A comprehensive study of deep learning for soil moisture prediction

Yanling Wang[1], Liangsheng Shi[1*], Yaan Hu[2], Xiaolong Hu[1], Wenxiang Song[1], Lijun Wang[1]

5 [1]State Key Laboratory of Water Resources Engineering and Management, Wuhan University, Wuhan, China

[2]State Key Laboratory of Hydrology-Water Resources and Hydraulic Engineering, Nanjing Hydraulic Research Institute, Nanjing, China

Correspondence to: Liangsheng Shi (liangshs@whu.edu.cn)

10 **Abstract.** Soil moisture plays a crucial role in the hydrological cycle, but accurately predicting soil moisture presents challenges due to the nonlinearity of soil water transport and variability of boundary conditions. Deep learning has emerged as a promising approach for simulating soil moisture dynamics. In this study, we explore ten different network structures to uncover their mechanisms of data utilization and maximize the potential of deep learning for soil moisture prediction, including three basic feature

15 extractors and seven diverse hybrid structures, six of which are applied to soil moisture prediction for the first time. We compare the predictive abilities and computational costs of the models across different soil textures and depths systematically. Furthermore, we exploit the interpretability of the models to gain insights into their workings and attempt to advance our understanding of deep learning in soil moisture dynamics. For soil moisture forecasting, our results demonstrate that the temporal modeling capability

20 of Long Short-Term Memory (LSTM) is well-suited. Besides, the improved accuracy achieved by feature attention LSTM (FA-LSTM) and the generative adversarial network-based LSTM (GAN-LSTM), along with the Shapley additive explanations (SHAP) analysis, help us discover the effectiveness of attention mechanisms and the benefits of adversarial training in feature extraction. These findings provide effective network design principles. The Shapley values also reveal varying data leveraging approaches

25 among different models. The t-Distributed Stochastic Neighbor Embedding (t-SNE) visualization illustrates differences in encoded features across models. In summary, our comprehensive study provides insights into soil moisture prediction and highlights the importance of the appropriate model design for specific soil moisture prediction tasks. We also hope this work serves as a reference for deep learning studies in other hydrology problems. The codes of 3 machine learning and 10 deep learning models are

30 open sourced.

Key words: Soil moisture; deep learning; hybrid models; machine learning; forward modeling; model interpretability

1.Introduction

35   Soil moisture is significant in simulating many hydrological processes since it controls the interaction of water and energy between the land surface and the atmosphere (Entin et al., 2000; Vereecken et al., 2022). Accurately providing information on soil moisture dynamics is crucial for effective water resources planning and management, agricultural production, climate prediction, and flood disaster monitoring (Vereecken et al., 2008; Sampathkumar et al., 2013). However, caused by the randomness of

40   rainfall and the nonlinear features of infiltration and evaporation processes (Guswa et al., 2002), soil moisture is highly variable and nonlinear in space and time (Heathman et al., 2012), which makes it difficult to forecast.

Since various mainstream approaches have been applied for soil moisture dynamics prediction, a comprehensive study is needed to provide suitable solutions for different predicting tasks, encourage

45   improvements on models and build confidence in this area. Traditionally, soil moisture dynamics prediction is widely based on physical models, such as the soil-plant-air model (Saxton et al., 1974), HYDRUS (Simunek et al., 2005), and CATHY (Camporese et al., 2015). Though these models are interpretable, they perform poorly in practical applications, because of the inestimable parameters (Gill et al., 2006) and inadequate description of physical processes (Li et al., 2022b). With the reduction in

50   data acquisition costs and advancements in computation, there has been an increasing focus on data-driven models. Initially, multiple linear regression (Qiu et al., 2003; Hummel et al., 2001) and empirical models (Azhar et al., 2011; Verma and Nema, 2021) are applied for soil moisture prediction. However, one nonnegligible problem is that these methods require calibrations and have limited generalization capabilities (Holzman et al., 2017; Jackson, 2003). Compared to these traditional data-driven models,

55   machine learning methods appear to possess stronger data fitting ability. For instance, support vector regression (SVR) (Gill et al., 2006) and random forest (RF) (Prasad et al., 2019) have both shown satisfactory and robust results with low computing costs in soil moisture prediction. Additionally, the single-layer feedforward neural network with generalized inverse operation -- Extreme Learning Machine (ELM) (Huang et al., 2006) can precisely predict the future trends of soil moisture and support

60   future irrigation scheduling. (Liu et al., 2014).

Currently, deep learning is the state-of-the-art data-driven method, which has made obvious improvements in many research areas (Lecun et al., 2015). Due to their powerful approximation ability, deep neural networks (DNNs) (Goodfellow et al., 2016) have been extensively researched from soil moisture descriptions (Cai et al., 2019; Prakash et al., 2018). Notably, recurrent neural networks (RNNs) (Pollack, 1990) excel at capturing temporal information in time series data and models sequential dependencies for predictions (Mikolov et al., 2011). This is consistent with the characteristics of soil moisture dynamics simulation. Fang et al. (2019) utilized Long Short-term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) for soil moisture and received satisfactory results. From a different perspective, Convolution Neural Networks (CNNs) (LeCun, 1989) are capable of extracting features from training data in specific dimensions, making them widely used in dealing with 2-D (Albawi et al., 2018; Patil and Rane, 2021) or 1-D data (Severyn and Moschitti, 2015; Shi et al., 2015). Therefore, 1D-CNNs are applied in many hydrology researches(Hussain et al., 2020; Chen et al., 2021). Additionally, attention mechanisms enable the selection of critical information from multiple input features or model outputs, which can be visualized using attention weight (Ding et al., 2020; Li et al., 2022a). On this foundation, self-attention can model dependencies and aggregate features from inputs disregarding their distance (Vaswani et al., 2017), which shows great potential in soil moisture prediction.

As various deep learning approaches focused on distinct mechanisms of data utilization, hybrid structures become a vital research area. On one hand, combining the feature importance processing methods -- attention mechanisms, with deep learning models, can indeed lead to improvements (Ahmed et al., 2021; Ding et al., 2019; Kilinc and Yurtsever, 2022). Li et al. proposed an attention-aware LSTM to estimate soil moisture and temperature and achieved better performance than LSTM alone (2022). In their work, three attention mechanisms help obtain the spatial-temporal feature vectors of LSTM inputs or outputs. On the other hand, the combinations of multiple neural networks tend to perform better than a single network alone (Semwal et al., 2021). The hybrid CNN-GRU model proposed by Yu et al.(2021) outperformed the independent CNN or GRU model in predicting root zone moisture. Besides, Li, et al. (2022) proposed EDT-LSTM, a stacked LSTM model based on the encoder-decoder structure (Sutskever et al., 2014) and residual learning (He et al., 2016). This achieved more stable results than a single LSTM. Regarding the optimization of training strategies, adversarial training in generative adversarial networks (GANs) (Goodfellow et al., 2014) can capture more information of real data. This helps to address the

90   problem of fuzzy prediction and provides a superior solution for weather forecast (Jing et al., 2019; Ravuri et al., 2021).

Therefore, it is essential to design effective and suitable neural network structures for soil moisture prediction tasks. In this study, we comprehensively evaluate the performance of various deep learning methods in soil moisture prediction, highlighting their key characteristics in terms of prediction accuracy

95   and computational costs. The models evaluated in this research range from machine learning models such as RF, ELM, and SVR to basic deep learning models, including 1D-CNN, LSTM, and the Encoder of Transformer(Vaswani et al., 2017), and hybrid deep learning models, including CNN-LSTM, LSTM-CNN, CNN-with-LSTM, FA-LSTM, TA-LSTM, FTA-LSTM, and GAN- LSTM. Notably, the Encoder of Transformer is first developed in soil moisture prediction, with CNN-LSTM, LSTM-CNN, FA-LSTM,

100   TA-LSTM, FTA-LSTM, and GAN-LSTM first applied and systematically compared for soil moisture. To gain insights into their workings and provide a thorough analysis of why some methods perform better, we utilize the SHAP (Lundberg et al., 2018) method to demonstrate the importance of features in different models and employ t-SNE visualization (Van der Maaten and Hinton, 2008) to show the encoded features across models. The systematical assessment of the models is carried out across multiple sites at 5 depths.

105   For forecasting soil moisture, the utilized data include meteorological data, soil temperature data, and soil moisture content data from previous days, as these inputs are closely associated with evaporation and infiltration processes.

In the remainder of this article, Sect. 2 describes the data used and the deep learning background; Sect. 3 presents a detailed description of the participating methods; Sect. 4 analyzes comparison results and

110   discusses the interpretability of the models. The conclusion is drawn in Sect. 5.

2. Data Description and Backgrounds

2.1 Data Description

To create a comprehensive evaluation under different soil types, in-situ observations at ten different

115   sites are downloaded from the International Soil Moisture Network (ISMN) (https://ismn.geo.tuwien.ac.at/en/), including the Monahans-6-ENE (Monahans), Necedah-5-WNW (Necedah), Falkenberg, AAMU-jtg (AAMU), Cullman-NAHRC (Cullman), Cape-Charles -5-ENE(Cape), LittleRiver, Spickard, Weslaco, and UpperBethlehem (UB). Detailed site information is

provided in Table 1, and their spatial locations are shown on a world map in Fig. 1. The temporal changes
120   of soil moisture at various depths are also drawn in Fig. 1.

This study builds individual predictive models for each site and depth, disregarding the inclusion of
static properties such as land cover, soil hydraulic properties, and topography. All deep learning models
in this study utilize the past four days' data to forecast the soil moisture content on the fifth day. This
means the input vector $I, \{x_{t-3}, x_{t-2}, x_{t-1}, x_t\}$ is used to predict the target value y_t, that is the soil
125   moisture SM_t at time t. The input data x_t at time t consists of the past soil moisture (SM), soil
temperature (ST), and meteorological data including precipitation (P), atmospheric temperature(AT),
long-wave radiation(LR), short-wave radiation(SR), wind speed(WS), and relative humidity(RH)), $x_t =$
$\{P_t, T_t, LW_t, SW_t, RH_t, WS_t, ST_t, SM_{t-1}\}$. For machine learning, we only utilize the x_t to generate
predictions. These features are selected as inputs because they are closely related to the soil
130   evapotranspiration, infiltration processes, and the internal behavior of the soil. Since groundwater level
observations are difficult to obtain, changes in the lower boundary conditions are excluded from the
inputs. Meteorological data is downloaded from NASA's website (https://power.larc.nasa.gov/data-
access-viewer/), while soil moisture and soil temperature data are obtained from the ISMN.

135   **Table 1.** Summary of main characteristics of ten sites.

	Sand	Silt	Clay	Period	Latitude	Longitude
Monahans-6-ENE	83	6	11	2010.4.21-2022.7.19	31.62190	102.80710
Necedah-5-WNW	83	11	6	2009.10.13-2022.11.26	44.06040	-90.17370
Falkenberg	73	21	6	2003.1.17-2020.6.30	52.16694	14.12417
AAMU-jtg	53	22	25	2010.2.6-2022.11.18	34.78300	-86.55000
Cullman-NAHRC	49	27	24	2006.5.18-2022.11.30	34.20000	-86.80000
Cape-Charles-5-ENE	49	27	24	2011.6.15-2022.11.1	37.29070	-75.92700
LittleRiver	47	30	23	2005.10.18-2020.1.1	31.50000	-83.55000
Spickard	35	41	24	2010.10.8-2022.11.26	40.25000	-93.71700
Weslaco	34	45	21	2017.1.1-2021.5.1	26.16000	-97.96000
UpperBethlehem	32	38	30	2008.9.15-2010.5.1	17.71700	-64.80000

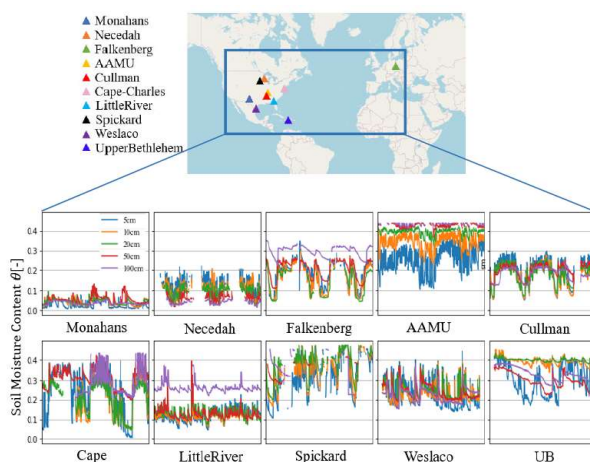



Figure 1. The spatial locations and soil moisture content time series at various depths of ten sites.

2.2 Deep Learning Backgrounds

140

Deep learning enhances the complexity and learning capability of traditional machine learning methods by adding multiple layers (Kamilaris and Prenafeta-Boldú, 2018). At each layer, input signals are weighted through the connections of each neuron and subsequently activated by activation functions (Schmidhuber, 2015). Deep learning discovers intricate structures in training data by utilizing

145 backpropagation to guide the machine in adjusting its internal parameters (Lecun et al., 2015).

In this study, the primary challenge in soil moisture prediction is processing the time-series data with specific dimensions and simulating soil moisture dynamics with high spatiotemporal variability. Given the diversity of neural networks, numerous methods have the potential to deal with specific time-series data. CNNs can extract local temporal information from the data by sliding convolutional kernels along

150 the time dimension. On the other hand, RNNs excel at capturing the overall temporal sequence information. Additionally, self-attention has the potential to associate inputs and make predictions, making them capable of handling sequence data effectively. These three types of networks can be regarded as fundamental feature extractors in deep learning. Furthermore, hybrid deep learning models integrate the characteristics of multiple models, enhancing their prediction capacities (Yu et al., 2021).

155 Combinations of CNNs, RNNs, and attention mechanisms have been widely utilized in many studies. Besides, employing specified training strategies with suitable network structures can also improve prediction performance. For instance, GANs enable the training objective of neural networks to go beyond minimizing data mean squirt error and utilize adversarial training to fully capture data regularities.

By designing appropriate network structures and training strategies, it is possible to further improve

160 prediction accuracy.

It is necessary to conduct a comprehensive evaluation to analyze the internal combining meaning of models and decide the most suitable combination rule for soil moisture prediction. With the collected data in 2.1, it is possible to deeply explore the prediction abilities of the deep learning models. We evaluate models from the perspectives of prediction accuracy and computational costs to provide a

165 reference for soil moisture dynamics predictions. Further research on model interpretability can provide insights into how the model structure influences the utilization of data, leading to a more effective design of the model structure.

3.Models and Methodology

170 Three machine learning models and seven deep learning models take part in this comparative research. Introductions to each model are provided below, along with key references for interested readers. The parameters of each model are recorded in Appendix A.

3.1 Machine Learning Methods

175 In this study, machine learning models Random Forest (RF), Extreme Learning Machine (ELM), and Support Vector Machine (SVM) are applied to compare with the deep learning models as a benchmark.

Random Forest, proposed by Breiman (Breiman, 2001), is used for regression and classification tasks, and has gained popularity for its high accuracy. RF works by constructing multiple decision trees on randomly sampled subsets of the training data. Each tree is trained on a random subset of features, and

180 the final prediction is made by averaging the predictions of the individual trees. This approach reduces overfitting and increases model stability. For soil moisture prediction, RF has proven to be a stable and reliable method (Carranza et al., 2021).

Extreme Learning Machine (Huang et al., 2006) utilizes a single-layer feedforward neural network as its foundation. ELM achieves fast learning speed and strong generalization ability by employing random

185 input layer weights and biases and applying generalized inverse matrix theory to calculate the output layer weights. The algorithm has been applied in various fields and has shown promising results. Liu et al.(2014) employed ELM to predict the large-scale soil moisture in Australian orchards. The results demonstrated that the model was capable of accurate forecasting.

Support Vector Machine (Cortes and Vapnik, 1995) was proposed for applications in classification

190   and regression. It aims to find the maximum-margin hyperplane that best separates sample points. To

make this hyperplane more robust in high-dimensional feature spaces, SVM uses kernel functions to

perform nonlinear mapping and create a new feature space where the data can be linearly separable. The

algorithm then finds the optimal classification hyperplane with the maximum margin. SVMs have

achieved great success in various fields. Gill et al. (2006) applied SVM to soil moisture prediction and

195   compared it with DNNs. The results showed that SVM was suitable for soil moisture content prediction.

Support Vector Regression (SVR) is a variant of SVM that is specifically designed for regression tasks,

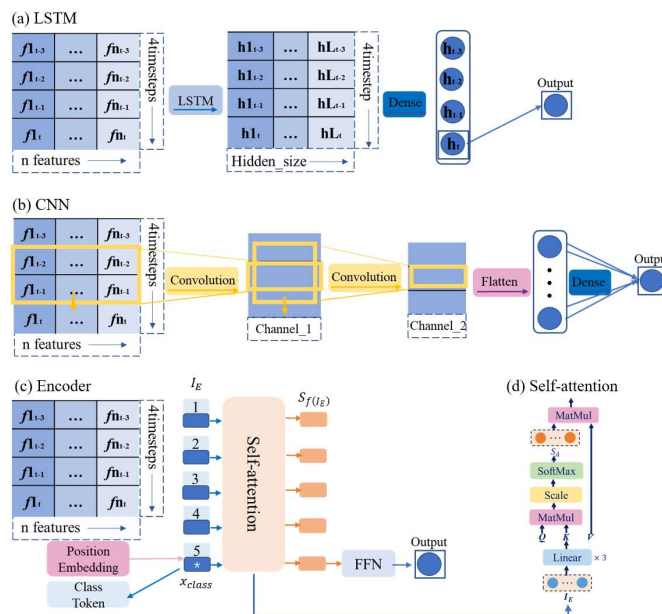
which is applied in this study.

For machine learning, x_t and y_t represents the input feature and target object, respectively. The

input data corresponds one-to-one in time to the target and serves as both the input and output of the

200   machine learning models. The prediction accuracy of machine learning serves as a comparison for deep

learning models. Hyperparameters used in models are recorded in Appendix A.

3.2 Basic Deep Neural Networks

205   **Figure 2.** Network structures of the LSTM(a), the 1D-CNN(b), and the proposed Encoder(c) inspired by Dosovitskiy

et al.(2020) with the self-attention structure (d)for soil moisture prediction.

3.2.1 LSTM

RNNs (Pollack, 1990) operate by recursing in the direction of sequence progression, with all nodes in

210 the network being chained together. These unique properties make RNNs effective in processing

sequence data and extracting temporal information, which has led to breakthroughs in natural language

processing (Connor et al., 1994). The ability of RNNs to model temporal dependencies is suitable for

predicting soil moisture.

Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) neural networks, were

215 proposed to address the limitations of traditional RNNs. LSTM can overcome the issue of gradient

vanishing and memorize more useful information through a special unit, which is called the cell state.

Thus, LSTM operates as follows:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{1}$$

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{2}$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{3}$$

$$\tilde{C}_t = tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \tag{4}$$

$$c_t = f_t \cdot c_{t-1} + i_t \cdot \tilde{C}_t \tag{5}$$

$$h_t = o_t \cdot tanh(c_t) \tag{6}$$

where W_i and b_i are the parameters for the input gate, W_f and b_f are the parameters for the forget

gate, W_o and b_o are the parameters for the output gate, W_c and b_c are used for cell state updating; σ

220 is the activation function.

We generate the time-dependent hidden states $\boldsymbol{H}$, $\{\boldsymbol{h}_{t-3}, \boldsymbol{h}_{t-2}, \boldsymbol{h}_{t-1}, \boldsymbol{h}_t\}$ from

input $\boldsymbol{I}$, $\{\boldsymbol{x}_{t-3}, \boldsymbol{x}_{t-2}, \boldsymbol{x}_{t-1}, \boldsymbol{x}_t\}$ through the LSTM. After sequentially processing all inputs in the LSTM,

the last hidden state $\boldsymbol{h}_t$ of the sequential output is used as the prediction for network training, as depicted

in Fig. 2a. This is because the input features at each time step can be encoded in the last hidden state.

225 The parameters in this model are recorded in Appendix A.

3.2.2 1D-CNN

CNNs (LeCun, 1989) were originally applied for image recognition. The convolution and pooling

layers in CNNs can extract the distinguishing features of the given data while reducing the amount of

230   data to be processed (Ajit et al., 2020). Consequently, CNNs are highly effective in processing data that

come in the form of multiple arrays.

For time series data, 1D-CNNs can extract local temporal features via convolution kernels that slide

along the time dimension. 1D-CNNs have demonstrated success in speech and natural language

processing applications (Abdel-Hamid et al., 2014; Severyn and Moschitti, 2015). Hence, 1D-CNNs are

235   capable of soil moisture prediction tasks. The complete forward-propagation process of a simple 1D-

CNN for soil moisture prediction is illustrated in Fig. 2b. Given that the input vector

$I, \{x_{t-3}, x_{t-2}, x_{t-1}, x_t\}$, two convolution layers are employed in the 1D-CNN architecture. The

convolution kernel size (Kernel_size) is set to 2, with a stride of 1. Specific parameters are listed in Table

A1. To preserve the information of the data, pooling layers are intentionally omitted.

240

3.2.3 Encoder

The self-attention mechanism can model the dependencies and aggregate features from inputs.

Therefore, a stacking structure of self-attention mechanisms like Transformer (Vaswani et al., 2017)

can achieve the functions of CNNs and RNNs without iterations. This provides a novel way for

245   predictions. In this study, we utilize the encoder structure of Transformer (Vaswani et al., 2017), Encoder,

as depicted in Fig. 2c, to predict soil moisture. The self-attention is shown in Fig. 2d, which operates as

follows:

$$S_A = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right) = softmax\left(\frac{(W_Q I_E)(W_K I_E)^T}{\sqrt{d_k}}\right) \tag{7}$$

$$S_{f(I_E)} = S_A \otimes V = softmax\left(\frac{(W_Q I_E)(W_K I_E)^T}{\sqrt{d_k}}\right) W_V I_E \tag{8}$$

where W_K, W_V and W_Q are the key, value, and query parameter matrices, respectively; I_E is the

Encoder input; $\frac{1}{\sqrt{d_k}}$ is the scaling factor, $d_k = 4$.

250   The outputs generated by the self-attention mechanism correspond to the inputs one-to-one. In this

study, a "class token" vector x_{class} is introduced as additional input to start the prediction process. The

class token is randomly initialized and can be trained, serving as the fifth input. It enables aggregate

global features from all other inputs and avoids bias towards a specific time step in the sequence.

However, the self-attention mechanism ignores the temporal order of the inputs. To address this issue,

255 we incorporate positional encoding to preprocess the inputs. Both the learnable positional encoding and sine cosine coding are tested in this research. The sine cosine positional encoding is defined as:

$$PE_{(pos1,2pos2)} = sin\left(\frac{pos1}{10000^{\frac{2pos2}{d_{model}}}}\right) \qquad (9)$$

$$PE_{(pos1,2pos2\ )} = cos\left(\frac{pos1}{10000^{\frac{2pos2}{d_{model}}}}\right) \qquad (10)$$

where the parameters $pos1$ and $pos2$ represent the positions of the first and second dimensions of the input, respectively. Here, $d_{model} = 8$ denotes the parameter of self-attention, which is equal to the input features at each time step.

260 The encoded position vectors PE are added to the original inputs before feeding them into the Encoder. With PE, the input of the Encoder is defined as follows:

$$I_E = \{x_{t-3}, x_{t-2}, x_{t-1}, x_t, x_{class}\} + PE \qquad (11)$$

3.3 Hybrid Deep Learning Models

3.3.1 Hybrid structure of CNN and LSTM

265 In this section, three connecting ways of CNNs and LSTMs, CNN-LSTM, LSTM-CNN, and CNN-with-LSTM, are considered. These hybrid models possess advanced capabilities in handling diverse types of data, generally leading to improved prediction accuracy. To ensure a rigorous comparison with the previous 1D-CNN and LSTM models, the parameters of the CNN and LSTM layers in our hybrid models are kept as consistent as possible with the 1D-CNN and LSTM models before. The detailed

270 parameter setting information can be found in Table A1.

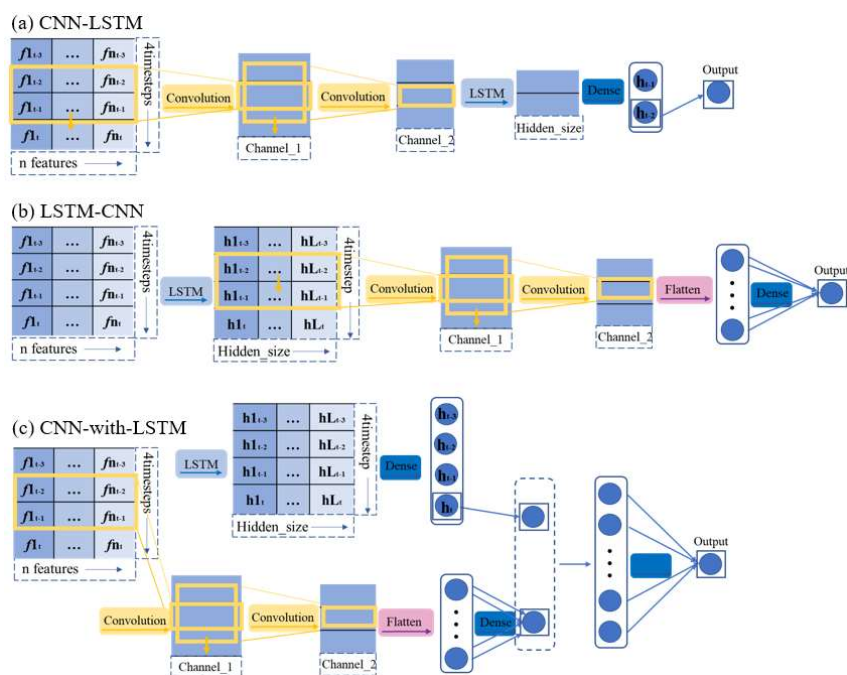



Figure 3. The framework of the proposed CNNs and LSTMs hybrid models: CNN-LSTM(a), LSTM-CNN(b), and CNN-with-LSTM(c).

275   **CNN-LSTM**

Generally, the CNN-LSTM model is comprised of CNN layers followed by LSTM layers. The input data first passes through convolution layers to better extract local features in the sequence data. Then LSTM layers are used to associate the time-series extracted features. Therefore, this kind of model excels at handling the input data in image format, which has been widely utilized in prediction tasks, yielding

280   positive outcomes in various applications (Semwal et al., 2021). In our soil moisture prediction task, CNN-LSTM consists of 2 convolution layers and an LSTM layer, which is shown in Table A1. As we mentioned in Section 3.2, the last hidden state h_t is still applied as the prediction. Fig. 3a depicted the structure of CNN-LSTM applied in this research.

285   **LSTM-CNN:**

In contrast to the CNN-LSTM model, the LSTM-CNN model first utilizes LSTM layers to associate the time series data and output high-dimensional related hidden states. Subsequently, convolution layers

are employed to extract the features of these time-dependent hidden states. This model has also been widely adopted in various applications (Xia et al., 2020). In this study, LSTM-CNN for soil moisture prediction consists of an LSTM layer and 2 convolution layers sequentially. The structure of LSTM-CNN can be seen in Fig. 3b. Detailed layers and parameters of this model are presented in Table A1.

CNN-with-LSTM:

CNN-with-LSTM is a model that employs the parallel combination of both CNN and LSTM, merging their outputs through concatenation, and uses a fully connected network for regression analysis. By combining the feature extraction capabilities of CNN with the time series memory ability of LSTM, this model captures both the local and global temporal characteristics of the input data. This kind of hybrid structure has been used in soil moisture prediction and achieved satisfactory results(Yu et al., 2021). In our work, CNN-with-LSTM is comprised of an LSTM layer and 2 convolution layers parallelly, and the structure is depicted in Fig. 3c. Table A1 lists the network structures of the CNN and LSTM models in addition to the parameter settings.

3.3.2 Hybrid Structure of Attention and LSTM

To enhance the accuracy of deep learning models and address the issue of lack of interpretability, attention mechanisms have been incorporated into LSTM models to weigh the importance of different input and output vectors dimensions (Li et al., 2022a; Ding et al., 2020; Xia et al., 2020). Attention mechanisms are commonly used in combination with other neural networks as a form of pre-processing or post-processing. Through training, attention mechanisms dynamically generate spatiotemporal attention importance weights to selectively focus on critical parts of the input or output. According to the specific roles of the attention mechanisms, the hybrid models can be classified into three categories: FA-LSTM (a feature attention mechanism with LSTM), TA-LSTM (a temporal attention mechanism with LSTM), and FTA-LSTM (an LSTM combines both feature and temporal attention mechanisms). Ding et al.(2020) conducted experiments on this three kind of hybrid models in flood prediction, confirming the effectiveness of incorporating LSTM with attention mechanisms.

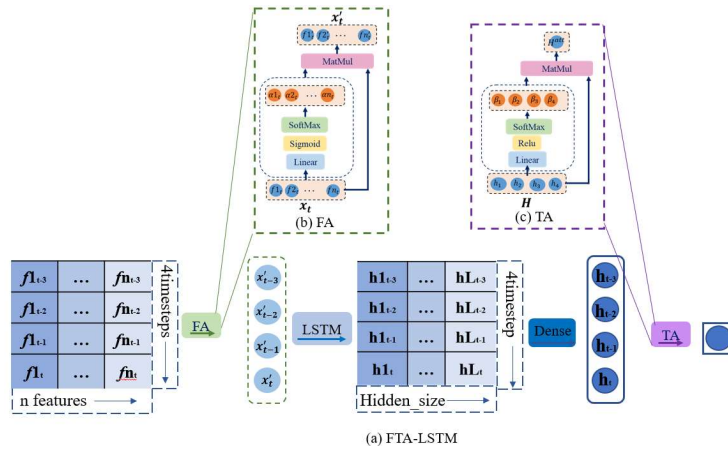



Figure 4. Framework of the proposed FA-LSTM hybrid models (a), the feature attention mechanism (FA) (b), and the temporal attention mechanism (TA) (c), inspired by Ding et al.(2020).

FA-LSTM:

FA-LSTM applies an attention mechanism to assign weights for distinct features in the input vector. In this study, for soil moisture prediction, the feature attention mechanism in FA-LSTM processes the input vector $I, \{x_{t-3}, \dots, x_t\}$, where $x_t = \{f1_t, f2_t, \dots, fn_t\}$ and generate the weighted output $\{x'_{t-3}, \dots, x'_t\}$. Through the attention mechanism, the output x'_t remains the same dimension size as the input x_t. The feature attention importance weight α_t and attention mechanism output x'_t are defined as follows:

$$\alpha_t = FA(x_t) \tag{12}$$

$$x'_t = \alpha_t \otimes x_t \tag{13}$$

Fig. 4b also shows the operation of the feature attention mechanism. The FA-LSTM model consists of an LSTM and a feature attention mechanism for input preprocessing, as detailed in Table A1.

TA-LSTM:

TA-LSTM utilizes the temporal attention mechanism to weigh the importance of LSTM output vectors across time steps. This enables the model to concentrate on the most relevant hidden states, potentially enhancing its performance on tasks that involve temporal modeling. The temporal attention mechanism is shown in Fig. 4c. In our work, the output vector H^{att}, which is obtained through the temporal attention

mechanism in TA-LSTM, is the weighted sum of all states in $\boldsymbol{H}, \{\boldsymbol{h}_{t-3}, \boldsymbol{h}_{t-2}, \boldsymbol{h}_{t-1}, \boldsymbol{h}_t\}$. The temporal

attention weight β and attention mechanism output $\boldsymbol{H}^{att}$ can be defined as:

$$\beta = TA(\boldsymbol{H}) \tag{14}$$

$$\boldsymbol{H}^{att} = \sum_{i=1}^{4} \beta_i \otimes \boldsymbol{h}_i \tag{15}$$

Compared to LSTM, the difference with TA-LSTM lies in the post-processing of the LSTM output.

LSTM utilizes the last hidden state output for prediction, while TA-LSTM employs temporal weighting

to utilize all hidden state outputs. Table A1 contains the network structure and parameters information.

FTA-LSTM:

FTA-LSTM is the model that combines both feature and temporal attention mechanisms, as illustrated

in Fig. 4a. It applies feature attention mechanism before the LSTM layer to assign weights for the input

features, and the temporal attention mechanism after the LSTM layer to weigh the importance of the

LSTM output vectors of different time steps. The parameters of FTA-LSTM can be found in Table A1.

3.3 GAN-LSTM

GANs (Goodfellow et al., 2014) comprise a generator and a discriminator. The generator is designed

to generate predictions that are similar to the truth, while the discriminator tries to distinguish between

the truth and the predictions. The unique network structure and adversarial training of GANs make them

highly effective in various fields, particularly in dealing with fuzzy prediction (Jing et al., 2019). Thus,

GANs offer a promising way to predict soil moisture, potentially leading to accurate results in real

situations. For predicting soil moisture, the GAN-LSTM model is used, where the generator G employs

an LSTM model capable of processing time series data, and the discriminator D uses a single-layer

feedforward neural network, similar to the work of Li et al. (2020). Alternating adversarial training is

performed between G and D, meaning that one of them is trained while keeping the other one fixed. The

structure and training strategies of GAN-LSTM are shown in Fig. 5.

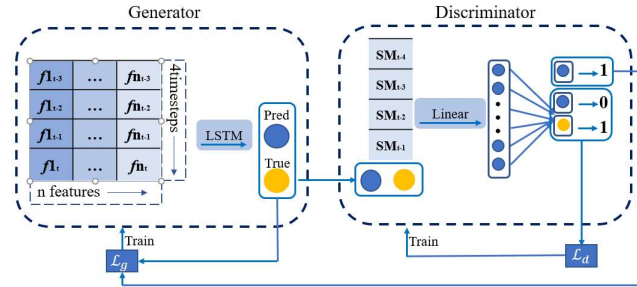



Figure 5. The framework of the proposed GAN-LSTM model.

360   The training objective of the discriminator D is to distinguish between predictions generated by the generator G and the ground truth, by minimizing the loss function $\mathcal{L}_d$. The binary cross-entropy loss is utilized as the similarity evaluation metric, with the objective of training D to output 1 when presented with ground truth as input and 0 when presented with predictions as input:

$$\mathcal{L}_d = \mathcal{L}_{bce}(d([SM_{t-4}, .., SM_{t-1}, y_t]), 1) + \mathcal{L}_{bce}(d([SM_{t-4}, .., SM_{t-1}, \hat{y}_t]), 0) \qquad (16)$$

where $\mathcal{L}_{bce}$ is the binary cross-entropy loss, which is defined as:

$$\mathcal{L}_{bce}(\hat{p}, p) = -plog(\hat{p}) - (1-p)log(1-\hat{p}) \qquad (17)$$

365   where p denotes the label (0 or 1) and $\hat{p}$ denotes the logit value between 0 and 1.

For generator G, there are two training objectives: first, to generate soil moisture dynamics predictions that are accurate and consistent with the truth, which is achieved by minimizing the fitting error of the soil moisture content data, denoted as $\mathcal{L}_{mse}$. Second, to deceive D, which is achieved by minimizing the binary cross-entropy loss $\mathcal{L}_{bce}$ between the predictions and the truth in D. The output of D should be

370   close to 1 when inputting the G predictions into D, ensuring that the prediction is close to the truth. Therefore, we train G by minimizing the following loss function $\mathcal{L}_g$:

$$\mathcal{L}_g = \mathcal{L}_{mse}(y_t, \hat{y}_t) + \lambda_{bce}\mathcal{L}_{bce}(d([SM_{t-4}, .., SM_{t-1}, \hat{y}_t]), 1) \qquad (18)$$

where λ_{bce} is the hyperparameter that controls the importance of the second term. Here we determine λ_{bce} to be 1×10^{-7} through manual testing. For adversarial training in our GAN-LSTM, the parameter update ratio of G and D in the model is 3:1, that is, every time G is updated (the learning rate is set to

375   0.0005), D will be updated 3 times (the learning rate is set to 0.001). The network structure parameters of GAN-LSTM are recorded in Table A1.

4. Results and Discussions

This study evaluates the performance of 3 machine learning methods and 10 deep learning models in
predicting soil moisture at 10 sites and 5 depths. To evaluate the model's ability for predicting over time
series, we examined forecasts for 1, 3, and 7 days ahead. When making predictions longer than 1 day,
we adopted iterative predictions. Three standard metrics, R^2, mean absolute error (MAE), and root mean
square error (RMSE) are used to evaluate the performance of the models. R^2 represents how well the
model captures the variability in data, while MAE and RMSE measure the accuracy of the model's
predictions. These metrics are calculated as follows:

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y}_i)^2} \tag{19}$$

$$MAE = \frac{\sum_{i=1}^{N}|y_i - \hat{y}_i|}{N} \tag{20}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{N}} \tag{21}$$

where y_i denotes the ground truth; $\hat{y}_i$ denotes the model prediction, $\bar{y}_i$ denotes the mean of the
ground truth, and N denotes the sample size.

The collected data in Section 2 is split into training, validation, and test sets in a 6:2:2 ratio in time
order. The training set is used to train the models with a learning rate of 0.001 unless stated otherwise.
We train the deep learning models for at least 1500 epochs, with a batch size of 50. In each epoch, 20
batches are used for training. The validation set is employed to determine whether the deep learning
model should be updated. If the trained model performs worse on the validation set compared to the
previous model, the previous model is retained. Finally, the test set is utilized to evaluate and compare
the accuracy of the trained models. To ensure statistical robustness, each final result is obtained by
averaging the outcomes of 25 repetitions of the training process.

4.1 Comparisons of Machine Learning and LSTM

This section compares the machine learning models with the deep learning model, represented by
LSTM. Table 2 summarizes the R^2 between the soil moisture predictions of the three machine learning
methods and the ground truth at ten sites and five depths for the following 1, 3, and 7 days. The results
show that all three methods perform well on short-term (1-3 days) soil moisture forecasts, but their

performance tends to diverge when predicting at the lead time of 7 days. Among these three, RF is the most stable and best-performing model.

Fig. 10(a-e) compares the RMSE of the soil moisture predictions of the machine learning models and
405   LSTM at different depths for 1, 3, and 7 days ahead at Cape-Charles. It reveals that LSTM outperforms the three machine learning models in terms of prediction accuracy and stability, which suggests that deep learning has a better capability of processing time series data for soil moisture dynamics simulation than traditional machine learning.

Machine learning models are limited in handling inputs from multiple time steps when processing time
410   series data. Therefore, while they exhibit proficiency in short-term predictions, they may not perform well in long-term prediction tasks and demonstrate comparatively lower accuracy and stability than deep learning models. Nevertheless, a notable advantage of machine learning models is that they require little training time, enabling rapid deployment, which incurs lower computational costs compared to deep learning models.

415

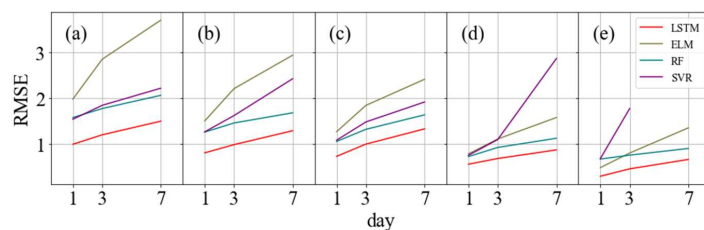



Figure 6. Top: RMSE comparisons between RF, ELM, SVR, and LSTM at the Cape site at 5 depths: 0.05m(a), 0.10m(b), 0.20m(c), 0.50m(d), 1.00m(e).

420   **Table 2.** The values of R^2 between the predictions (1, 3, and 7 days) of RF, ELM, and SVR and the ground truth for ten sites at five depths.

depth/m	RF R^2			ELM R^2			SVR R^2		
	1d	3d	7d	1d	3d	7d	1d	3d	7d
0.05	**0.871**	**0.783**	**0.659**	0.816	0.660	0.488	0.874	0.767	0.626
0.10	**0.881**	**0.812**	**0.708**	0.863	0.718	0.552	0.874	0.753	0.548
0.20	**0.878**	**0.813**	**0.713**	0.874	0.689	0.536	0.865	0.687	0.609
0.50	**0.892**	**0.827**	**0.722**	0.875	0.730	0.586	0.832	0.599	0.381
1.00	0.812	0.754	**0.692**	**0.955**	**0.856**	0.521	0.719	0.688	0.378

4.2 Comparisons of 1D-CNN, LSTM and Encoder

In this section, we conduct a comparative analysis of three basic deep learning networks. We evaluate
their prediction performance by assessing both prediction accuracy and computational costs. The values
of R^2 between the soil moisture predictions generated by the three models and the ground truth across
ten sites and five depths are presented in Table 3. Additionally, Fig. 7(a-e) displays the RMSE for soil
moisture predictions at the LittleRiver site.

The results reveal that the LSTM model achieves the highest prediction accuracy, followed by the 1D-
CNN model and subsequently the Encoder model. Notably, LSTM and Encoder are more stable when
making long-term or deep soil moisture predictions, while 1D-CNN is better suited for short-term and
shallow prediction tasks. This aligns with the inherent characteristics of the three models. In essence,
LSTM is designed to model temporal dependencies in sequence data, emphasizing global features.
Encoder operates by modeling relationships in input time series without iterations and highlights
important features by self-attention weighting. These characteristics prevent overfitting in the LSTM and
Encoder, resulting in stability in long-term predictions. In contrast, 1D-CNN excels at extracting and
expressing local features, which facilitates it to capture the connections between subtle feature changes
and their corresponding outcomes. This capability allows for adaptation to shallow soil moisture
prediction tasks with significant variations.

Fig. 7g shows the training epochs required for each model, while Fig. 7h illustrates the time taken for
100 epochs. The 1D-CNN demonstrates the fastest training speed and achieves early convergence.
Conversely, LSTM shows slower training speed attributed to its iterations. The Encoder trains quickly
but converges at a slower pace than LSTM, resulting in a similar total training time. In summary, although
1D-CNN offers the lowest computational costs, LSTM has been proven to be the most appropriate for
soil moisture prediction tasks among the three with the highest accuracy.

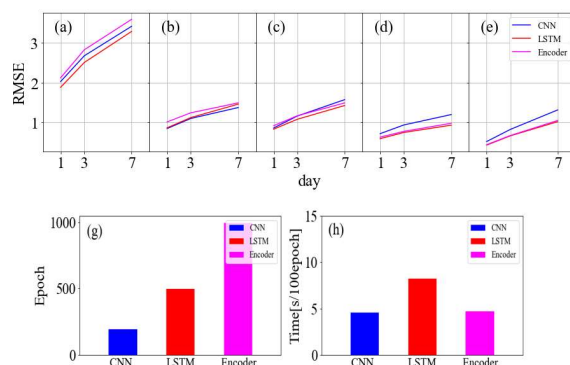



Figure 7. Top: RMSE comparisons between CNN, LSTM, and Encoder at the LittleRiver site at 5 depths: 0.05m(a), 0.10m(b), 0.20m(c), 0.50m(d), 1.00m(e). Bottom: comparisons of the training epoch (g) and training time (h) for three models.

450

Table 3. The values of R^2 between the predictions (1, 3, and 7 days) generated by CNN, LSTM and Encoder, and the ground truth across ten sites at five depths.

depth/m	CNN R^2			LSTM R^2			Encoder R^2		
	1d	3d	7d	1d	3d	7d	1d	3d	7d
0.05	0.907	0.822	0.685	**0.916**	**0.841**	**0.719**	0.896	0.819	0.693
0.10	**0.931**	**0.866**	**0.762**	0.924	0.854	0.753	0.909	0.844	0.745
0.20	**0.939**	**0.870**	0.744	0.937	0.856	0.739	0.926	0.866	**0.767**
0.50	0.920	0.828	0.686	**0.941**	**0.881**	0.754	0.928	0.871	**0.775**
1.00	0.878	0.728	0.348	**0.951**	**0.895**	**0.760**	0.934	0.856	0.679

4.3 Comparisons of CNN and LSTM Hybrid Models

455   This section compares the three CNN and LSTM hybrid models (LSTM-CNN, CNN-LSTM, and CNN-with-LSTM) across 10 sites in terms of prediction accuracy and computational costs. Table 4 presents the e R^2 values between the soil moisture predictions generated by the three hybrid models and the ground truth across ten sites at five depths. It can be observed that the prediction accuracy of these models is comparable, with LSTM-CNN slightly outperforming the others. Moreover, Fig. 8(a-e) shows

460   the RMSE results of hybrid models and LSTM at the Spickard site, indicating that the hybrid models do not exhibit obvious advantages over the standard LSTM.

Specifically, the three models are hybrids of CNN and LSTM with varying incorporation degrees. According to their combination ways, we can infer that the models excel in handling different types of

data and place different emphases on data characteristics. The CNN-LSTM appears to prioritize local

465 features and model long-distance dependencies, while LSTM-CNN focuses on global features and

context information. CNN-with-LSTM simultaneously considers both local features and temporal

information for predictions. These integrations increase the complexity and enhance the expression

capacities of models, but their applications should depend on the input data and prediction task. In the

case of soil moisture prediction, the benefits of this combination approach are not significant.

470   Fig. 8g and Fig. 8h display the computational costs of the three hybrid models. It is evident that the

CNN-LSTM shows the fastest training speed and the lowest computational costs, owing to its

convolution layers for input data pre-processing. Besides, the computational costs of LSTM-CNN are

higher than CNN-with-LSTM. Overall, compared to LSTM and 1D-CNN, we could draw the conclusion

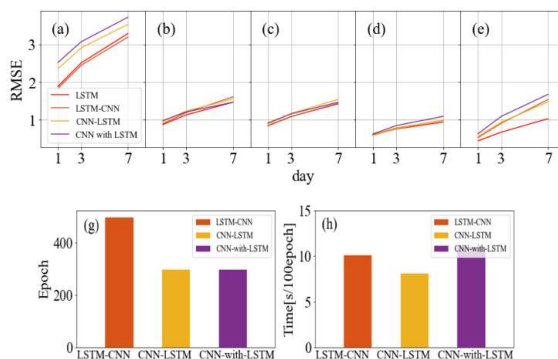
that the hybrid models have limited practical values in soil moisture prediction.

475

Figure 8. Top: RMSE comparisons between LSTM-CNN, CNN-LSTM and CNN-with-LSTM at the Spickard site

at 5 depths: 0.05m(a), 0.10m(b), 0.20m(c), 0.50m(d), 1.00m(e). Bottom: comparisons of the training epoch (g) and

training time (h) for three models.

480   **Table 4**. The values of R^2 between the predictions (1, 3, and 7 days) of LSTM-CNN, CNN-LSTM, and CNN-with-

LSTM and the ground truth for ten sites at five depths

depth/m	LSTM-CNN R^2			CNN-LSTM R^2			CNN-with-LSTM R^2		
	1d	3d	7d	1d	3d	7d	1d	3d	7d
0.05	**0.907**	**0.827**	**0.701**	0.899	0.815	0.689	0.899	0.821	0.694
0.10	**0.926**	**0.854**	**0.737**	0.920	0.837	0.705	0.905	0.828	0.720
0.20	**0.931**	0.838	0.703	0.923	**0.842**	0.712	0.921	0.842	**0.722**
0.50	**0.933**	0.856	0.719	0.926	**0.857**	**0.741**	0.920	0.851	0.734
1.00	**0.926**	**0.823**	**0.581**	0.901	0.795	0.570	0.897	0.788	0.549

4.4 Comparisons of Attention Mechanisms and LSTM Hybrid Models

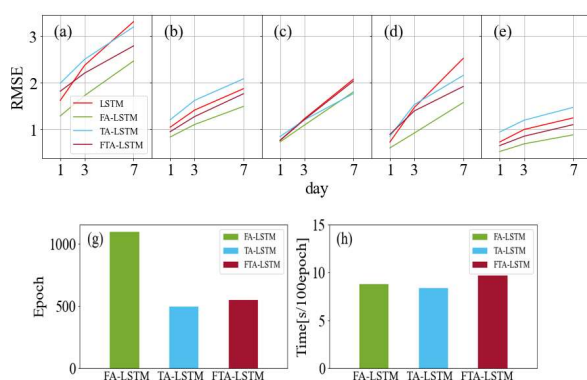
485 To investigate the impact of different attention mechanisms on models, this section compares these three models: FA-LSTM, TA-LSTM, and FTA-LSTM. Fig. 9(a-e) displays the RMSE values of the soil moisture predictions for 1, 3, and 7 days ahead generated by these three models and the standard LSTM at Weslaco. Table 5 records the values of R^2 between the soil moisture predictions of three models and the ground truth across ten sites and five depths.

490

Figure 9. Top: RMSE comparisons between FA-LSTM, TA-LSTM, and FTA-LSTM at the Weslaco site at 5 depths: 0.05m(a), 0.10m(b), 0.20m(c), 0.50m(d), 1.00m(e). Bottom: comparisons of the training epoch (g) and training time (h) for three models.

495 **Table 5**. The values of R^2 between the predictions (1, 3, and 7 days) of FA-LSTM, TA-LSTM, and FTA-LSTM and the ground truth for ten sites at five depths.

depth/m	FA-LSTM R^2			TA-LSTM R^2			FTA-LSTM R^2		
	1d	3d	7d	1d	3d	7d	1d	3d	7d
0.05	**0.918**	**0.856**	**0.746**	0.903	0.825	0.700	0.910	0.845	0.736
0.10	**0.932**	**0.871**	**0.783**	0.908	0.820	0.714	0.919	0.848	0.761
0.20	**0.945**	**0.887**	**0.794**	0.930	0.856	0.718	0.920	0.841	0.711
0.50	**0.949**	**0.908**	**0.824**	0.927	0.865	0.754	0.927	0.874	0.775
1.00	**0.959**	**0.916**	**0.805**	0.910	0.815	0.577	0.910	0.827	0.583

Based on the results, the prediction accuracy of the three models ranked from high to low is FA-LSTM, FTA-LSTM, and TA-LSTM in most situations. It can be found that the feature attention mechanism has

500 a stable gain effect on LSTM, potentially because it assigns the appropriate feature importance weights

to various influencing factors, especially in deep soil moisture prediction tasks. On the contrary, the improvement of the temporal attention mechanism is not evident and may lead to deterioration. TA-LSTM differs from LSTM in its output post-processing, as it is trained to weigh the LSTM output at each time step to make predictions. The reason why TA-LSTM is worse may be that LSTM already encodes

505 enough past features for predictions in the last hidden state. Moreover, the FTA-LSTM model, which combines both feature and temporal attention mechanisms, is the most complex but not necessarily the optimal one among the three. From the results, we can also infer the effective feature learning ability of attention mechanisms.

According to Fig. 9(g-h), attention mechanisms introduce some acceptable computational costs.

510 Notably, FA-LSTM requires more training steps to reach convergence. However, despite this computational requirement, we believe that the implementation of FA-LSTM is still advantageous for soil moisture prediction tasks.

Fig. 10 provides visualizations of the input feature importance and temporal importance weights learned by FA-LSTM and TA-LSTM for soil moisture prediction at the AAMU site across 5 depths. The

515 feature importance in Fig. 10(a-e) shows a reasonable adaptation to the varying depth, demonstrating the effective feature selection capability of attention mechanisms. Moreover, the temporal importance in Fig. 10(f-j) indicates the high utilization of recent temporal features, which is consistent with the real situations. This indicates the effective feature learning capacity of attention mechanisms. What's more, these results contribute to a deeper understanding of the utilization mechanisms of feature and temporal

520 information within the model.

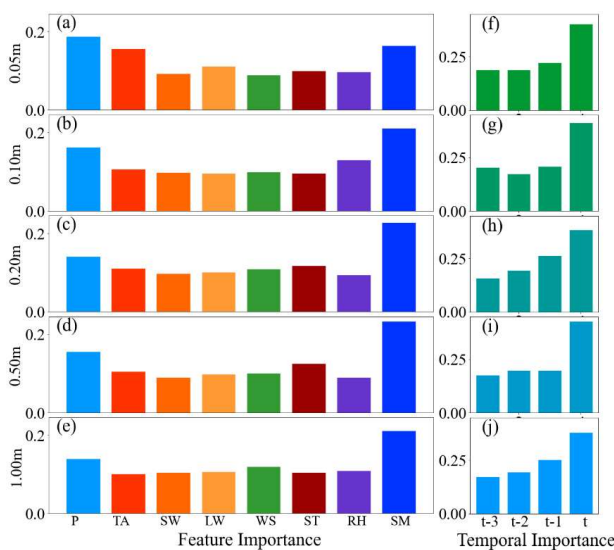



Figure 10. Feature importance and temporal importance for soil moisture prediction at the AAMU site across 5 depths.

4.5 Comparisons of GAN-LSTM and LSTM

In this section, we evaluate the impact of the GAN structure and adversarial training strategy on the standard LSTM model. LSTM and GAN-LSTM for soil moisture prediction are compared. The R^2 values for the following 1, 3, and 7 days across ten sites at different depths are recorded in Table 6. Fig. 11(a-e) shows the RMSE results of LSTM and GAN-LSTM at the Weslaco site.

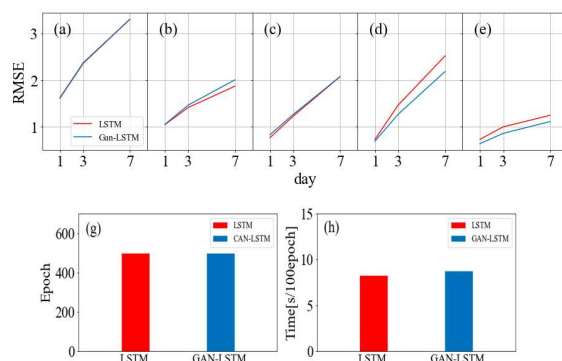
The results demonstrate that the GAN-LSTM achieves better performance than the standard LSTM in most situations, particularly in long-term prediction tasks (3-7 days). The application of GAN structure and training strategies enhances the prediction accuracy of LSTM. The adversarial training of GAN-LSTM allows the model to not only learn from the data itself but also extract additional information embedded in the data. This helps address performance degradations due to overfitting on data mean square error. We can regard this training strategy as a general principle to enhance the performance of neural networks. However, the selection of hyperparameters in the loss function of GAN is crucial and currently requires manual adjustments. In future work, adaptive methods can be adopted to automatically adjust the GAN-LSTM loss function to increase training flexibility and prediction accuracy.

540

Figure 11. Top: RMSE comparisons between LSTM and GAN-LSTM at the Weslaco site at 5 depths: 0.05m(a), 0.10m(b), 0.20m(c), 0.50m(d), 1.00m(e). Bottom: comparisons of the training epoch (g) and training time (h).

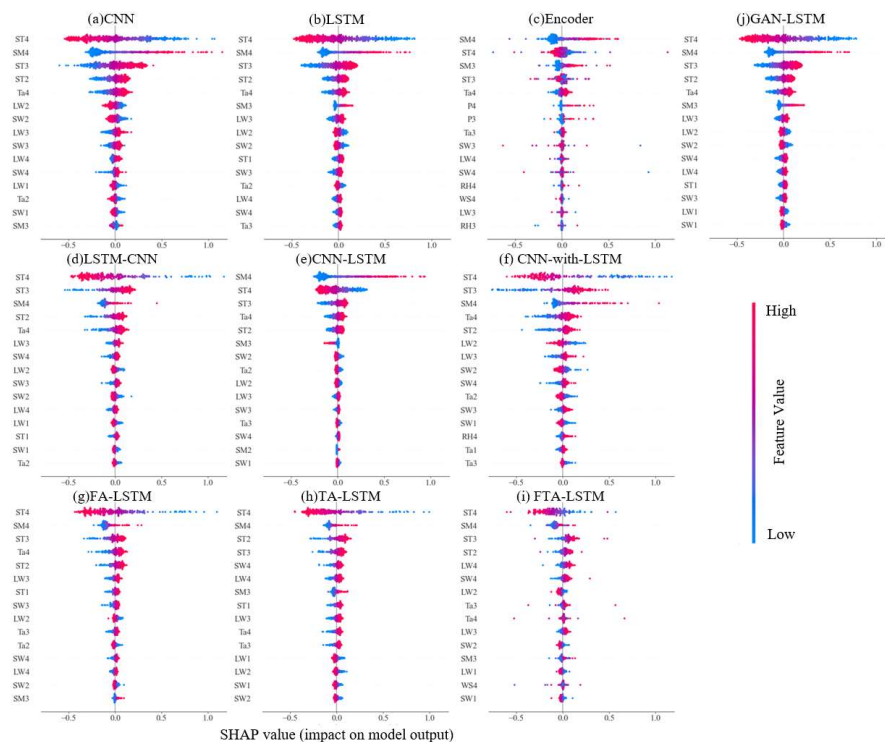
Table 6. The values of R^2 between the predictions (1, 3, and 7 days) generated by LSTM and GAN-LSTM and the

545   ground truth across ten sites at five depths.

depth/m	LSTM R^2			GAN-LSTM R^2		
	1d	3d	7d	1d	3d	7d
0.05	**0.916**	**0.841**	**0.719**	0.916	0.840	0.715
0.10	0.924	0.854	0.753	**0.926**	**0.856**	**0.755**
0.20	0.937	0.856	0.739	**0.938**	**0.865**	**0.754**
0.50	0.941	0.881	0.754	**0.942**	**0.889**	**0.777**
1.00	0.951	0.895	0.760	**0.961**	**0.914**	**0.803**

Based on the computational cost comparisons presented in Fig. 11(g-h), both LSTM and GAN-LSTM exhibit similar computational costs. Consequently, in most scenarios, it is advisable to apply the GAN-LSTM model to predict soil moisture dynamics. It improves the stability and prediction ability of the

550   model without imposing a significant increase in computational costs.

4.6 Visualization Analysis

555   **Figure 12.** SHAP summary plots for ten deep learning models. The samples are from the test set of the Monahans site at 0.0500m.

In this study, we employ the SHAP (Lundberg et al., 2018) to quantify the contributions of input features to investigate the distinct mechanisms of data utilization in different network structures. Brief

560   introductions to SHAP are provided in Appendix B. Fig. 12 illustrates the SHAP summary plots of these ten deep learning models utilizing samples from the test set of the Monahans site. The y-axis represents the input features ranked by importance. Each point shows the Shapley value of a specific feature in a sample, with the color indicating the value of the input feature. The plot clearly shows the identified main influential factors and established correlations between input features and soil moisture by the models.

565   Fig. 12(a-c) displays the Shapley values of three basic deep learning models: CNN, LSTM, and Encoder. It can be observed that CNN shows a broader range of Shapley values compared to the others, indicating its greater feature expression capacity. This suggests that CNN focuses more on specific local

features, while LSTM emphasizes capturing global features. However, both CNN and LSTM tend to learn incorrect features and correlations. For instance, the learned positive correlation between the feature

570 ST3 and soil moisture is contrary to the facts. The Encoder model, which aggregates features from all other inputs, appears to perform better in this aspect. Although the Shapley value of Encoder exhibits the lowest range, the important features Encoder identifies are derived from the recent input time series, which aligns better with real situations. This reflects the effective feature learning ability of attention mechanisms. Overall, each of these models—CNN, LSTM, and Encoder—possesses unique advantages

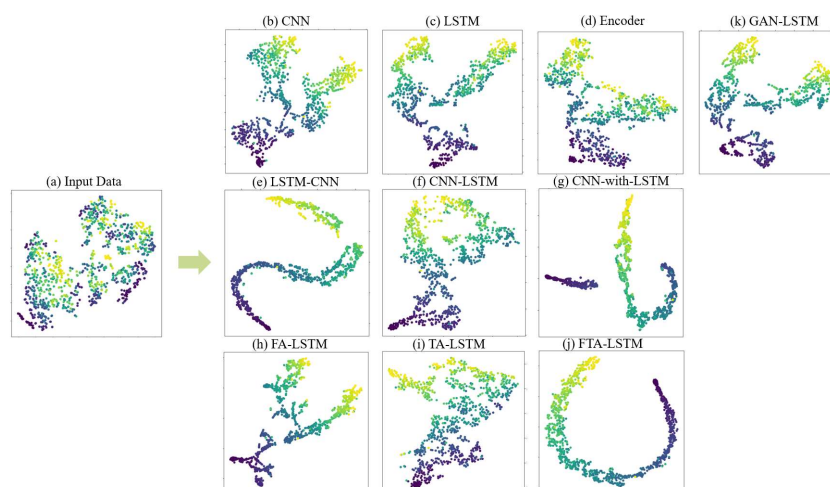
575 in terms of data utilization.

Fig. 12(d-f) compares the hybrid models of CNNs and LSTMs. The CNN-LSTM keeps high Shapley values in important features while showing minimal response to the others. This suggests that CNN-LSTM tends to sequentially process the extracted crucial features, enabling itself to effectively capture both local data features and long-range dependencies, resembling more the CNN. LSTM-CNN shows

580 similar Shapley values to the LSTM. By employing CNN to extract sequential modeling features, LSTM-CNN emphases more on global features, resembling the characteristics of the LSTM. The Shapley value of the CNN-with-LSTM is the highest, displaying a heightened sensitivity to feature perturbations. This can be attributed to the repeated utilization of features in parallel networks. These three models represent different degrees of fusion between CNNs and LSTMs, and the hybrid architecture design depends on

585 the specific task requirements and data characteristics.

In the case of hybrid models that integrate attention mechanisms with LSTM, FA-LSTM, TA-LSTM, and FTA-LSTM, their Shapley values in Fig. 12(g-i) are found to differ slightly from that of LSTM. Considering the attention importance analysis discussed in section 4.4, we can infer that the attention mechanisms introduce slight adjustments to the time and feature attributions on the basis of the LSTM.

590 Fig. 12 (j) also presents the Shapley value of GAN-LSTM. Through the resembled Shapley value, we can infer that the GAN-LSTM model introduces slight modifications during adversarial training, influencing some feature contributions to improve the prediction accuracy of the LSTM model. This demonstrates that adversarial training strategies contribute to the refinement and enhancement in models.

595 **Figure 13.** The t-SNE visualizations of the original input data (a) and encoded hidden states of 10 models (b-k) obtained from the test set of the LittleRiver site. The colors of the points indicate the corresponding soil moisture content values.

Besides, t-SNE (Van der Maaten and Hinton, 2008), a dimension reduction and visualization method
600 is employed to discover the structure and patterns in the high-dimensional data. When mapping data onto a two-dimensional space, t-SNE retains the relative distance relationships between the original data points, ensuring that similar samples are mapped closer to each other. The details of t-SNE can be found in Appendix B. Fig. 13 presents the t-SNE visualizations of the input data and the encoded hidden states from 10 models. The color of each point corresponds to the soil moisture content value. It is evident that
605 through training, the low-dimensional embeddings of the encoded hidden states gradually transition from an initially irregular pattern to a more structured shape. However, the visualization shapes vary across the different models. Notably, LSTM-CNN, CNN-with-LSTM, and FTA-LSTM exhibit distinct clustering patterns in their embedding plots. Additionally, both GAN-LSTM and FA-LSTM demonstrate a tendency to gather the data points more closely, based on LSTM. These results suggest that different
610 models possess varying capabilities when learning the input data features. To gain a more comprehensive understanding of their differences, further research is warranted in the future.

5. Conclusions

In this research, we have conducted a comprehensive analysis of traditional machine learning models and various deep learning models for soil moisture predictions across different sites at 5 depths. Based on our comparisons of these models, we draw the following conclusions:

In traditional machine learning, RF seems to be the most stable method in soil moisture prediction tasks. However, deep learning models have been found to possess stronger capabilities in processing time series data for better predictions. Among the three basic deep learning models, LSTM demonstrates a high level of accuracy because of its temporal information modeling capability, while 1D-CNN exhibits the lowest computational cost. Encoder also shows stable long-term forecasting ability. When considering the hybrid models, three combinations of CNN and LSTM did not enhance the prediction abilities in this task. Despite the attractiveness of hybridizing the benefits of CNN and LSTM, the results did not find notable advantages in soil moisture prediction in terms of accuracy and computational costs.

However, the feature attention mechanism has a constant positive effect on LSTM, while temporal attention mechanisms have little significance. In addition, combining generative adversarial network structures and training strategies into LSTM models (GAN-LSTM) has been found to improve prediction accuracy, especially in long-term predictions. To summarize, FA-LSTM and GAN-LSTM are found to be the most stable and effective models for soil moisture prediction. Furthermore, this study attempts to provide a thorough analysis of models' performances and advance the understanding of machine learning in soil moisture prediction. Through the Shapley analysis, we can infer the different data utilization ways of the 10 models. Besides, the t-SNE visualizations illustrate the varying encoding capabilities in different models.

The results emphasize the importance of appropriate and effective neural network structure design for a given task. For soil moisture prediction, several principles of effective network design can be concluded. Firstly, leveraging the temporal modeling capability of LSTM is well-suited for soil moisture forecasting. Secondly, incorporating attention mechanisms properly facilitates efficient feature learning. The feature selection capability of attention mechanisms has been proven through the performance of the Encoder and the attention mechanisms and LSTM hybrid models. Lastly, applying special GAN structures and adversarial training strategies in models helps extract additional information embedded within data, which also potential for better soil moisture dynamics simulation.

This study provides a reference and lay the groundwork for the development of specialized deep learning models for soil moisture dynamics simulation. However, although data-driven models have

645 shown satisfactory performance, they cannot make long-term predictions precisely due to their lack of physical laws. In the future, the integration of known physical laws with deep learning models will become a promising research direction for soil moisture dynamics simulation.

ACKNOWLEDGEMENTS

This work was supported by the National Key Research and Development Program of China
650 (2021YFC3201203), the Priority Research and Development Projects for Ningxia (Grant No.2021BBF02027) and the National Natural Science Foundation of China (Grant 51979200 and 52179038).

CODE/DATA AVAILABILITY

655 The data and codes used in this paper are available on the website (https://doi.org/10.5281/zenodo.8103621).

AUTHOR CONTRIBUTION

Yanling Wang: Conceptualization, Methodology, Software, Writing- Original draft preparation.
660 **Liangsheng Shi:** Writing-Reviewing and Editing, Supervision. **Yaan Hu:** Writing-Reviewing and Editing. **Xiaolong Hu:** Methodology, Writing-Reviewing and Editing. **Wenxiang Song:** Methodology, Writing-Reviewing and Editing. **Lijun Wang:** Writing-Reviewing and Editing.

COMPETING INTERESTS

665 The contact author has declared that none of the authors has any competing interests.

Appendix A. Parameters Used in machine learning and deep learning models

Table A1. Parameters settings of the deep learning models.

Network Type	Layers	Kernel_size	Hidden_size (L)	Activation function
1D-CNN	Convolution	2	32	Tanh
	Convolution	2	64	Tanh
	Flatten			
	Fully-connected		1	Tanh
CNN-LSTM	Convolutional	2	32	Tanh
	Convolutional	2	64	Tanh
	LSTM		16	Sigmoid, Tanh
LSTM -CNN	LSTM		16	Sigmoid, Tanh
	Convolutional	3	32	Tanh
	Convolutional	3	64	Tanh
	Flatten			
	Fully-connected		1	Tanh
CNN-with-LSTM	CNN			
	Convolutional	3	32	Tanh
	Convolutional	3	64	Tanh
	Flatten			
	Fully-connected		1	Tanh
	LSTM			
	LSTM		16	Sigmoid, Tanh
	Fully-connected		1	Tanh
	CNN-with-LSTM			
	CONCAT			
	Fully-connected		10	Tanh
	Fully-connected		1	Tanh
FA-LSTM	F-Attention		8	Sigmoid
	LSTM		16	Sigmoid, Tanh
TA-LSTM	LSTM		16	Sigmoid
	T-Attention		8	Relu
FTA-LSTM	F-Attention		8	Sigmoid
	LSTM		16	Sigmoid, Tanh
	T-Attention		8	Relu
	Fully-connected		1	Tanh
GAN-LSTM	Generator			
	LSTM		16	Sigmoid, Tanh
	Fully-connected		1	Tanh
	Discriminator			
	Fully-connected		1	Sigmoid

670   RF: the default parameter values in RandomForestRegressor of the sklearn library

SVR: C=1.0, ε=0.1, kernel γ = 'poly'

ELM: Hidden_size (L) =20

LSTM: num_layers = 2, Hidden_size(L) = 16

Encoder: d_k=d_v=4, d_model=feature=8, d_ff=20, n_heads=1

675

Appendix B. Shapley additive explanations (SHAP)

SHAP (Lundberg et al., 2018) is a game theoretic approach to explain the output of machine learning models. It measures the impact of the input feature on the prediction of an individual sample. SHAP employs the additive feature attribution method to provide a specific explanation:

$$f(x) = g(x') = \phi_0 + \sum_{i=1}^{M} \phi_i x' \tag{B1}$$

680 where $f(x)$ denotes the original model, $g(x)$ represents the explanation model with simplified input $x', x' \in \{0,1\}^M$, M is the number of input features; through a mapping function, $x = h_x(x')$; ϕ_i denotes the feature attribution of feature i. The explanation model g has a unique solution:

$$\phi_i(f, x) = \sum_{z' \subseteq x'} \frac{|z'|! (M - |z'| - 1)!}{M!} [f_x(z') - f_x(z' \backslash i)] \tag{B2}$$

where $|z'|$ is the non-zero entries number in z', $z' \subseteq x'$; $f(x') = f(h_x(z')) = E[f(z)|z_s]$, S denotes the non-zero indexes set in z'.

685

Appendix C. The t-Distributed Stochastic Neighbor Embedding (t-SNE) visualization

The t-SNE is a nonlinear dimension reduction technique that assumes the presence of a low-dimensional nonlinear manifold within the high-dimensional data. Its primary task is to bring similar neighboring points close together in the low-dimensional representation. The working process of t-SNE

690 can be divided into several steps: calculate the similarity between data points in high-dimensional space, and then calculate the corresponding probability of points in low-dimensional space. The similarity of points is calculated as conditional probability. If interested, more information can be found in the work of Van der Maaten and Hinton (2008). The following is the formula for calculating the similarity P_{ij} and probability q_{ij} of the points:

695 The similarity between data points in high-dimensional space:

$$P_{ij} = (P_{j|i} + P_{i|j})/2N \tag{C1}$$

Corresponding probability of points in low-dimensional space:

$$q_{ij} = \frac{\left(1 + \|y_i - y_j\|^2\right)^{-1}}{\sum_{k \neq l}(1 + \|y_k - y_l\|^2)^{-1}} \tag{C2}$$

where $P_{i|j}$ denotes the conditional probability of point i picking point j as its neighbor if neighbors are chosen according to their probability density under a Gaussian distribution centered at i, and N denotes the data points number. y_i denotes the low-dimensional representation of point i, and $\|y_i - y_j\|$ denotes the Euclidean distance between y_i and y_j.

Reference:

Abdel-Hamid, O., Mohamed, A. R., Jiang, H., Deng, L., Penn, G., and Yu, D.: Convolutional neural networks for speech recognition, IEEE Trans. Audio, Speech Lang. Process., 22, 1533–1545, https://doi.org/10.1109/TASLP.2014.2339736, 2014.

Ahmed, A. A. M., Deo, R. C., Ghahramani, A., Raj, N., Feng, Q., Yin, Z., and Yang, L.: LSTM integrated with Boruta-random forest optimiser for soil moisture estimation under RCP4.5 and RCP8.5 global warming scenarios, Springer Berlin Heidelberg, 1851–1881 pp., https://doi.org/10.1007/s00477-021-01969-3, 2021.

Ajit, A., Acharya, K., and Samanta, A.: A Review of Convolutional Neural Networks, Int. Conf. Emerg. Trends Inf. Technol. Eng. ic-ETITE 2020, 1–5, https://doi.org/10.1109/ic-ETITE47903.2020.049, 2020.

Albawi, S., Mohammed, T. A., and Al-Zawi, S.: Understanding of a convolutional neural network, Proc. 2017 Int. Conf. Eng. Technol. ICET 2017, 2018-Janua, 1–6, https://doi.org/10.1109/ICEngTechnol.2017.8308186, 2018.

Azhar, A. H., Perera, B. J. C., and Nabi, G.: A Simple Soil Moisture Simulation Model to Address Irrigation Water Management Issues, Mehran Univ. Res. J. Eng. Technol., 30, 193–206, 2011.

Breiman, L.: Random forests, Mach. Learn., 45, 5–32, 2001.

Cai, Y., Zheng, W., Zhang, X., Zhangzhong, L., and Xue, X.: Research on soil moisture prediction model based on deep learning, PLoS One, 14, 1–19, https://doi.org/10.1371/journal.pone.0214508, 2019.

Camporese, M., Daly, E., and Paniconi, C.: Catchment-scale Richards equation-based modeling of evapotranspiration via boundary condition switching and root water uptake schemes, Water Resour. Res., 51, 5756–5771, 2015.

Carranza, C., Nolet, C., Pezij, M., and van der Ploeg, M.: Root zone soil moisture estimation with Random Forest, J. Hydrol., 593, 125840, https://doi.org/10.1016/j.jhydrol.2020.125840, 2021.

725   Chen, Y., Li, L., Whiting, M., Chen, F., Sun, Z., Song, K., and Wang, Q.: Convolutional neural network model for soil moisture prediction and its transferability analysis based on laboratory Vis-NIR spectral data, Int. J. Appl. Earth Obs. Geoinf., 104, 102550, 2021.

Connor, J. T., Martin, R. D., and Atlas, L. E.: Recurrent Neural Networks and Robust Time Series Prediction, IEEE Trans. Neural Networks, 5, 240–254, https://doi.org/10.1109/72.279188, 1994.

730   Cortes, C. and Vapnik, V.: Support-vector networks, Mach. Learn., 20, 273–297, 1995.

Ding, Y., Zhu, Y., Wu, Y., Jun, F., and Cheng, Z.: Spatio-Temporal attention lstm model for flood forecasting, Proc. - 2019 IEEE Int. Congr. Cybermatics 12th IEEE Int. Conf. Internet Things, 15th IEEE Int. Conf. Green Comput. Commun. 12th IEEE Int. Conf. Cyber, Phys. So, 458–465, https://doi.org/10.1109/iThings/GreenCom/CPSCom/SmartData.2019.00095, 2019.

735   Ding, Y., Zhu, Y., Feng, J., Zhang, P., and Cheng, Z.: Interpretable spatio-temporal attention LSTM model for flood forecasting, Neurocomputing, 403, 348–359, https://doi.org/10.1016/j.neucom.2020.04.110, 2020.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N.: An Image is Worth 16x16 Words: 740   Transformers for Image Recognition at Scale, 2020.

Entin, J. K., Robock, A., Vinnikov, K. Y., Hollinger, S. E., Liu, S., and Namkhai, A.: Meteorologic i Land Surface, J. Geophys. Res., 105, 11865–11877, 2000.

Fang, K., Pan, M., and Shen, C.: The Value of SMAP for Long-Term Soil Moisture Estimation with the Help of Deep Learning, IEEE Trans. Geosci. Remote Sens., 57, 2221–2233, 745   https://doi.org/10.1109/TGRS.2018.2872131, 2019.

Gill, M. K., Asefa, T., Kemblowski, M. W., and McKee, M.: Soil moisture prediction using support vector machines, J. Am. Water Resour. Assoc., 42, 1033–1046, https://doi.org/10.1111/j.1752-1688.2006.tb04512.x, 2006.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and 750   Bengio, Y.: Generative adversarial nets, Adv. Neural Inf. Process. Syst., 27, 2014.

Goodfellow, I., Bengio, Y., and Courville, A.: Deep learning, MIT press, 2016.

Guswa, A. J., Celia, M. A., and Rodriguez-Iturbe, I.: Models of soil moisture dynamics in ecohydrology: A comparative study, Water Resour. Res., 38, 5-1-5–15, https://doi.org/10.1029/2001wr000826, 2002.

755   He, K., Zhang, X., Ren, S., and Sun, J.: Deep residual learning for image recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 770–778, 2016.

Heathman, G. C., Cosh, M. H., Merwade, V., and Han, E.: Multi-scale temporal stability analysis of surface and subsurface soil moisture within the Upper Cedar Creek Watershed, Indiana, Catena, 95, 91–103, https://doi.org/10.1016/j.catena.2012.03.008, 2012.

760   Hochreiter, S. and Schmidhuber, J.: Long short-term memory, Neural Comput., 9, 1735–1780, 1997.

Holzman, M., Rivas, R., Carmona, F., and Niclòs, R.: A method for soil moisture probes calibration and validation of satellite estimates, MethodsX, 4, 243–249, https://doi.org/10.1016/j.mex.2017.07.004, 2017.

Huang, G. Bin, Zhu, Q. Y., and Siew, C. K.: Extreme learning machine: Theory and applications,
765   Neurocomputing, 70, 489–501, https://doi.org/10.1016/j.neucom.2005.12.126, 2006.

Hummel, J. W., Sudduth, K. A., and Hollinger, S. E.: Soil moisture and organic matter prediction of surface and subsurface soils using an NIR soil sensor, Comput. Electron. Agric., 32, 149–165, https://doi.org/10.1016/S0168-1699(01)00163-6, 2001.

Hussain, D., Hussain, T., Khan, A. A., Naqvi, S. A. A., and Jamil, A.: A deep learning approach for
770   hydrological time-series prediction: A case study of Gilgit river basin, Earth Sci. Informatics, 13, 915–927, https://doi.org/10.1007/s12145-020-00477-2, 2020.

Jackson, S. H.: Comparison of calculated and measured volumetric water content at four field sites, Agric. Water Manag., 58, 209–222, https://doi.org/10.1016/S0378-3774(02)00078-1, 2003.

Jing, J. R., Li, Q., Ding, X. Y., Sun, N. L., Tang, R., and Cai, Y. L.: AENN: A GENERATIVE
775   ADVERSARIAL NEURAL NETWORK for WEATHER RADAR ECHO EXTRAPOLATION, Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci. - ISPRS Arch., 42, 89–94, https://doi.org/10.5194/isprs-archives-XLII-3-W9-89-2019, 2019.

Kamilaris, A. and Prenafeta-Boldú, F. X.: Deep learning in agriculture: A survey, Comput. Electron. Agric., 147, 70–90, https://doi.org/10.1016/j.compag.2018.02.016, 2018.

780   Kilinc, H. C. and Yurtsever, A.: Short-Term Streamflow Forecasting Using Hybrid Deep Learning Model Based on Grey Wolf Algorithm for Hydrological Time Series, Sustain., 14, https://doi.org/10.3390/su14063352, 2022.

Lecun, Y., Bengio, Y., and Hinton, G.: Deep learning, Nature, 521, 436–444, https://doi.org/10.1038/nature14539, 2015.

785   LeCun, Y.: Generalization and network design strategies, Connect. Perspect., 19, 18, 1989.

        Li, Q., Hao, H., Zhao, Y., Geng, Q., Liu, G., Zhang, Y., and Yu, F.: GANs-LSTM Model for Soil

        Temperature Estimation from Meteorological: A New Approach, IEEE Access, 8, 59427–59443,

        https://doi.org/10.1109/ACCESS.2020.2982996, 2020.

        Li, Q., Zhu, Y., Shangguan, W., Wang, X., Li, L., and Yu, F.: An attention-aware LSTM model for

790   soil   moisture   and   soil   temperature   prediction,   Geoderma,   409,   1–17,

        https://doi.org/10.1016/j.geoderma.2021.115651, 2022a.

        Li, Q., Li, Z., Shangguan, W., Wang, X., Li, L., and Yu, F.: Improving soil moisture prediction using

        a   novel   encoder-decoder   model   with   residual   learning,   Comput.   Electron.   Agric.,   195,

        https://doi.org/10.1016/j.compag.2022.106816, 2022b.

795   Liu, Y., Mei, L., and Ki, S. O.: Prediction of soil moisture based on Extreme Learning Machine for an

        apple orchard, CCIS 2014 - Proc. 2014 IEEE 3rd Int. Conf. Cloud Comput. Intell. Syst., 400–404,

        https://doi.org/10.1109/CCIS.2014.7175768, 2014.

        Lundberg, S. M., Erion, G. G., and Lee, S.-I.: Consistent Individualized Feature Attribution for Tree

        Ensembles, 2018.

800   Van der Maaten, L. and Hinton, G.: Visualizing data using t-SNE., J. Mach. Learn. Res., 9, 2008.

        Mikolov, T., Kombrink, S., Burget, L., Černocký, J., and Khudanpur, S.: Extensions of recurrent

        neural network language model, in: 2011 IEEE international conference on acoustics, speech and signal

        processing (ICASSP), 5528–5531, 2011.

        Patil, A. and Rane, M.: Convolutional Neural Networks: An Overview and Its Applications in Pattern

805   Recognition, Smart Innov. Syst. Technol., 195, 21–30, https://doi.org/10.1007/978-981-15-7078-0_3,

        2021.

        Pollack, J. B.: Recursive distributed representations, Artif. Intell., 46, 77–105, 1990.

        Prakash, S., Sharma, A., and Sahu, S. S.: Soil moisture prediction using machine learning, in: 2018

        Second International Conference on Inventive Communication and Computational Technologies

810   (ICICCT), 1–6, 2018.

        Prasad, R., Deo, R. C., Li, Y., and Maraseni, T.: Weekly soil moisture forecasting with multivariate

        sequential, ensemble empirical mode decomposition and Boruta-random forest hybridizer algorithm

        approach, Catena, 177, 149–166, https://doi.org/10.1016/j.catena.2019.02.012, 2019.

Qiu, Y., Fu, B., Wang, J., and Chen, L.: Spatiotemporal prediction of soil moisture content using

815   multiple-linear regression in a small catchment of the Loess Plateau, China, Catena, 54, 173–195,

https://doi.org/10.1016/S0341-8162(03)00064-X, 2003.

Ravuri, S., Lenc, K., Willson, M., Kangin, D., Lam, R., Mirowski, P., Fitzsimons, M., Athanassiadou,

M., Kashem, S., Madge, S., Prudden, R., Mandhane, A., Clark, A., Brock, A., Simonyan, K., Hadsell, R.,

Robinson, N., Clancy, E., Arribas, A., and Mohamed, S.: Skilful precipitation nowcasting using deep

820   generative models of radar, Nature, 597, 672–677, https://doi.org/10.1038/s41586-021-03854-z, 2021.

Sampathkumar, T., Pandian, B. J., Rangaswamy, M. V, Manickasundaram, P., and Jeyakumar, P.:

Influence of deficit irrigation on growth, yield and yield parameters of cotton–maize cropping sequence,

Agric. Water Manag., 130, 90–102, 2013.

Saxton, K. E., Johnson, H. P., and Shaw, R. H.: Modeling Evapotranspiration and Soil Moisture.,

825   Trans. Am. Soc. Agric. Eng., 17, 673–677, https://doi.org/10.13031/2013.36935, 1974.

Schmidhuber, J.: Deep Learning in neural networks: An overview, Neural Networks, 61, 85–117,

https://doi.org/10.1016/j.neunet.2014.09.003, 2015.

Semwal, V. B., Gupta, A., and Lalwani, P.: An optimized hybrid deep learning model using ensemble

learning approach for human walking activities recognition, J. Supercomput., 77, 12256–12279,

830   https://doi.org/10.1007/s11227-021-03768-7, 2021.

Severyn, A. and Moschitti, A.: UNITN: Training Deep Convolutional Neural Network for Twitter

Sentiment Classification, SemEval 2015 - 9th Int. Work. Semant. Eval. co-located with 2015 Conf. North

Am. Chapter Assoc. Comput. Linguist. Hum. Lang. Technol. NAACL-HLT 2015 - Proc., 464–469,

https://doi.org/10.18653/v1/s15-2079, 2015.

835   Shi, X., Chen, Z., Wang, H., Yeung, D. Y., Wong, W. K., and Woo, W. C.: Convolutional LSTM

network: A machine learning approach for precipitation nowcasting, Adv. Neural Inf. Process. Syst.,

2015-Janua, 802–810, 2015.

Simunek, J., Van Genuchten, M. T., and Sejna, M.: The HYDRUS-1D software package for simulating

the one-dimensional movement of water, heat, and multiple solutes in variably-saturated media, Univ.

840   California-Riverside Res. Reports, 3, 1–240, 2005.

Sutskever, I., Vinyals, O., and Le, Q. V: Sequence to sequence learning with neural networks, Adv.

Neural Inf. Process. Syst., 27, 2014.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I.: Attention is all you need, Adv. Neural Inf. Process. Syst., 30, 2017.

845   Vereecken, H., Huisman, J. A., Bogena, H., Vanderborght, J., Vrugt, J. A., and Hopmans, J. W.: On the value of soil moisture measurements in vadose zone hydrology: A review, Water Resour. Res., 46, 1–21, https://doi.org/10.1029/2008WR006829, 2008.

Vereecken, H., Amelung, W., Bauke, S. L., Bogena, H., Brüggemann, N., Montzka, C., Vanderborght, J., Bechtold, M., Blöschl, G., Carminati, A., Javaux, M., Konings, A. G., Kusche, J., Neuweiler, I., Or, D., Steele-Dunne, S., Verhoef, A., Young, M., and Zhang, Y.: Soil hydrology in the Earth system, Nat. Rev. Earth Environ., 3, 573–587, https://doi.org/10.1038/s43017-022-00324-6, 2022.

Verma, S. and Nema, M. K.: Development of an empirical model for sub-surface soil moisture estimation and variability assessment in a lesser Himalayan watershed, Model. Earth Syst. Environ., https://doi.org/10.1007/s40808-021-01316-z, 2021.

855   Xia, K., Huang, J., and Wang, H.: LSTM-CNN Architecture for Human Activity Recognition, IEEE Access, 8, 56855–56866, https://doi.org/10.1109/ACCESS.2020.2982225, 2020.

Yu, J., Zhang, X., Xu, L., Dong, J., and Zhangzhong, L.: A hybrid CNN-GRU model for predicting soil moisture in maize root zone, Agric. Water Manag., 245, 106649, https://doi.org/10.1016/j.agwat.2020.106649, 2021.

860