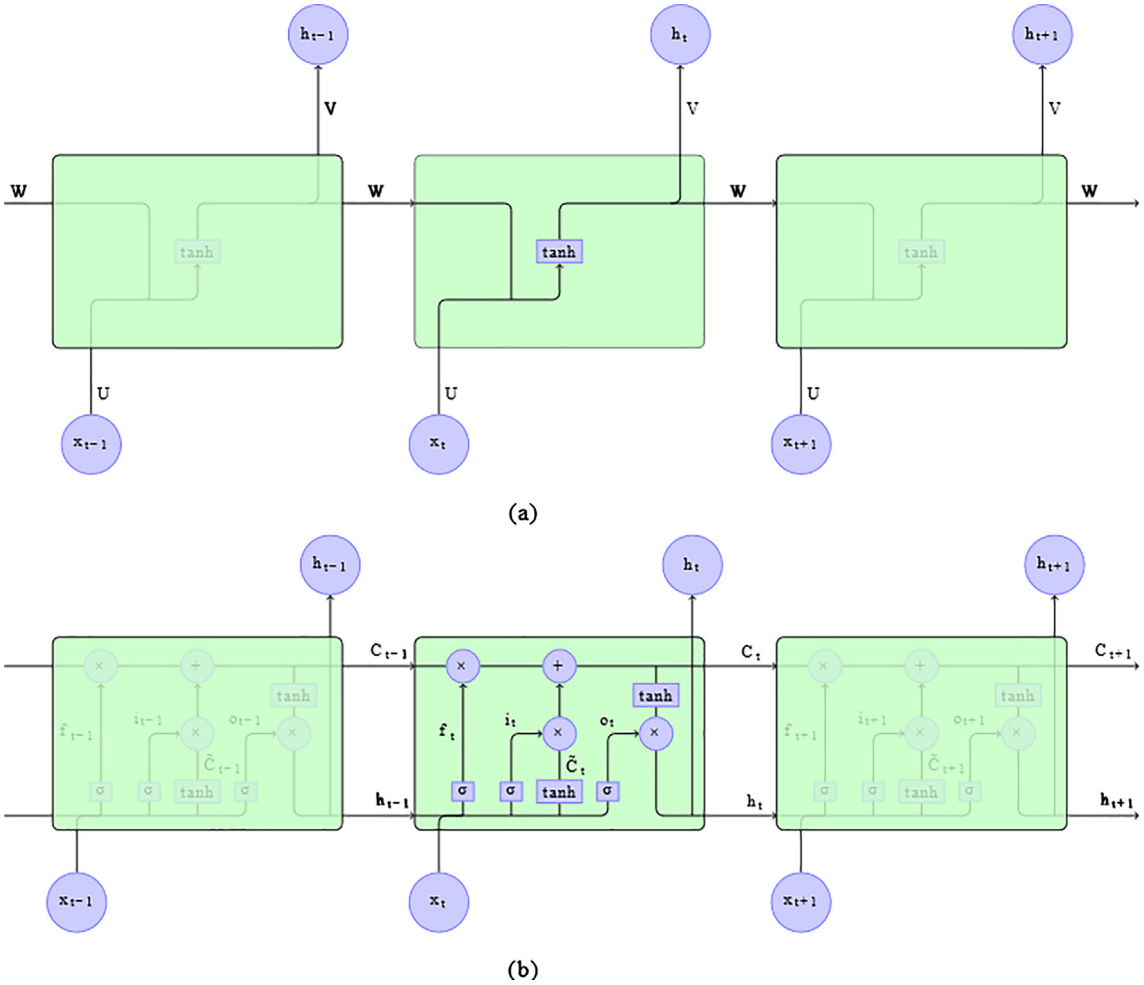

Fig. 3. (a) Chain like structure of the recurrent neural network. The self-connected hidden units allow information to be passed from one step to the next. (b) A graphical representation of LSTM's memory block.

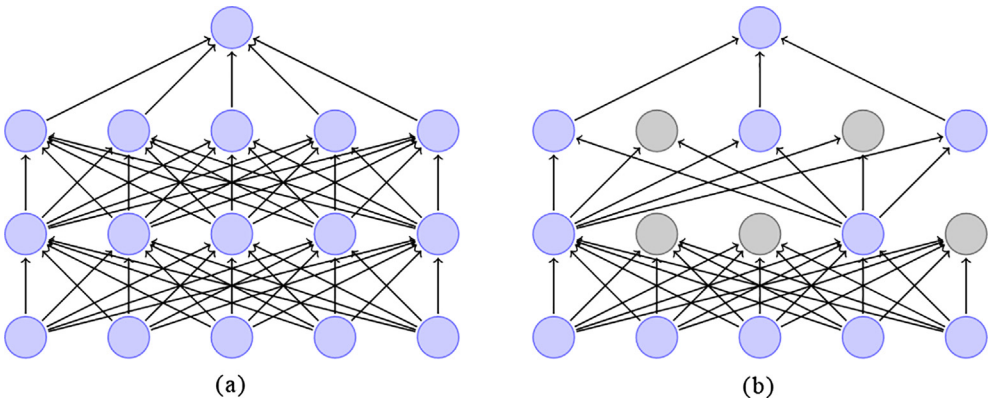

Fig. 4. (a) A standard feed-forward neural network with two hidden layers; (b) Applying dropout to two hidden layers on network (a). Grey units on (b) have been dropped.

where W_i, W_f, and W_o denote the matrix of weights from the input, forget, and output gates to the input, respectively. Similarly, U_i, U_f, and U_o denote the matrix of weights from the input, forget, and output gates to the hidden, respectively. b_i, b_f, b_o denote the input, forget, and output gate bias vectors, respectively. σ is an element-wise non-linear activation function: *logistic sigmoid*. i_t, f_t, ot and C_t are the input, forget, output gates and the cell state vectors at time t, respectively, all of which are the same size as the cell output vector h_t. The element-wise multiplication of two vectors is denoted with ⊗.

3.2. Dropout for neural networks

Deep neural networks are well suited to process big data. However, deep neural networks with large number of parameters can easily be overfitting, especially when data are limited. Therefore, dropout provides an effective regularization method that can be used to solve this problem (Hinton et al., 2012; Srivastava et al., 2014). The most important idea of dropout method is that at each training iteration, when the neural network is updating a certain layer where the dropout is

applied, it randomly does not update, or "dropout" some neurons (with probability p) in this layer. This means that a part of the neural network was sampled and it was trained at one iteration. In each iteration of training, a different part of the network was sampled and trained. With dropout, the weights of the neurons learned through backpropagation become somewhat more insensitive to the weights of the other neurons. Thus, dropout can help to prevent the networks from relying on certain neurons in the layers too much and reduce the neurons' co-adaptability. At testing time, all of the neurons of the networks are retained (no dropout), but the activations are scaled by p (probability of dropout). A neural network, whose first and second hidden layers have been applied dropout, is shown in Fig. 4.

3.3. Our proposed model framework

In this work, we were interested in predicting water table depth. This is a time series problem because the current water table depth has changed in a way that is dependent on previous. This time series prediction was cast as a regression problem. The proposed model is illustrated in Fig. 6(a). The input data were first put into the LSTM layer. The input gate of the LSTM layer will recompose input data and decide which input data is important; this process is similar to principal component analysis (PCA). However, the LSTM layer can preserve previous information, which can help to improve the ability of the model to learn time series data. A fully connected layer is set atop the LSTM layer in order to improve the model's learning ability. Moreover, dropout is set on the LSTM layer to prevent overfitting. The loss function is defined below:

$$LOSS = \sum_{i=1}^{N} (y_i - \hat{y}_i)^2, \tag{10}$$

where y_i is measured value at time i; and $\hat{y}_i$ is predicted value at time i.

However, this model framework has some limitations. First, the initial parameters of the proposed model will affect the final results. In addition, even though an LSTM layer has a strong ability to learn time series data, its fitting ability may be insufficient. Therefore, a fully connected layer was added atop of a single LSTM layer. Nevertheless, exactly how many LSTM layers should be used as hidden layers in order to reach the optimized results remained unknown.

The flowchart of the proposed model framework is displayed in Fig. 5. All of the code in this study was written in the Theano package of Python software. Throughout, the computation was performed on an Intel Core i5-4210U CPU with 4G RAM.

3.4. Model evaluation criteria

In this study, the root mean square error (*RMSE*), and coefficient of determination (R^2) were used to evaluate the model's accuracy between the measured and predicted values. R^2 measures the degree of how well the outcomes are replicated by the model, ranging between $[-\infty, 1]$ where for optimal model prediction an R^2 score close to 1 is preferred. The R^2 equation is as follows,

$$R^2 = \frac{\sum_{i=1}^{N} (y_i - \bar{y})^2 - \sum_{i=1}^{N} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{N} (y_i - \bar{y})^2}, \tag{11}$$

where y_i is measured value at time i, $\bar{y}$ is mean of y_i, (i = 1,…, *N*) and $\hat{y}_i$ is predicted value at time i.

Diverse types of information related to the predictive capacities of the model were measured through *RMSE*. *RMSE* measures the prediction precision which creates a positive value by squaring the errors. *RMSE* scores range between $[0, \infty]$, and the model prediction is ideal if *RMSE* is 0. *RMSE* is defined as,

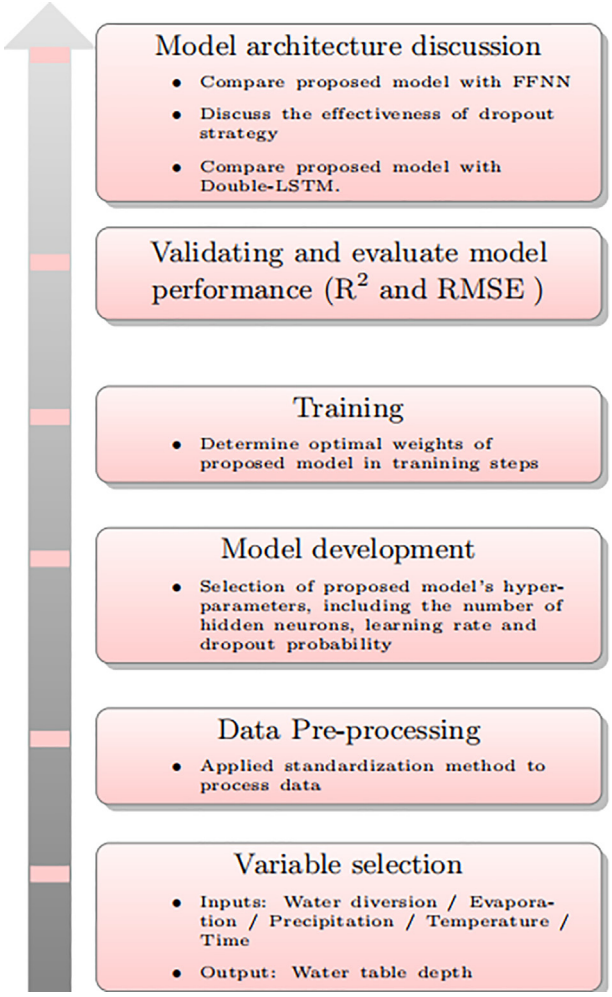

Fig. 5. Flowchart of this study.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N} (y_i - \hat{y}_i)^2}{N}}. \tag{12}$$

4. Results and discussions

In Section 4.1, the first 12 years of data were used, including water diversion, evaporation, precipitation, temperature, month and water table depth to train the proposed model (different models were used for different areas). Each model was trained using stochastic gradient descent (SGD) method. In order to observe the changes in water table depth in the entire Hetao Irrigation District, the proposed model also has been evaluated in the entire area, whose temperature was the average of five sub-areas and precipitation, water diversion and evaporation volume were sum of five sub-areas. This "all-encompassing" area was named "Hetao". It is reasonable to expect that the proposed model can work well in the entire area if it can perform well in other sub-areas. In addition, the proposed model was compared with FFNN in the water table depth prediction. In Section 4.2, the effectiveness of dropout was explored. Finally, Section 4.3 discusses the architecture of the proposed model and compares it with a Double-LSTM model.

4.1. Water table depth prediction results

As mentioned above, in the training process, the first 12 years of data were used to train the proposed model in different areas. After the

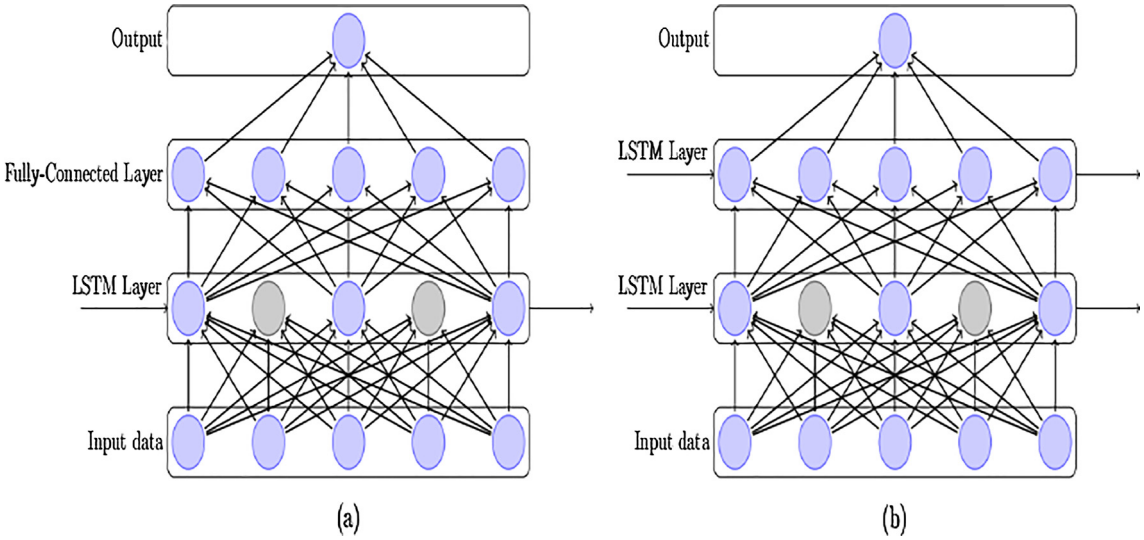

Fig. 6. (a) Structure of the proposed model. Dropout has been applied at the Long Short-Term Memory (LSTM) layer. (b) Structure of the Double-LSTM model. Dropout has been applied at the first LSTM layer. Grey units have been dropped.

Table 2
Results of Yongji's water table depth prediction, with different hyper-parameters set to the proposed model. The hyper-parameters in first row are used in this study. The bold significances represent the optimal hyper-parameters used in the proposed model which leads to the best R-Square score.

Neurons	Learning rate	Dropout	Iterations	Loss	R^2	Time (min)
40	**10^{-4}**	**0.5**	**20,000**	9.86	**0.82**	4.38
40	10^{-4}	0.5	40,000	7.11	0.82	8.41
40	10^{-4}	0.5	5000	30.10	0.35	1.21
40	10^{-4}	0.0	20,000	0.72	0.59	5.31
40	10^{-4}	0.8	20,000	30.70	0.64	4.02
40	10^{-3}	0.5	20,000	5.08	0.62	4.46
40	10^{-5}	0.5	20,000	52.07	0.69	4.29
70	10^{-4}	0.5	20,000	8.76	0.49	9.76
10	10^{-4}	0.5	20,000	23.37	0.41	3.30

models had been trained, each model was validated through use of the validation set and tuned the hyper-parameters of the models. Two performance metrics of different models in different sub-areas and "Hetao" were computed in order to obtain the optimal model hyper-parameters (the number of hidden neurons in the LSTM layer, and the learning rate). The optimal hyper-parameters of the proposed model used for water table depth prediction are shown in Table 2 (first row). Note that models for different areas use the same hyper-parameters.

In order to illustrate how hyper-parameters affect the results, different hyper-parameters were set for the proposed model and the Yongji sub-area was used as an example. Results, including training loss, R^2 and running time, are also displayed in Table 2. From Table 2 it can be learned that a higher learning rate (10^{-3}) may cause the optimization task to miss the optimal point (R^2 was only 0.62); however, a lower learning rate (10^{-5}) may help to avoid overshooting but may cause the model to use a longer time to converge. Too many training iterations may not ensure more optimal results (the results from 20,000 and 40,000 training iterations were the same). In addition, it is common to set a dropout probability to 0.5 in the deep learning field, as introduced by Hinton et al., (2012). Moreover, too many neurons (70 in this study) are computationally expensive (20,000 iterations in 9.76 min) and often cause overfitting, meanwhile, having an insufficient number of neurons (10 in this study) may decrease the network's learning ability, as can be seen in Table 2. Therefore, during implementation, 40 hidden neurons were chosen in the LSTM layer.

To compare the proposed model with FFNN model, FFNN model in different areas were also trained using the same hyper-parameters as mentioned above. Results in terms of the two evaluation metrics in different areas are summarized in Table 3.

From Table 3, the R^2 of the proposed model ranged from 0.789 to 0.952 and *RMSE* from 0.070 m to 0.184 m. The R^2 of the proposed model was much higher than that of FFNN model, which ranged from 0.004 to 0.466. In addition, the *RMSE* of the proposed model was much smaller than that of FFNN model with the *RMSE* ranging from 0.239 m to 0.372 m. These findings indicate that the proposed model can predict water table depth more accurately than other methods. Table 3 also shows that the proposed model has the best performance in Ulanbuh's water table depth prediction (R^2 = 0.952, *RMSE* = 0.070 m). We believe this occurred because the range and standard deviation of water table depth in Ulanuh were the smallest (Table 1), meaning the water table depth was more stable in Ulanbuh than in other areas. The proposed model produced better results with respect to R^2 and *RMSE* than the FFNN model, especially in Yichang and Urad where the

Table 3
Comparison of performance statistics for the proposed model, FFNN, model without dropout and Double-LSTM model in water table depth prediction in the last two years.

Area	Proposed model			FFNN		Model without dropout			Double-LSTM	
	Loss	R^2	*RMSE* (m)	R^2	*RMSE* (m)	*Loss*	R^2	*RMSE* (m)	R^2	*RMSE* (m)
Ulanbuh	9.32	0.952	0.070	0.447	0.239	0.738	0.874	0.114	0.864	0.119
Jiefangzha	11.42	0.810	0.184	0.466	0.287	0.771	0.499	0.278	0.387	0.308
Yongji	8.47	0.842	0.145	0.327	0.279	0.725	0.591	0.217	0.550	0.228
Yichang	9.81	0.849	0.144	0.004	0.372	1.204	0.645	0.222	0.620	0.230
Urad	10.63	0.789	0.167	0.495	0.258	0.832	0.368	0.289	0.170	0.331
Hetao	7.34	0.841	0.137	0.373	0.273	1.084	0.599	0.218	0.357	0.226

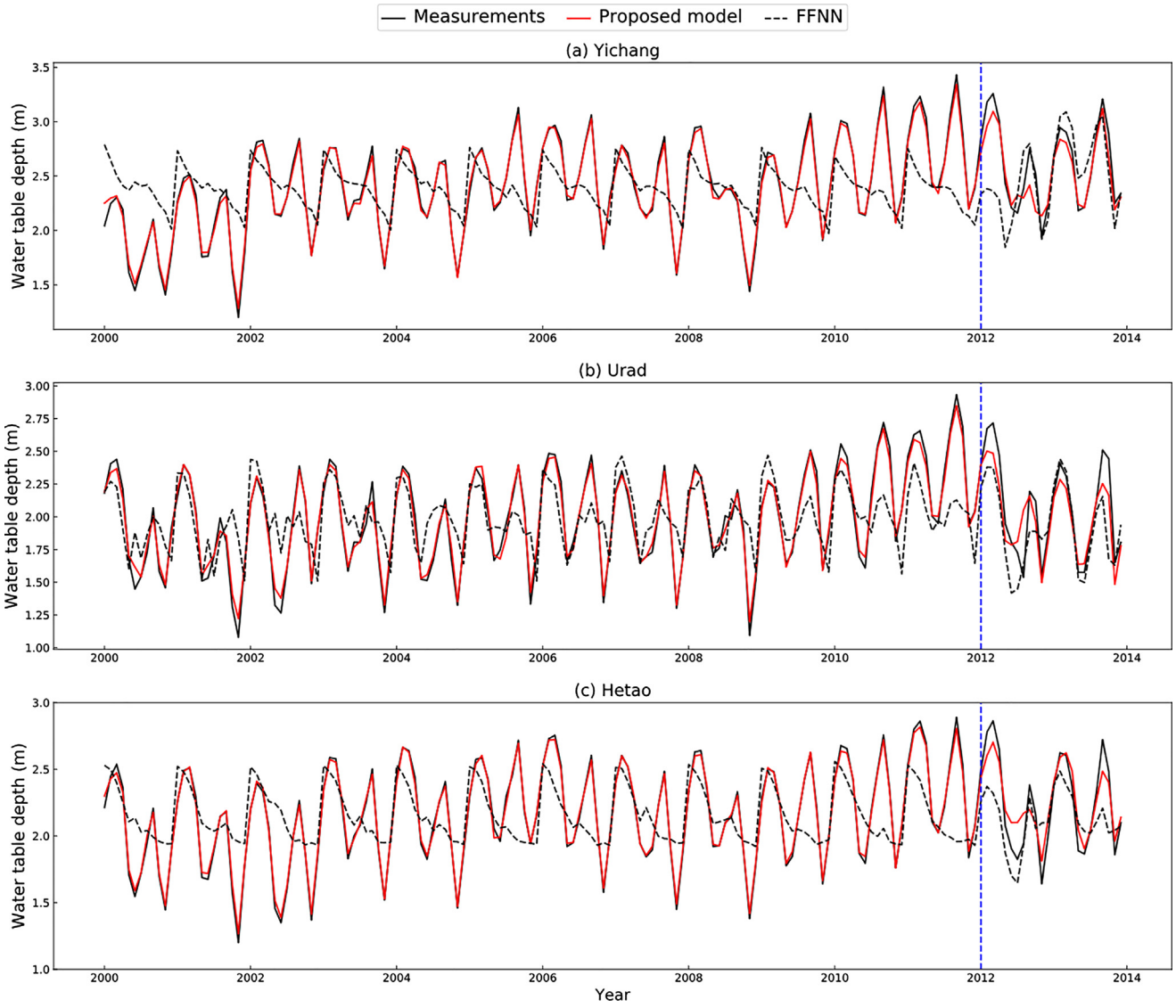

Fig. 7. Comparison of measured and simulated water table depth using the proposed model and a feed-forward neural network (FFNN) model in different areas. The blue dash line separates the data into two sets: the training and validating sets. For brevity, we only present three typical areas here.

performance metrics of the proposed model were much better than the FFNN model's. In addition, Table 3 illustrates that the proposed model also worked well in predicting the water table depth in the "Hetao" area, meaning the proposed model can be used to predict the water table depth in the entire Hetao Irrigation District.

Fig. 7 displays the measured and predicted water table depth from both the proposed and the FFNN models in different sub-areas. For brevity, only the results of Yichang, Urad and "Hetao" are presented (Fig. 7), showing both the proposed and the FFNN models can capture the variation trend of water table depth in these three areas. However, the FFNN model fails to accurately predict the water table depth because it cannot capture the previous information, which is further illustrated in Fig. 8. The FFNN model was found to not accurately predict the water table depth in the first month of 2012, indicating that the FFNN model cannot "remember" the previous water table depth. This phenomenon is especially obvious in Yichang and Urad, because during 2009–2011, the government decreased the irrigation volume there, leading to a 40–60 cm increase in water table depth. Therefore, even though a traditional FFNN model has strong ability to represent the variation in water table depth, it cannot predict water table depth as accurately as the proposed model; this further confirms the LSTM's ability to process time series data.

4.2. Dropout effect

In order to explore the dropout method that was used to prohibit overfitting, the results from the proposed model were compared with a model with same architecture but without applying dropout in the LSTM layer. The same hyper-parameters (40 hidden neurons, 10^{-4} learning rate and 20,000 training iterations) were set in the model without dropout and its performance was compared with the proposed model in water table depth prediction. Training loss and two evaluation metrics were used for evaluating these two models. Fig. 9 shows the prediction results of the proposed model and the model without dropout in three areas. The evaluation metrics and training loss are shown in Table 3. From Fig. 9, one can see that the proposed model and model without dropout provided very similar results in the training process in different areas. However, the results in the validating process showed stronger deviations from the predicted value. Table 3 shows that the training loss of the proposed model ranged from 7.34 to 11.42, whereas that of model without dropout ranged from 0.725 to 1.204. However, the R^2 of the proposed model was higher than that of model without dropout, which ranged from 0.368 to 0.874; also, the *RMSE* of the proposed model was smaller than that of model without dropout, which ranged from 0.114 m to 0.289 m. These findings indicate that

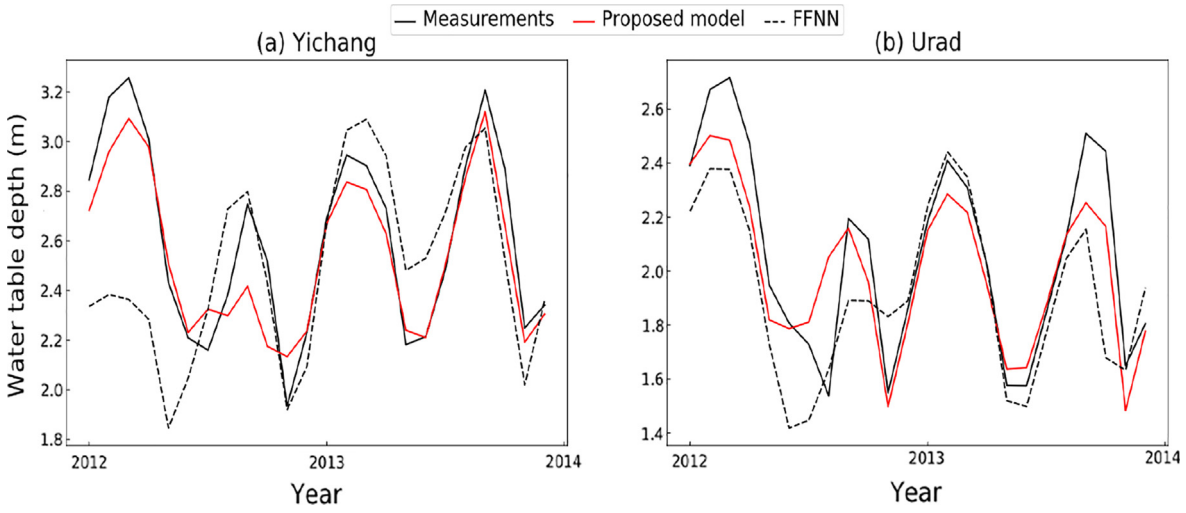

Fig. 8. The last two years of water table depth prediction results in (a) Yichang and (b) Urad.

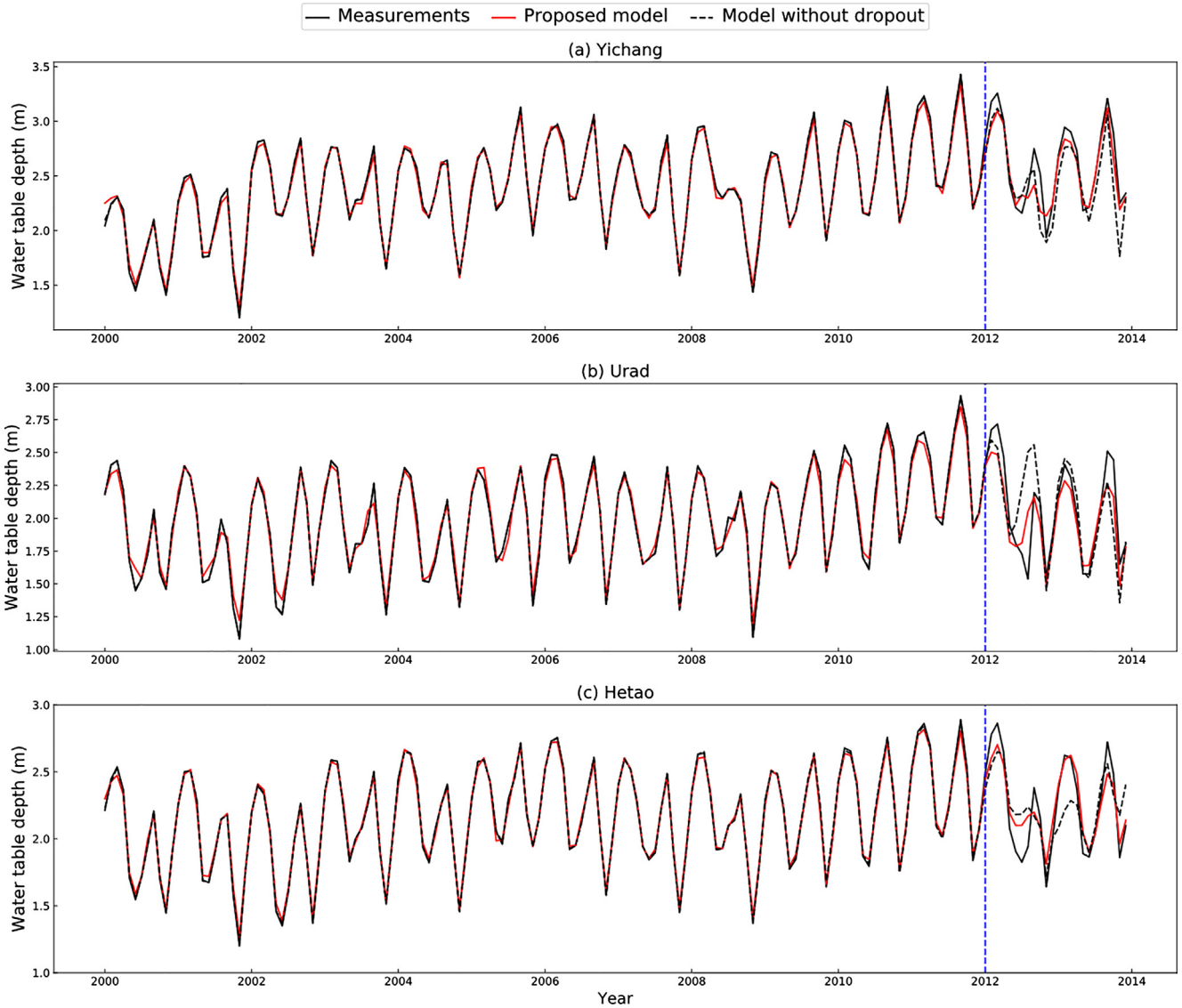

Fig. 9. Comparison of measured and simulated water table depth using the proposed model and model without dropout in different areas. The blue dash line separates the data into two sets: the training and validating sets. For brevity, we only present three typical areas here.

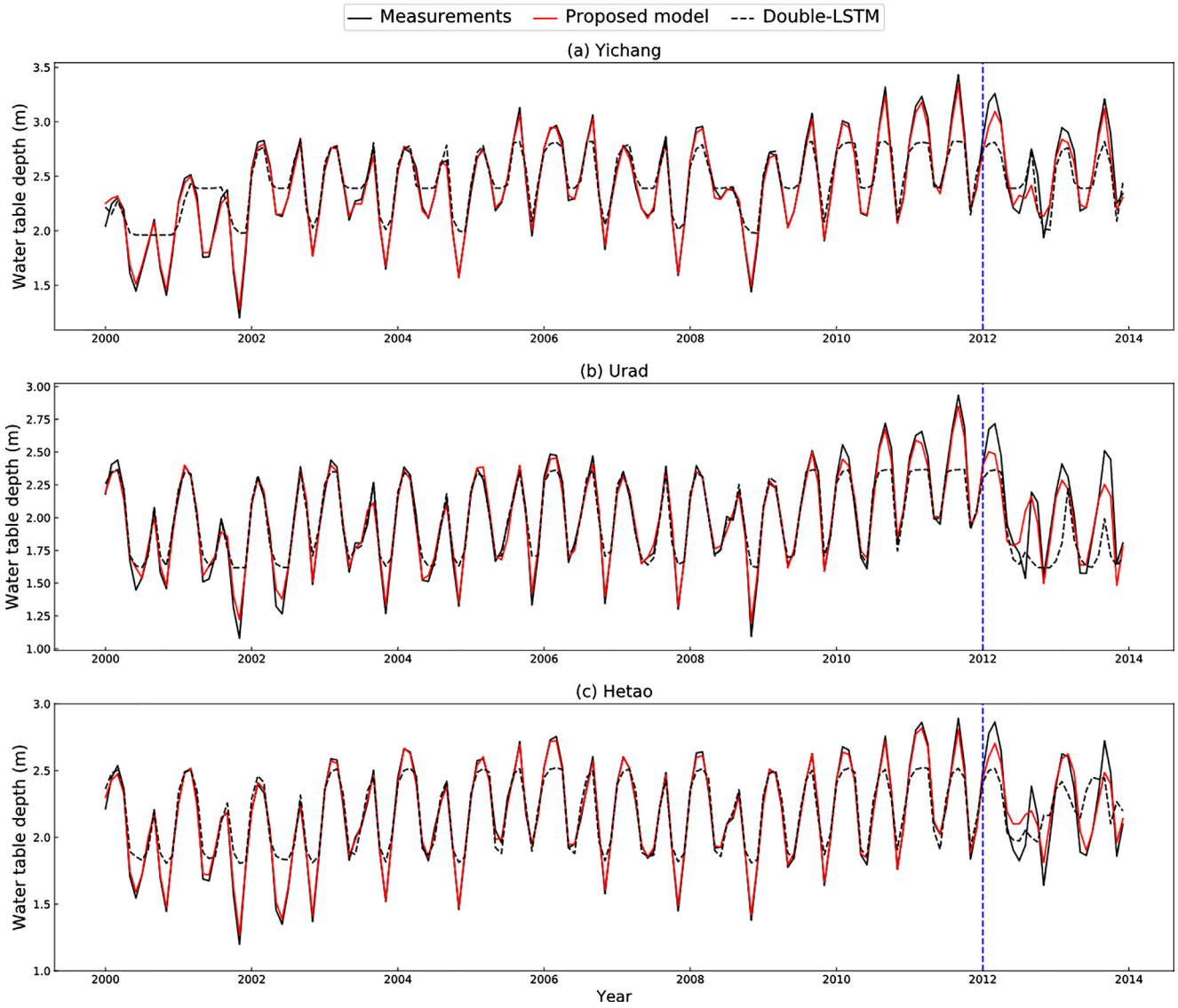

Fig. 10. Comparison of measured and simulated water table depth using the proposed model and Double-LSTM model in different areas. The blue dash line separates the data into two sets: the training and validating sets. For brevity, we only present three typical areas here.

even though the model without dropout can fit training data perfectly, it cannot predict the last two years of water table depth as well as the proposed model. This phenomenon is a typical overfitting. However, from the reported result of the proposed model, the conclusion could be drawn that the dropout method can help the proposed model by preventing overfitting.

4.3. Discussion about model structure

Note that the proposed model has one layer of LSTM and one fully connected layer. The proposed model was compared with the FFNN model in Section 4.1 to discuss the proposed model's ability to preserve previous information and process time series data. Therefore, in order to discuss the learning ability of the proposed model, it was also compared with a model that has different structures but the same architectural scale (two hidden layers), like a model that has two LSTM layers (Double-LSTM). Since a Double-LSTM model has two hidden LSTM layers, one can intuitively expect that this model can forecast water table depth well in these areas. The architecture of the Double-LSTM model is illustrated in Fig. 6(b).

Similar to Section 4.1, two evaluation metrics of these two models were compared. The Double-LSTM model is very similar to the proposed model; only change the fully connected layer to LSTM layer in Double-LSTM model. The same hyper-parameters were used to train the Double-LSTM model in different areas. Table 3 shows summarized results of these two models. Table 3 shows that the R^2 of the proposed model was higher than that of the Double-LSTM model, which ranged from 0.170 to 0.864; meanwhile, the *RMSE* of the proposed model was smaller than that of the Double-LSTM model, which ranged from 0.119 m to 0.331 m. Fig. 10 shows the prediction results of both the proposed and Double-LSTM models. One can see that although both the models performed well in capturing variations in the water table depth, the Double-LSTM model failed to fit data in the training process, as can obviously be seen at the peak values. We believe this phenomenon was caused by the insufficient fitting ability of the LSTM layer or the lack of an adequate number of hidden cell; thus, the Double-LSTM model cannot fit the training data. In addition, the simulation results become worse in the validation process. The failure of the Double-LSTM model to predict water table depth further proves that a fully connected layer has strong fitting ability. Considering the results in Table 3, one can conclude that the proposed model is more effective than both the FFNN and Double-LSTM models in water table depth prediction. The proposed model cannot only preserve previous hydrologic and climatic information while performing well in the time series process; it also has

a strong learning ability.

5. Conclusions

In this study, a new framework composed by one LSTM layer and one fully connected layer is proposed for predicting water table depth, in order to help engineers and decision-makers to plan and manage groundwater resources in agricultural areas. This framework is different from a traditional neural network model and has not been widely used in the field of hydrology. The ability of the proposed model to predict monthly water table depth in five sub-areas of Hetao Irrigation District and in the entire district itself were evaluated and discussed. In addition, the proposed model was also compared with Double-LSTM model. The major conclusions are as follows:

(1) The proposed model provides a promising new method for predicting water table depth, as evidenced by satisfactory performance on water table depth prediction in five sub-areas and “Hetao”.
(2) The architecture of the proposed model is reasonable. The LSTM layer helps to maintain previous information and contributes to learning the time series data. The dropout method helps to prevent overfitting during the training process. A fully connected layer atop the LSTM layer helps to improve the learning and fitting ability of the model.
(3) The newly proposed model provides a valuable tool for predicting water table depth. It can serve as an alternative model to predict water table depth in places with complex hydrogeological characteristics and hydrogeological data are difficult to obtain.

In the future, we can design a deeper, wider and more robust LSTM-based model, in order to provide more accurate water table depth prediction worldwide. In addition, the proposed model also can be combined with other methods, such as PCA and wavelet transform. Furthermore, the proposed model can be applied to other time series prediction tasks, such as soil water change and streamflow prediction.

Acknowledgments

The study was partially supported by Natural Science Foundation of China through Grants (51790533, 11671313, 51779178), National Key Research and Development Program of China (2017YFC0403301), Natural Science Foundation of Hubei Province, China (2016CFB576). All resources of this paper, including training and testing code, and demo data are publicly available at https://github.com/jfzhang95/LSTM-water-table-depth-prediction/.

References

Adamowski, J., Chan, H.F., 2011. A wavelet neural network conjunction model for groundwater level forecasting. J. Hydrol. 407, 28–40.
Bahrami, S., Ardejani, F.D., Baafi, E., 2016. Application of artificial neural network coupled with genetic algorithm and simulated annealing to solve groundwater inflow problem to an advancing open pit mine. J. Hydrol. 536, 471–484.
Batelaan, O., Smedt, F.D., Triest, L., 2003. Regional groundwater discharge: phreatophyte mapping, groundwater modelling and impact analysis of land-use change. J. Hydrol. 275, 86–108.
Belayneh, A., Adamowski, J., Khalil, B., Ozga-Zielinski, B., 2014. Long-term SPI drought forecasting in the Awash River Basin in Ethiopia using wavelet neural network and wavelet support vector regression models. J. Hydrol. 508, 418–429.
Bouwer, H., 2000. Integrated water management: emerging issues and challenges. Agric. Water Manage. 45, 217–228.
Cannas, B., Fanni, A., See, L., Sias, G., 2006. Data preprocessing for river flow forecasting using neural networks: Wavelet transforms and data partitioning. Phys. Chem. Earth 31, 1164–1171.
Chang, F.J., Chang, L.C., Huang, C.W., Kao, I.F., 2016. Prediction of monthly regional groundwater levels through hybrid soft-computing techniques. J. Hydrol. 541, 965–976.
Chu, H.J., Chang, L.C., 2009. Application of optimal control and fuzzy theory for dynamic groundwater remediation design. Water Resour. Manage. 23, 647–660.
Daliakopoulos, I.N., Coulibaly, P., Tsanis, I.K., 2005. Groundwater level forecasting using artificial neural networks. J. Hydrol. 309, 229–240.
Evans, R.G., Sadler, E.J., 2008. Methods and technologies to improve efficiency of water use. Water Resour. Res. 44http://dx.doi.org/10.1029/2007WR006200. W00E04.
Faulkner, J., Hu, B., Kish, S., Hua, F., 2009. Laboratory analog and numerical study of groundwater flow and solute transport in a karst aquifer with conduit and matrix domains. J. Contam. Hydrol. 110, 34–44.
Gers, F.A., Schmidhuber, J., Cummins, F., 2000. Learning to forget: continual prediction with LSTM. Neural Comput. 12, 2451–2471.
Graves, A., 2012. Supervised Sequence Labelling with Recurrent Neural Networks. Springer-Verlag, Berlin, Heidelberg.
Graves, A., Jaitly, N., 2014. Towards end-to-end speech recognition with recurrent neural networks. In: Proceedings of the 31st International Conference on Machine Learning, Beijing, China, pp. 1764–1772.
Hinton, G.E., Srivastava, N., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.R., 2012. Improving neural networks by preventing co-adaptation of feature detectors. Comput. Sci. 3, 212–223.
Hochreiter, S., 1998. The vanishing gradient problem during learning recurrent neural nets and problem solutions. Int. J. Uncertain Fuzz. 6, 107–116.
Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural Comput. 9, 1735–1780.
Jozefowicz, R., Zaremba, W., Sutskever, I., 2015. An empirical exploration of recurrent network architectures. In: Proceedings of the 32nd International Conference on Machine Learning. Lille, France, pp. 2342–2350.
Karandish, F., Šimunek, J., 2016. A comparison of numerical and machine-learning modeling of soil water content with limited input data. J. Hydrol. 543, 892–909.
Karandish, F., Salari, S., Darzi-Naftchali, A., 2015. Application of virtual water trade to evaluate cropping pattern in arid regions. Water Resour. Manage. 29, 4061–4074.
Ketabchi, H., Ataie-Ashtiani, B., 2015. Evolutionary algorithms for the optimal management of coastal groundwater: a comparative study toward future challenges. J. Hydrol. 520, 193–213.
Kisi, O., Shiri, J., 2012. Wavelet and neuro-fuzzy conjunction model for predicting water table depth fluctuations. Hydrol. Res. 43, 286–300.
Mehr, A.D., Kahya, E., 2017. A Pareto-optimal moving average multigene genetic programming model for daily streamflow prediction. J. Hydrol. 549, 603–615. http://dx.doi.org/10.1016/j.jhydrol.2017.04.045.
Mikolov, T., Karafiat, M., Burget, L., Cernocký, J., & Khudanpur, S., 2010. Recurrent neural network based language model. In: Proceedings of the 11th Annual Conference of the International Speech Communication Association, Makuhari, Chiba, Japan, pp. 1045–1048.
Mirzavand, M., Ghazavi, R., 2015. A stochastic modelling technique for groundwater level forecasting in an arid environment using time series methods. Water Resour. Manage. 29, 1315–1328.
Mohanty, S., Jha, M.K., Kumar, A., Panda, D.K., 2013. Comparative evaluation of numerical model and artificial neural network for simulating groundwater flow in Kathajodi-Surua Inter-basin of Odisha, India. J. Hydrol. 495, 38–51.
Napolitano, G., Serinaldi, F., See, L., 2011. Impact of EMD decomposition and random initialisation of weights in ANN hindcasting of daily stream flow series: An empirical examination. J. Hydrol. 406, 199–214.
Nourani, V., Alami, M.T., Vousoughi, F.D., 2015. Wavelet-entropy data pre-processing approach for ANN-based groundwater level modeling. J. Hydrol. 524, 255–269.
Pang, L.P., Close, M.E., Watt, J.P.C., Vincent, K.W., 2000. Simulation of picloram, atrazine, and simazine leaching through two New Zealand soils and into groundwater using HYDRUS-2D. J. Contam. Hydrol. 44, 19–46.
Parchami-Araghi, F., Mirlatifi, S.M., Dashtaki, S.G., Mahdian, M.H., 2013. Point estimation of soil water infiltration process using artificial neural networks for some calcareous soils. J. Hydrol. 481, 35–47.
Parkin, G., Birkinshaw, S.J., Younger, P.L., Rao, Z., Kirk, S., 2007. A numerical modelling and neural network approach to estimate the impact of groundwater abstractions on river flows. J. Hydrol. 339, 15–28.
Polomčić, D., Gligorić, Z., Bajić, D., Cvijović, Č., 2017. A hybrid model for forecasting groundwater levels based on fuzzy C-mean clustering and singular spectrum analysis. Water 9, 541.
Raman, H., Sunilkumar, N., 1995. Multivariate modelling of water resources time series using artificial neural networks. Hydrol. SCI. J. 40, 145–163.
Rumelhart, D.E., Hinton, G.E., Williams, R.J., 1986. Learning representations by back-propagating errors. Nature 323, 533–536.
Sahoo, S., Jha, M.K., 2013. Groundwater-level prediction using multiple linear regression and artificial neural network techniques: a comparative assessment. Hydrogeol. J. 21, 1865–1887.
Sahoo, S., Russo, T.A., Elliott, J., Foster, I., 2017. Machine learning algorithms for modeling groundwater level changes in agricultural regions of the U.S. Water Resour. Res. 2017 (53), 3878–3895.
Sarangi, A., Singh, M., Bhattacharya, A.K., Singh, A.K., 2006. Subsurface drainage performance study using SALTMOD and ANN models. Agric. Water Manage. 84, 240–248.
Seo, Y., Kim, S., Kisi, O., Singh, V.P., 2015. Daily water level forecasting using wavelet decomposition and artificial intelligence techniques. J. Hydrol. 520, 224–243.
Silva, I.N.D., Cagnon, J.Â., Saggioro, N.J., 2013. Recurrent Neural Network Based Approach for Solving Groundwater Hydrology Problems. InTech Open Access Publisherhttp://dx.doi.org/10.5772/51598.
Sophocleous, M., 2005. Groundwater recharge and sustainability in the High Plains aquifer in Kansas. USA. Hydrogeol. J. 13, 351–365.
Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014. Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn Res. 15, 1929–1958.
Sundermeyer, M., Schluter, R., Ney, H., 2012. LSTM neural networks for language modeling. In: Proceedings of the 12th Annual Conference of the International Speech

Communication Association, Portland, Oregon, USA, pp. 601–608.
Sutskever, I., Vinyals, O., & Le, Q. V., 2014. Sequence to sequence learning with neural networks. In: Proceedings of the 28th Conference on Neural Information Processing Systems, Montreal, Canada, pp. 3104–3112.
Tibshirani, R., 1996. Regression shrinkage and selection via the Lasso. J. R. Statist. Soc. B 58, 267–288.
Wang, C., Yang, H., Bartz, C., Meinel, C., 2016. Image captioning with deep bidirectional LSTMs. In: Proceedings of the 24th ACM Multimedia Conference, Amsterdam, The Netherlands, pp. 988–997.
Wang, W., Yu, Z., Zhang, W., Shao, Q., Zhang, Y., Luo, Y., Jiao, X., Xu, J., 2014. Responses of rice yield, irrigation water requirement and water use efficiency to climate change in China: historical simulation and future projections. Agric. Water Manage. 146, 249–261.
Werbos, P.J., 1990. Backpropagation through time: what it does and how to do it. Proc. IEEE 78, 1550–1560.
White, I., Falkland, T., 2010. Management of freshwater lenses on small Pacific Islands. Hydrogeol. J. 18, 227–246.
Xu, X., Huang, G., Qu, Z., Pereira, L.S., 2010. Assessing the groundwater dynamics and impacts of water saving in the Hetao Irrigation District, Yello River basin. Agric. Water Manage. 98, 301–313.
Xu, X., Huang, G., Zhan, H., Qu, Z., Huang, Q., 2012. Integration of SWAP and MODFLOW-2000 for modeling groundwater dynamics in shallow water table areas. J. Hydrol. 412–413, 170–181.
Yang, L., Shen, R., Cao, X., 2003. Scheme of groundwater use in Hetao Irrigation District in Inner Mongolia. Trans. of the CSAE. 19, 56–59.
Yoon, H., Jun, S.C., Hyun, Y., Bae, G.O., Lee, K.K., 2011. A comparative study of artificial neural networks and support vector machines for predicting groundwater levels in a coastal aquifer. J. Hydrol. 396, 128–138.
Yuan, M., Lin, Y., 2006. Model selection and estimation in regression with grouped variables. J. R. Stat. Soc. B 68, 49–67.
Yu, P.S., Chen, S.T., Chang, I.F., 2006. Support vector regression for real-time flood stage forecasting. J. Hydrol. 328, 704–716.
Zume, J., Tarhule, A., 2008. Simulating the impacts of groundwater pumping on stream-aquifer dynamics in semiarid northwestern Oklahoma, USA. Hydrogeol. J. 16, 797–810.