Development of a Drought Prediction System Based on Long Short-Term Memory Networks (LSTM)

Nian Zhang[1*], Xiangguang Dai[2], M. A. Ehsan[1], and Tolessa Deksissa[3]

[1]Department of Electrical and Computer Engineering, University of the District of Columbia, Washington, D.C., 20008, USA nzhang@udc.edu and mdamimul.ehsan@udc.edu
[2]Key Laboratory of Intelligent Information Processing and Control of Chongqing Municipal Institutions of Higher Education, Chongqing Three Gorges University, Chongqing, 404000, China daixiangguang@163.com
[3]Water Resources Research Institute (WRRI), University of the District of Columbia, Washington, D.C., 20008, USA tdeksissa@udc.edu

Abstract. As streamflow quantity and drought problem become increasingly severe, it's imperative than ever to seek next generation machine learning models and learning algorithms which can provide accurate prediction. Reliable prediction of drought variables such as precipitation, soil moisture, and streamflow has been a significant challenge for water resources professionals and water management districts due to their random and nonlinear nature. This paper proposes a long short-term memory networks (LSTM) based deep learning method to predict the historical monthly soil moisture time series data based on the MERRA-Land from 1980 to 2012. The proposed LSTM model learns to predict the value of the next time step at each time step of the time sequence. We also compare the predication accuracy when the network state is updated with the observed values and when the network state is updated with the predicted values. We find that the predictions are more accurate when updating the network state with the observed values instead of the predicted values. In addition, it demonstrated that the proposed method has much lower MSE than the autoregressive integrated moving average model (ARIMA) model and autoregressive model (AR) model.

Keywords: Convolutional Neural Networks, Long Short-Term Memory Networks (LSTM), Deep Learning, Time Series Prediction, Drought Prediction.

1 Introduction

The streamflows of a river basin may be near or below normal, influenced by lower than normal precipitation and much below normal soil moisture contents. If below average rainfall continues then further degradation is expected to occur. Monthly monitoring of a river basin will prepare for the possibility that serious drought conditions may develop in the future [1]. As drought and streamflow quantity problem become increasingly severe, it's imperative to provide an effective drought early warning system which uses the historical data to make prediction of the probability of flows dropping below drought trigger levels [2]. Reliable estimation of streamflow has been a significant challenge for water resources professionals and water management districts. This is very much essential to manage water supply, floods, and droughts efficiently. Streamflow characteristics are primarily governed by climatic and watershed characteristics. Over the last decade it has been recognized that climate is changing and there can be significant impacts on the streamflow. The hydraulic consequences of a climate change can cause natural disaster, such as drought that occurs when there is a significant deficit in precipitation. It will also have serious impact to flooding, water quality, and ecosystems that are closely related to the society of human beings. Accurate estimation of streamflow quantity from a watershed will provide important information to determine urban watershed modeling, water quantity management, development of legislation, and strategies on water supply. In addition, drought prediction is one of most complicated and difficult hydrological problems because the nature of the drought variables is random and unpredictable and the physical processes underneath the phenomenon are too complex. It is also because of the insufficient knowledge on the driving factors and their impact on streamflow, as well as the lack of reliable prediction and design methodologies. Therefore, accurate drought prediction including streamflow quantity prediction, precipitation and soil moisture prediction are all critical to enhance the water resource management plan and operational performance assessment.

Recently some deep learning algorithms have been successfully applied to the water quantity prediction and drought prediction problems. A deep Belief Network layered by coupled Restricted Boltzmann Machines was proposed for long-term drought prediction across the Gunnison River Basin [3]. By using time lagged standardized streamflow index (SSI) sequence, it demonstrated lower error rate than multilayer perceptron neural network and support vector regression. A long short-term memory (LSTM) network was presented for streamflow prediction using previous streamflow data for a particular period [4]. It showed that the LSTM model can not only predict the relativey steady streamflow in the dry season, but can also capture data characteristics in the rapidly changing streamflow in the rainy season. However, the performance of LSTM hasn't been proved on the effect of drought variables, such as precipitation, soil moisture, streamflow for long-term drought prediction.

The novelty of this paper is the inclusion of a wider ranged hydrological variables to predict soil moisture content (%) for a higher elevation of interest using existing regression models, which differentiates this work from previously done research works as described in the literature. In addition, this paper presents how to design the architecture of the model and layer specifications to the time series prediction problem. Further it customizes the LSTM based time series model to solve the drought prediction problem. It describes the proposed long short-term memory networks (LSTM) based deep learning method to predict the historical monthly soil moisture time series data.

The rest of this paper is organized as follows. Section 2 describes the methodology including deep learning approach, deep neural network, and convolutional neural network. In section 3, time series prediction using LSTM network is discussed. Performance evaluation metrices are presented in Section 4. In Section 5, the Modern-Era Retrospective analysis for Research and Applications (MERRA)-Land data set is described. The simulations and experimental results are demonstrated. In Section 6, the conclusions are given.

2 Methodology

Many state-of-the-art machine learning techniques, such as neural network, support vector machine, radial basis function, naive Bayes, decision tree, k-nearest neighbors, and deep learning have been applied to the time series prediction. However, few of them has been applied to the forecast of the probability of streamflows. These machine learning methods have been proven effective in predicting time series. Since streamflow prediction is a special case of time series prediction, therefore, they should be very promising in the streamflow prediction problems.

2.1 Deep Learning Approach

Deep learning algorithms are now applied to solve problems of a diverse nature, including prediction [5]. Therefore, we are considering deep learning algorithms for this research. Firstly, we would like to review a few basics of deep learning. The building blocks of deep learning or artificial neural networks are called perceptron, which mimics an equivalent functionality (in computation) as neuron (a biological cell of the nervous system that uniquely communicates with each other) [6].

Now, perceptron or artificial neurons receive input signals $(x_1, x_2, \ldots, x_m)$, multiply input by weight $(w_1, w_2, \ldots, w_m)$, add them together with a pre-determined bias, and pass through the activation function, $f(x)$. The signal goes to output as 0 or 1 based on the activation function threshold value. A perceptron with inputs, weights, summation and bias, activation function, and output all together forms a single layer perceptron. However, in common neural network diagrams, only input and output layers are shown. In a practical neural network, hidden layers are added between the input and output layers. The number of hidden layers is a hyperparameter and usually determined by evaluating the model performance. If the neural network has a single hidden layer, the model is called a shallow neural network, while a deep neural network consists of several hidden layers. In this research, we have considered DNN, convolutional neural network, and recurrent neural network in the form of long short-term memory, all of which will be discussed in the following sections.

2.2 Deep Neural Network (DNN)

DNN is composed of three neural network layers, namely an input layer, hidden layers, and an output layer. The number of hidden layers is tuned through trial and error [6]. Figure 1 illustrates such a model structure with two hidden layers consisting of three neurons each, five input neurons, and one output neuron. The number of neurons depends on the number of inputs and outputs. In Figure 1,

Inputs: $[x_1, x_2, x_3, x_4, x_5]$

Hidden layer weights: h

Output: $\hat{y}$

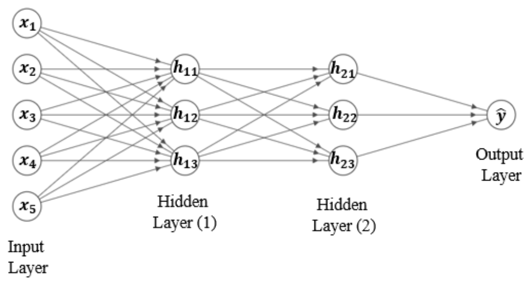


Figure 1 Simplified architecture of a deep neural network

$$f(x;W,c,w,b) = w^T \max(0, W^T + c) + b \quad (1)$$

$$h = g(W^T x + c) \quad (2)$$

$$f(x) = \max(0, x) \quad (3)$$

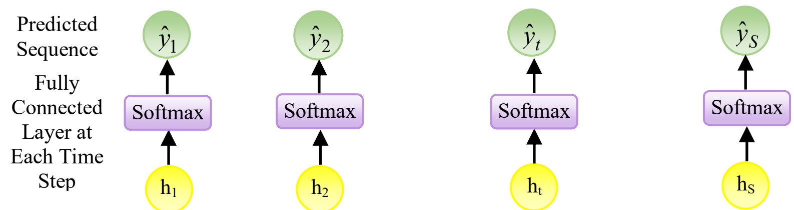
A simplified DNN kernel is formulated in (1) that considers linear modeling. *x, W,* and *c* symbolize input, weights, and bias, respectively, while *w* and *b* are linear model parameters. The hidden layer parameter *h* is shown in (2), where *g* is the activation function. For DNN modeling, ReLu (3) is used as the hidden layer activation function.

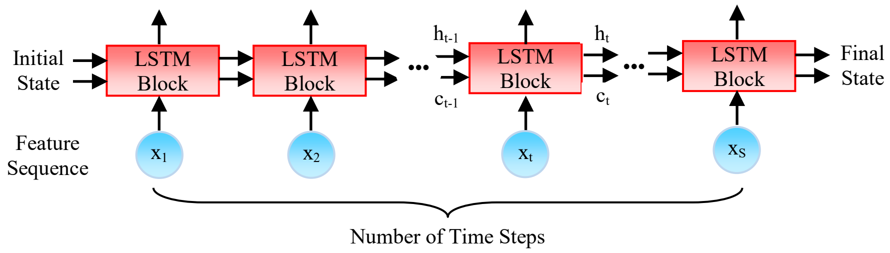
3 Proposed Method

This section describes the proposed long short-term memory networks (LSTM) based deep learning method to predict the historical monthly soil moisture time series data. It presents how to design the architecture of the model and layer specifications to the time series prediction problem. Further it customizes the LSTM based time series model to solve the drought prediction problem.

3.1 Time Series Prediction Using LSTM Network

An LSTM network inherits the characteristic of memory from the recurrent neural network (RNN) [7]. This memory unit enables long-term feature retention between time steps of sequence data [8]. Figure 2 illustrates the flowchart of a time series *X* with *C* features of length *S* through an LSTM layer. The output layer will generate the predicted values, which contains D features of length S. In the diagram, for the *t*th LSTM block, h_t and c_t denote the output, i.e. the hidden state and the cell state at time step *t*, respectively.





$$x_1 = \begin{pmatrix} x_{11} \\ x_{21} \\ \vdots \\ x_{C1} \end{pmatrix} \quad x_2 = \begin{pmatrix} x_{12} \\ x_{22} \\ \vdots \\ x_{C2} \end{pmatrix} \quad \cdots \quad x_t = \begin{pmatrix} x_{1t} \\ x_{2t} \\ \vdots \\ x_{Ct} \end{pmatrix} \quad x_S = \begin{pmatrix} x_{1S} \\ x_{2S} \\ \vdots \\ x_{CS} \end{pmatrix}$$

$$\hat{y}_1 = \begin{pmatrix} y_{11} \\ y_{21} \\ \vdots \\ y_{D1} \end{pmatrix} \quad \hat{y}_2 = \begin{pmatrix} y_{12} \\ y_{22} \\ \vdots \\ y_{D2} \end{pmatrix} \quad \cdots \quad \hat{y}_t = \begin{pmatrix} y_{1t} \\ y_{2t} \\ \vdots \\ y_{Dt} \end{pmatrix} \quad \hat{y}_S = \begin{pmatrix} y_{1S} \\ y_{2S} \\ \vdots \\ y_{DS} \end{pmatrix}$$

Figure 2 Unfolded single layer of LSTM network

Initially, the states of all the LSTM blocks will be initialized to all zeros. The first LSTM block to the left most uses the initial state of the network and the first time step of the sequence to compute the first output, h_1 and the updated cell state, c_1. At time step t, the tth LSTM block uses the current state of the network (h_{t-1} , c_{t-1}) and the tth time step of the sequence to compute the output state, h_t , and the cell state, c_t .

An LSTM layer contains an array of LSTM blocks. For each LSTM block, it is represented by two states, including an output state, i.e. the hidden state and a cell state. The hidden state at time step t not only contains the output of the current LSTM block for the time step, but also serves as the input for the LSTM block at the next time step. The cell state contains time dependent information extracted from the previous time steps.

Different from the classic RNN, the LSTM is a recurrent neural network equipped with gates [9]. At each time step, the LSTM layer can choose either add information to or removes information from the cell state. The layer controls these updates using gates. The gated circuit of the LSTM is proposed to implement the flow of data at time step t, as illustrated in Figure 3. LSTM introduces self-loops to produce paths where the gradient can flow for a long duration; thus, it is capable of learning long-term dependencies [6].

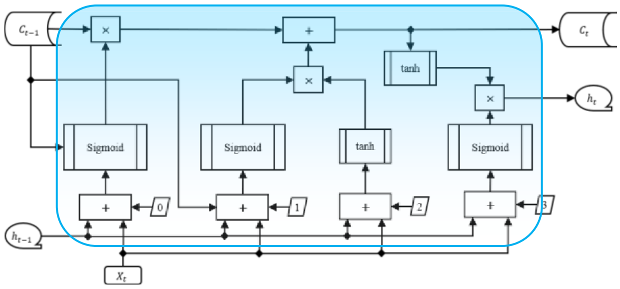


Figure 3 Block diagram of LSTM operations on a time series sequence

The equations describing the operations are listed below.

$$f(t) = \sigma_g (W_f x_t + U_f h_{t-1} + b_f) \tag{4}$$

$$i_t = \sigma_g (W_i x_t + U_i h_{t-1} + b_i) \tag{5}$$

$$o_t = \sigma_g (W_o x_t + U_o h_{t-1} + b_o) \tag{6}$$

$$c_t = f_t \circ c_{t-1} + i_t \circ \sigma_c (W_c x_t + U_c h_{t-1} + b_c) \tag{7}$$

$$h_t = o_t \circ \sigma_h (c_t) \tag{8}$$

where,

$x_t \in \Re^d$: Input vector to the LSTM unit

$f_t \in \Re^h$: Forget states activation vector

$i_t \in \Re^h$: Input/update gate's activation vector

$o_t \in \Re^h$: Output gate's activation vector

$h_t \in \Re^h$: Hidden state vector

$c_t \in \Re^h$: Cell state vector

$W \in \Re^{h \times d}, U \in \Re^{h \times h}, b \in \Re^h$: Weight matrices and bias vector parameters which will be adjusted during the training

σ_g : Sigmoid function

σ_c, σ_g : hyperbolic tangent function

In the performance evaluation, some commonly used accuracy parameters, such as root mean square error are employed to evaluate how well a model is performing to predict the intended parameter. Root mean square error (RMSE) is considered to investigate the model performances on the test set by comparing the differences between the predicted values by a model and the actual values. RMSE is the square root of the mean of the square of error terms (the difference between actual response (y_i) and predicted response ($\hat{y}_i$). n is the number of total input sets. The lower this value is, the better the model performance, while the desired is 0 or close value for this term. The formula for this measure is in (9).

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n}} \tag{9}$$

4 Experimental Results

The Modern-Era Retrospective analysis for Research and Applications (MERRA) data set is used to use the historical soil moisture (total profile soil moisture content) from 1980 to 2012 to predict the future soil moisture [9]. The data set is plotted in Figure 4. We train on the first 90% of the time series sequence and test on the last 10%. In order to obtain the identical data scale for different features, it is necessary to pre-process the raw data by standardizing the data to a normalized distribution. Within the scope of zero mean and unit variance, we prevent the training data, test data, and predicted responses from diverging.

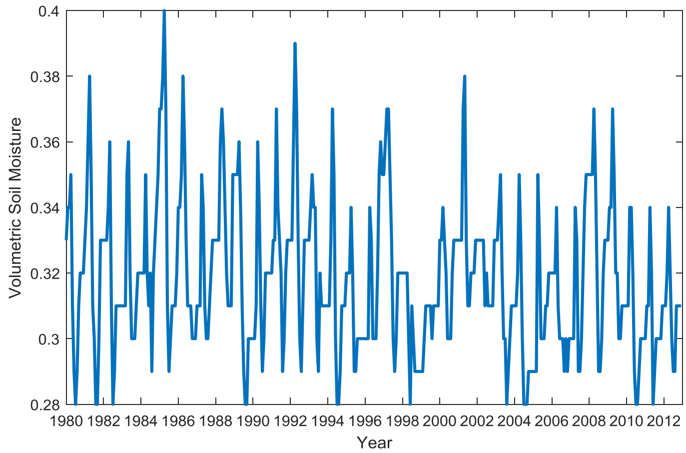


Figure 4 Monthly soil moisture (total profile soil moisture content) from MERRA-Land from 1980 to 2012

To forecast the values of a sequence at future time steps, we use the responses with values lagged by one time step to be the training sequences. The nonlinear autoregressive (NAR) model can be represented mathematically by predicting the values of a sequence at future time steps, $\hat{y}$ from the historical values of that time series, as shown in Figure 5. Time series without the final time step are used as the training sequences. The form of the prediction can be expressed as follows:

$$\hat{y}(t) = f(y(t-1), \ldots, y(t-d)) \tag{10}$$

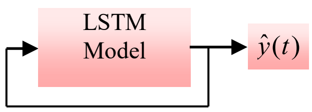


Figure 5 Nonlinear autoregressive LSTM prediction model

We set up a LSTM network in the sequence-to-one regression mode. The output of the LSTM layer is the last element of the sequence and will be fed into the fully connected layer. For example, if the input sequence is $\{x_1, x_2, x_3, x_4\}$, the output of the LSTM layer will be the hidden state, h_4. In this LSTM network, it consists of a sequence input layer, an LSTM layer, a fully connected layer, and a regression output layer. In the LSTM layer array, a sequence input layer inputs one sequence data to a network at a time. This LSTM layer contains 200 hidden units. We use the adam optimization algorithm featured with adjustable learning rate to train the dynamic neural networks for 600 epochs. To ensure a steady gradient change, we limit the threshold of the gradient to 1.

After several trials, we decide to set the initial learning rate to 0.005 to gain better performance. We slow down the learning rate to 20% of its original value when it has elapsed 150 epochs. Then we train the LSTM network with these parameter selections. The training progress is plotted in Figure 6. The top subplot reveals the root-mean-square error (RMSE) calculated from the standardized data. The bottom subplot displays the the error between the actual values and the predicted values.

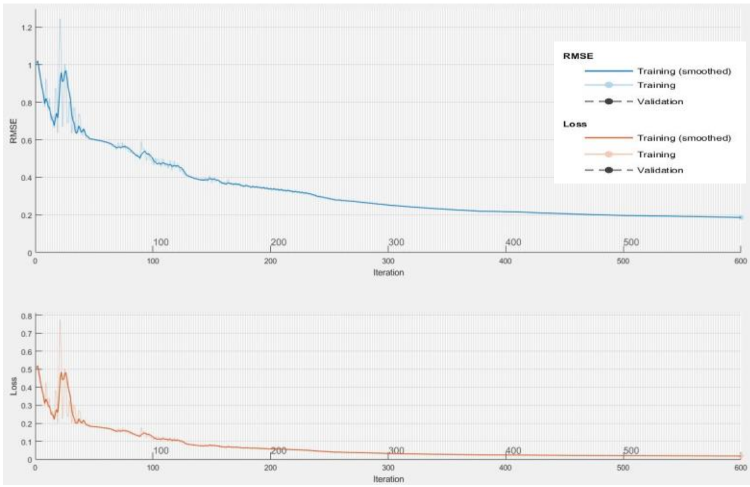


Figure 6 Training progress on the monthly soil moisture using LSTM network

Once the LSTM network has been trained, we will predict time steps one at a time and update the network state at each prediction. Therefore, we can forecast the values of multiple time steps in the future. Like what we did for the training data, we standardize the test data using the same mean of the population, μ and the standard deviation of the population, σ. In order to initialize the network state, h, we first predict on the training data. Then *we use the value at the last time step of the training response to make the very first prediction*. We then use Eq. (10) to *use the previous prediction* to predict value at the next time step one at a time for the remaining predictions. We un-standardize the predictions in order to observe the real world values of the soil moisture. The combination of training time series (in blue) with the forecasted values (in red) is shown in Figure 7.

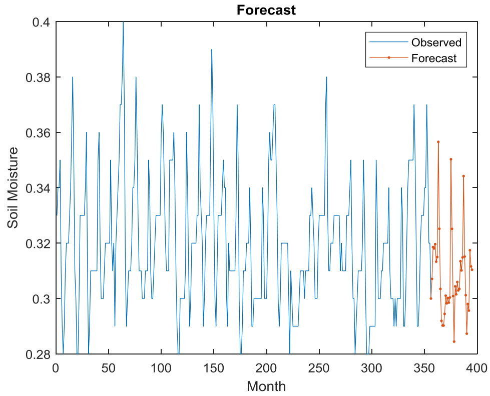


Figure 7 Training time series with the forecasted values

In order to visually compare the forecasted values with the actual data, we plot the first 40 predicted values at the time steps over the actual values, as shown in Figure 8. We also display the difference between them at each time step and the RMSE, i.e. 0.019717 from the unstandardized predictions in the lower subplot in Figure 8.

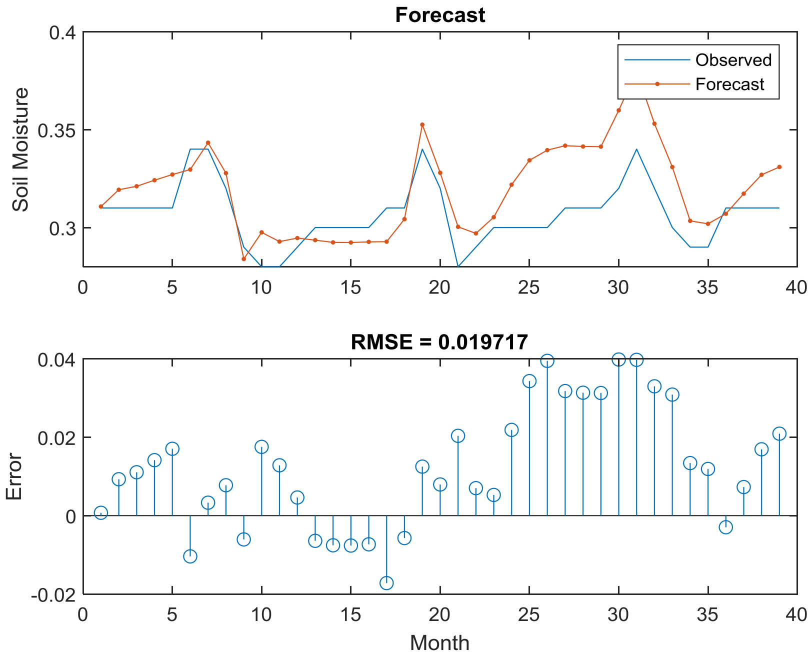


Figure 8 Comparison of predicted monthly soil moisture with the test data when updating the network state with previous predictions

We further explore the prediction performance by updating network state with observed values. Unlike the previous study where we used the previous prediction to predict value, we update the network state with the actual (observed) values instead of the predicted values. We first initialize the network state by *resetting the network state to an initial state of zeros*. Therefore, previous predictions will not affect the predictions on the new time sequence. We then initialize the network state by start predicting on the training data. At each time step, *we predict the value on the next time step using the observed value of the previous time step*. In order to retrieve the soil moisture information, we un-standardize the predictions using the same mean and the standard deviation of the population as before.

Similarly, we compare the forecasted values with the actual test data for the first 40 time steps, as shown in Figure 9. We also demonstrate the difference between them at each time step, as well as the RMSE, i.e. 0.0087584 from the unstandardized predictions in the bottom subplot. After comparing Figure 8 and Figure 9, we find that the prediction accuracy is much higher when we update the network state with the observed values instead of using the predicted values.

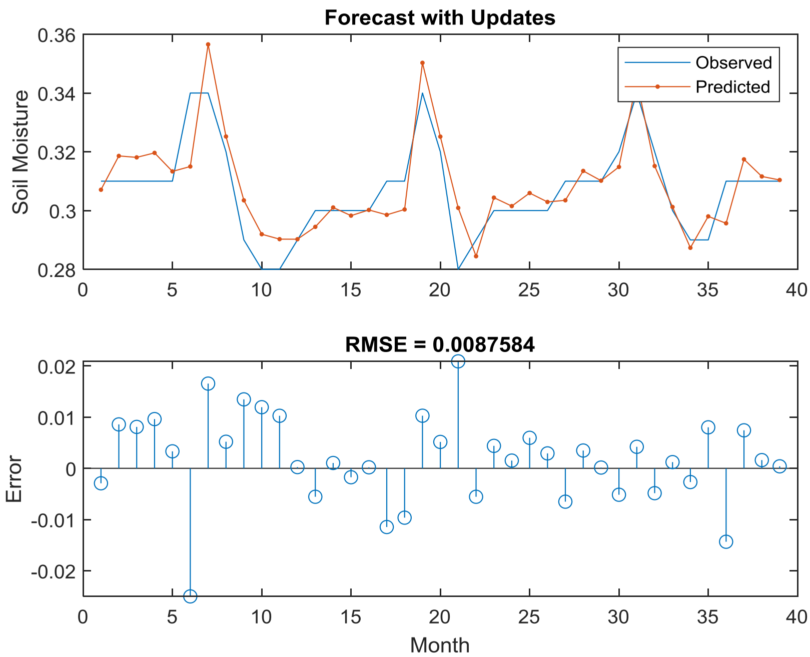


Figure 9 Comparison of predicted monthly soil moisture with the test data when updating the network state with the observed values

We also compare the performance of the proposed LSTM deep learning model with other popular predictive models, such as autoregressive integrated moving average model (ARIMA) and autoregressive model (AR). Table 1 depicts the root mean squared error (RMSE) for each algorithm on the test data. We found that our proposed algorithm has the lowest error rate.

Table 1 Comparative Model Performances

Algorithm	Root Mean Squared Error (MSE)
Autoregressive Integrated Moving Average model (ARIMA)	0.0950
Autoregressive model (AR)	0.0246
Proposed LSTM model	**0.0088**

5 Conclusions

This paper proposes a long short-term memory networks (LSTM) based deep learning method to predict the historical monthly soil moisture time series data based on the MERRA-Land from 1980 to 2012. The proposed LSTM model learns to predict the value of the next time step at each time step of the time sequence. We customize the dynamic LSTM model to solve the soil moisture prediction problem. We also compare the predication accuracy when the network state is updated with the observed values and when the network state is updated with the predicted values. We find that the predictions are more accurate when updating the network state with the observed values instead of the predicted values. Furthermore, we also compare the proposed method with other time series prediction methods. We find that it has much lower MSE than the autoregressive integrated moving average model (ARIMA) model and autoregressive model (AR) model. The future study will to obtain the soil moisture index, and use it to predict the Drought index. The drought prediction system will have profound impact to the water resources management, agriculture, and urban construction.

Acknowledgment

This work was supported in part by the National Science Foundation (NSF) grants #1505509 and #2011927, DoD grants #W911NF1810475 and #W911NF2010274, NIH grant #1R25AG067896-01, and USGS grant #2020DC142B, and in part by the Foundation of Chongqing Municipal Key Laboratory of Institutions of Higher Education ([2017]3), Foundation of Chongqing Development and Reform Commission (2017[1007]), Scientific and Technological Research Program of Chongqing Municipal Education Commission (Grant Nos. KJQN201901218 and KJQN201901203), Natural Science Foundation of Chongqing (Grant No. cstc2019jcyj-bshX0101), Foundation of Chongqing Three Gorges University.

References

1. Water Supply Outlook and Status, Interstate Commission on the Potomac River Basin, https://www.potomacriver.org.
2. Hao, Z., AghaKouchak, A., Nakhjiri, N., Farahmand, A.: Global Integrated Drought Monitoring and Prediction System. Scientific Data, Vol. 1:140001 (2014) 1-10

3. Agana, N. A., A. Homaifar: A Deep Learning Based Approach for Long-Term Drought Prediction. In: SoutheastCon 2017, Charlotte, USA, March 30-April 2 (2017) 1-8
4. Fu, M., Fan, T., Ding, Z., Salih, S. Q., Al-Ansari, N., Yaseen, Z. M.: Deep Learning Data-Intelligence Model Based on Adjusted Forecasting Window Scale: Application in Daily Streamflow Simulation. IEEE Access, Vol. 8 (2020) 32632-32651
5. Filik, Ü. B. and Filik T.: Wind Speed Prediction Using Artificial Neural Networks Based on Multiple Local Measurements in Eskisehir. Energy Procedia, Vol. 107 (2017) 264-269
6. Goodfellow, I.: Deep Learning. MIT Press (2016)
7. Rochac, J. F. R., Zhang, N., Xiong, J: A Spectral Feature Based CNN Long Short-Term Memory Approach for Classification. In: The Tenth International Conference on Intelligent Control and Information Processing (ICICIP), Marrakesh, Morocco, December 14-19, (2019)
8. Ehsan, M. A., Shahirinia, A., Zhang, N., Oladunni, T.: Interpretation of Deep Learning on Wind Speed Prediction and Visualization. In: The 10th International Conference on Information Science and Technology (ICIST), Bath, London, and Plymouth, United Kingdom, September 9-15 (2020)
9. Rochac, J. F. R., Zhang, N., Xiong, J, Zhong, J., Oladunni, T.: Data Augmentation for Mixed Spectral Signatures Coupled with Convolutional Neural Networks. In: The 9th International Conference on Information Science and Technology (ICIST 2019), Hulunbuir, Inner Mongolia, China, August 2-5 (2019)
10. Koster, R.D., De Lannoy G. J. M., Forman, B.A., Liu, Q., Mahanama, S. P. P., Touré, A.: Assessment and Enhancement of MERRA Land Surface Hydrology Estimates. J. Climate, Vol. 24 (2011) 6322–6338