Streamflow Simulation in Data-Scarce Basins Using Bayesian and Physics-Informed Machine Learning Models

DAN LU,a GOUTAM KONAPALA,b SCOTT L. PAINTER,b SHIH-CHIEH KAO,b AND SUDERSHAN GANGRADEb

a Computational Sciences and Engineering Division, Oak Ridge National Laboratory, Oak Ridge, Tennessee
b Environmental Sciences Division, Oak Ridge National Laboratory, Oak Ridge, Tennessee

(Manuscript received 2 April 2020, in final form 5 February 2021)

ABSTRACT: Hydrologic predictions at rural watersheds are important but also challenging due to data shortage. Long short-term memory (LSTM) networks are a promising machine learning approach and have demonstrated good performance in streamflow predictions. However, due to its data-hungry nature, most LSTM applications focus on well-monitored catchments with abundant and high-quality observations. In this work, we investigate predictive capabilities of LSTM in poorly monitored watersheds with short observation records. To address three main challenges of LSTM applications in data-scarce locations, i.e., overfitting, uncertainty quantification (UQ), and out-of-distribution prediction, we evaluate different regularization techniques to prevent overfitting, apply a Bayesian LSTM for UQ, and introduce a physics-informed hybrid LSTM to enhance out-of-distribution prediction. Through case studies in two diverse sets of catchments with and without snow influence, we demonstrate that 1) when hydrologic variability in the prediction period is similar to the calibration period, LSTM models can reasonably predict daily streamflow with Nash–Sutcliffe efficiency above 0.8, even with only 2 years of calibration data; 2) when the hydrologic variability in the prediction and calibration periods is dramatically different, LSTM alone does not predict well, but the hybrid model can improve the out-of-distribution prediction with acceptable generalization accuracy; 3) L2 norm penalty and dropout can mitigate overfitting, and Bayesian and hybrid LSTM have no overfitting; and 4) Bayesian LSTM provides useful uncertainty information to improve prediction understanding and credibility. These insights have vital implications for streamflow simulation in watersheds where data quality and availability are a critical issue.

KEYWORDS: Hydrologic models; Bayesian methods; Machine learning; Neural networks

1. Introduction

Streamflow predictions provide essential information to support short-term flood risk mitigation and long-term water resources system management to advance agricultural and economic development (Hirpa et al. 2016; Pagano et al. 2014). However, accurate prediction of streamflow is very challenging because rainfall–runoff is a complex dynamic process with high spatiotemporal variability and influenced by many factors such as topographic and climatic characteristics (Beven 1995; Werner and Yeager 2013). Description of these processes and their variations across multiple spatial and temporal scales requires a large number of data (Gupta et al. 2014), including site-specific watershed attributes and dynamic meteorological time series, which makes the streamflow simulation in data-scarce regions particularly challenging (Alipour and Kibler 2018). This study aims to improve streamflow simulation in poorly monitored rural catchments with short records of observations and in absence of detailed and reliable catchment attributes.

Hydrologic models allow simulating and forecasting of streamflow based on meteorological observations (Beven 2001). The classical physics-based hydrologic approach is to first develop conceptual models having fixed structures and parameterizations that reflect our physical understanding of internal catchment structure and functioning such as rainfall–runoff processes and their interactions, and then to apply these prespecified structures to different catchments by adjusting the values of the model parameters (Gupta and Nearing 2014). When the key processes are captured and the models are well calibrated, these physics-based hydrologic models can provide reasonable streamflow simulations even outside the range of historical observations and in the ungauged catchments. However, due to the complexity of hydrologic processes and the limited data, the physics-based models may have incomplete process representation and poor model calibration resulting in biased streamflow simulations in the data-scarce catchments (Clark et al. 2017).

Recently, machine learning approaches, long short-term memory (LSTM) networks in particular, have been applied for hydrologic simulations (Boyraz and Engin 2018; Kratzert et al. 2018, 2019b; Zhu et al. 2020; Fang and Shen 2020). LSTM is a purely data-driven model that enables learning, from time series input and output data, the system patterns associated with the observed dynamical system behaviors. In contrast to physics-based models, the LSTM does not know the principle of mass conservation and the governing process equations describing the rainfall–runoff processes a priori. It must both learn the physical laws and the appropriate model parameters during the model calibration/training purely from the observations. Thus, LSTM is data hungry and its simulation...
performance strongly depends on the data availability. At present, LSTM is mainly used under a big data paradigm. For example, Le et al. (2019) applied LSTM for flood forecasting based on 24 years of daily data with 18 years of data for training, 5 years for validation, and the remaining 2 years for testing. Tian et al. (2018) trained the LSTM models with 10 years of daily data and validated streamflow simulation at two river basins using another 5 years. Kratzert et al. (2019a) considered 30 years of and validated streamflow simulation at two river basins using an additional loop to the classical NN architecture, is

In this effort, we aim to 1) investigate the predictive capability of LSTM in data-scarce basins, 2) evaluate different regularization techniques to address the overfitting challenge, 3) introduce a Bayesian LSTM for UQ so as to provide accurate and credible predictions in small-data learning, and 4) introduce a physics-informed hybrid LSTM to improve the out-of-distribution prediction. We apply the methods to the East River Watershed, Colorado, which includes several headwater catchments that have only 2–3 years of daily streamflow observations. We analyze the potential of LSTM for learning rainfall–runoff dynamics using as few as 1 year of training data. We exploit different strategies, such as regularization, Bayesian inference, and hybrid modeling, to improve the streamflow simulation in the small-data catchment according to its specific hydrologic response to the meteorological and streamflow dynamics. Additionally, we consider four long-record gauge stations which provide more data to further verify our findings; one gauge station is located at the East River watershed outlet, and the other three stations are in the southeastern United States where the snow has little influence on runoff. The LSTM performance is evaluated in comparison with a physics-based hydrologic model calibrated using the same training data as in the LSTM. Results from this study should have important implications for streamflow simulation in rural watersheds where data quality and availability are a critical issue.

The paper is organized as follows. Section 2 describes the LSTM network, various regularization techniques, Bayesian LSTM and the physics-informed hybrid LSTM. Section 3 depicts the application of these methods to data-scarce basins including the introduction of the study area, data, physics-based hydrologic model, and streamflow simulation problems. Section 4 presents modeling results and discussion of the effectiveness of the methods. Last, section 5 provides concluding remarks and future research directions.

2. Methods

In this section, we first describe the LSTM network briefly, then we discuss some regularization techniques for LSTM training. We then introduce a special Bayesian LSTM model for mitigating overfitting and quantifying uncertainty, and last we present the hybrid model that combines a physics-based simulation and the data-driven LSTM network.

a. Long short-term memory network

LSTM is a special type of recurrent neural network (RNN) that is structured to tackle the limitations of the standard RNN to learn long-term dependence in time series prediction, making it particularly suitable for daily streamflow simulation where lag times between precipitation (including both rainfall and snow) and discharge can be up to years.

In daily streamflow modeling, we usually use previous \( t \) days of meteorological observations as inputs to predict discharge at the next day. However, given the complexity of rainfall–runoff processes and the “memory” of catchments, the temporal dependence of discharge on the meteorological inputs is unknown and should be learned dynamically. RNN, by creating an additional loop to the classical NN architecture, is
specifically designed to learn the dynamic temporal dependence structure within the data. As shown in Fig. 1, when we unroll the loop of RNN to a chain of repeating modules of simple NNs, we can see that each NN reads the input sequence \( x_t \) (i.e., meteorological data in this study) one time step at a time and the NN output from the previous time step is fed into the “next” NN as another input along with the input at current time step to affect the prediction, and so on. The outputs from the NN chain are saved in the hidden states \( h_t \), which dynamically add and store memory from the input sequence. Finally, RNN maps information saved in \( h_t \) to the quantity of prediction \( y \) (i.e., discharge in this study) using a dense layer. Mathematically,}

\[
\begin{align*}
\text{(1)} & \quad h_t = \sigma(Wx_t + Uh_{t-1} + b), \\
\text{(2)} & \quad y = W_yh_t + h_d,
\end{align*}
\]

where \( \sigma(\cdot) \) represents the sigmoid function; \( W, U, \) and \( W_d \) are the adjustable weight matrices connecting to the input sequence \( x_t \), to the hidden state \( h_t \), and of the dense layer, respectively; and \( b \) and \( b_d \) are two adjustable bias vectors. In the first time step \( t = 1 \), the hidden state \( h_0 \) is initialized as a vector of zeros with a user-defined length. Here we use a single-layer RNN as an example in Fig. 1 and Eq. (1). Multiple layers can be stacked on top of each other straightforwardly to form a stacked RNN. When a single RNN layer with \( n \) nodes is used to learn information from \( m \) input sequences, \((mn + n^2 + n) + (n + 1)\) parameters need to be calibrated from the training data where \( mn + n^2 + n \) is from the recurrent layer [Eq. (1)] and \( n + 1 \) from the dense layer [Eq. (2)].

RNN learns a mapping for the inputs over time to an output (Fig. 1). Thus, RNN knows what observations it has seen previously are relevant and how they are relevant for prediction, which enables a dynamical learning of temporal dependence. In principle, RNN is able to learn long-term information, but practically the range of data that a standard RNN can access is very limited. The problem is that the impact of inputs on the output decays exponentially as RNN cycles around the recurrent connections, because the definition of \( h_t \) in Eq. (1) involves a cascading multiplication of gradients whose values are less than one.

LSTM, with specially designed cells, can avoid the information-decay problem and is able to learn long-term temporal dependence. In comparison to the standard RNN cell that has a simple structure such as a single sigmoid function in Eq. (1), the LSTM cell, as shown in Fig. 2, uses four functions and three of them are acting as regulatory gates to control the information flow. Furthermore, LSTM introduces an additional cell state \( c_t \) to add and store information and then transfer the stored information to the hidden state \( h_t \). Specifically, LSTM first uses a sigmoid function \( f \) that acts as a forget gate to decide what information should be thrown away from the old cell state, i.e.,

\[
\text{(3)} & \quad f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f).
\]

Next, LSTM uses another two functions to decide what new information should be stored in the cell state. A hyperbolic tangent (tanh) function \( g \) first creates a vector of new candidate values that could be added to the cell state, and a sigmoid function \( i \) that performs as an input gate then decides which candidate values need to be updated. The two input functions are then combined with the forget gate to update the old cell state \( c_{t-1} \) to a new cell state \( c_t \). That is,

\[
\text{(4)} & \quad g_t = \tanh(W_g x_t + U_g h_{t-1} + b_g), \\
\text{(5)} & \quad i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i), \\
\text{(6)} & \quad c_t = f_t \times c_{t-1} + i_t \times g_t.
\]

Finally, LSTM uses the third sigmoid function \( o \) that acts as an output gate to decide what parts of the cell state should be exported to update the hidden state and updates \( h_t \) from a tanh-transformed cell state, i.e.,

\[
\text{(7)} & \quad o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o), \\
\text{(8)} & \quad h_t = o_t \times \tanh(c_t).
\]
This definition of cells enables LSTM to learn long-term dependence. First, the introduction of cell states avoids information loss that resulted from the recurrent calculation in Eq. (1) by defining $c_t$ as a linear function [Eq. (6)]. Additionally, the use of three regulatory gates effectively allows LSTM to preserve memory over long time steps. Whereas, on the other hand, the complex cell structure in LSTM greatly increases the number of adjustable parameters, resulting in overfitting issues especially for a small training dataset. To be exact, LSTM cell involves four function evaluations, and each has $mn + n^2 + n$ parameters as Eq. (1), which leads to $4(mn + n^2 + n)$ parameters for one hidden layer. Plus $n + 1$ parameters from the dense layer, a single-layer LSTM needs to calibrate $4mn + 4n^2 + 5n + 1$ parameters in total. And a stacked LSTM would greatly increase the parameter size further. This significantly amplifies the challenges in LSTM training for a small dataset and some regularization techniques are required.

b. LSTM training and regularization techniques

Several regularization methods have been proposed to address overfitting in LSTM, such as reducing the node size, L2 norm penalty, dataset augmentation, and dropout. Reducing the node size suggests using a fewer number of nodes in the LSTM to reduce the parameter size, but the resulted LSTM may not be deep or wide enough to learn the underlying complex input–output relationship, causing poor performance in both training and prediction. On the other hand, the technique of L2 norm penalty limits LSTM capacity by adding a parameter L2 norm to the loss function and controls its relative contribution using a hyperparameter. This strategy drives the parameters closer to the origin, so a strong penalty could result in a slow learning and possibly a poor training, whereas a weak penalty might not solve the overfitting problem. Practically, the determination of the hyperparameter to control the penalty is problem dependent and chosen through trial and error.

Data augmentation solves overfitting from the data perspective by training the LSTM model on a large amount of data. This regularization technique seems infeasible in application to data-scarce catchments. A possible way is to increase the diversity of the data, which could be the observations specific to the basin or the simulation data from the physics-based model. The first form of data, if available, will be well leveraged in creation of a site-specific model and the second will be considered in our hybrid model as discussed in detail in the following section 2d.

Dropout, which randomly drops network nodes along with all its coming and outgoing connections, is a regularization technique specifically for NNs. Like norm penalty, it is computationally fast and easy to use, and the degree of regularization is controlled by a hyperparameter called dropout rate. Despite its successful applications in feedforward NNs, the use of dropout in LSTM has been somewhat limited. Bayer et al. (2013) stated that applying dropout in the recurrent connection of LSTM may corrupt long-term memory learning and the dropout should be applied to the input-to-hidden and hidden-to-output connections solely, whereas Gal and Ghahramani (2016) demonstrated that when only applying the dropout to the coming and outgoing connections, the LSTM model still suffered from overfitting.

c. Bayesian LSTM network

Bayesian LSTM replaces the weight parameters of deterministic networks with probability distribution over these parameters and, instead of optimizing the network weights to
find a single set of values that best fit the training data, it considers all possible weights that are likely to have generated the data. Thus, Bayesian LSTM naturally addresses the overfitting issue, and more importantly, it quantifies predictive uncertainty using distributions (Lu et al. 2019).

However, solving Bayesian LSTM problems is very challenging. Traditional Bayesian inference methods, such as Markov chain Monte Carlo sampling and Hamiltonian approaches, are computationally unaffordable due to the curse of dimensionality. Gal and Ghahramani (2016) showed that dropout can be interpreted as a variational approximation to the posterior distribution of a Bayesian LSTM network by using a mixture of two Gaussians as the variational distribution. Under this definition, the Bayesian LSTM can be solved by training the network with dropout and then taking Monte Carlo (MC) samples of the prediction by using dropout in forward simulations to quantify the uncertainty. Such formulation of the Bayesian LSTM, also called MC-dropout, greatly reduces the computational complexity, making its calculation practically feasible (Gal and Ghahramani 2015).

Although it is defined in the dropout context, the MC-dropout is fundamentally different from the standard usage of dropout in LSTM. In standard dropout LSTM, the masks are only applied to input and output feed-forward connections and not to the recurrent connections; besides, different network nodes are dropped at different time steps. In contrast, the MC-dropout repeats the same mask (i.e., drops the same network units) at each time step for both inputs, outputs, and recurrent connections. Figure 2 shows the dropped connections in the recurrent layer using dashed lines. Mathematically, the four functions in the Bayesian LSTM cells can be calculated as follows:

\[
\begin{align*}
\mathbf{f} &= \text{sigmoid}(\mathbf{W}_f \mathbf{x}_t + \mathbf{U}_f d(h_{t-1}) + \mathbf{b}_f) \\
\mathbf{g} &= \tanh(\mathbf{W}_g \mathbf{x}_t + \mathbf{U}_g d(h_{t-1}) + \mathbf{b}_g) \\
\mathbf{i} &= \text{sigmoid}(\mathbf{W}_i \mathbf{x}_t + \mathbf{U}_i d(h_{t-1}) + \mathbf{b}_i) \\
\mathbf{o} &= \text{sigmoid}(\mathbf{W}_o \mathbf{x}_t + \mathbf{U}_o d(h_{t-1}) + \mathbf{b}_o)
\end{align*}
\]

where \(d\) is the dropout function defined as

\[
d(x) = \begin{cases} 
\text{mask} \times x, & \text{if in training phase} \\
(1-p)x, & \text{otherwise}
\end{cases}
\]

and where \(p\) is the dropout rate and mask is a vector sampled from the Bernoulli distribution with success probability \(1-p\). In this way, evaluating the prediction with parameter samples corresponds to randomly masking rows in each weight matrix during the forward pass (i.e., performing dropout). And each weight matrix is randomly masked once and the same mask is used throughout all time steps. In practice, we can use the sample mean of the prediction as the estimate, and the sample variance to quantify the prediction uncertainty.

Besides the capability of UQ, Eq. (9) indicates that the Bayesian LSTM also enables regularization. As we can see, this algorithm uses dropout to regulate gate values and cell updates, which allows for adding a regularizer on the model weights responsible for learning short- and long-term dependencies without affecting the ability to capture long-term relationships. Thus, this Bayesian LSTM model is especially useful for the daily streamflow simulation in the data-scarce watershed where overfitting can be an issue and where the long lag time between precipitation and discharge needs to be simulated with UQ.

### d. Physics-informed hybrid LSTM model

The performance of a data-driven LSTM model is mainly controlled by the quantity and quality of the training data. On the other hand, the performance of a physics-based hydrologic model is controlled by the reasonableness of the model structure and its parameterization. With limited historic observations, the purely data-driven LSTM cannot generalize well outside the training range, while the physics-based model cannot be well calibrated for a good simulation. It is expected that by combining the power of the data-driven and the physics-based models, we can overcome their complementary deficiencies and leverage information in both data and physics so as to enhance the simulation performance.

Generally speaking, hydrologic modeling information presents in two forms, as hydrologic principles/mathematical equations and in the form of numerical simulations of the hydrologic system. By combining the different forms of hydrologic information with the LSTM network in different ways, we can create a variety of hybrid models, such as implementing the hydrologic principles in the loss function of LSTM training and using the LSTM to learn the residuals from the physics-based model simulation. In this work, we consider a direct way for developing the hybrid model, i.e., using the physics-based hydrological model simulation outputs as another set of inputs in the LSTM network along with the meteorological data (Konapala et al. 2020; Karpatne et al. 2017). This setup still uses the LSTM network for training but it is fed with extra data/information from the physics-based model. Therefore, the hybrid model not only leverages the strong power of LSTM in information extraction and relationship learning, but also satisfies its data-hungry needs by providing process-supported simulation. The use of simulation results as inputs can accelerate the training convergence. The caveat is that, if the physics-based model has a systematic bias in simulating the streamflow, it is likely that LSTM might pick up the bias in some degrees, although we do not expect a strong destruction on the performance because LSTM also learns from other input sequences which enables to complement the structural deficiency.

### 3. Application

In this section, we apply standard LSTM, Bayesian LSTM, and the physics-informed hybrid model for daily streamflow simulation to the snowmelt dominated East River (ER) Watershed and the Alabama–Coosa–Tallapoosa (ACT) River basin where snow has little influence on runoff. We first introduce the study area, the available data, and the physics-based hydrologic model, then we describe the numerical experiments setup and model evaluation criteria.

#### a. Study areas and data

The ER Watershed in Colorado (Hubbard et al. 2018) contains several headwater catchments in the Upper Colorado
River basin (Fig. 3a). The watershed has 1420 m of topographic relief and significant variability in hydrology, vegetation, geology, and weather. The average elevation is 3266 m. Annual average precipitation is approximately 2000 mm yr$^{-1}$ and is mostly snow. River discharge is driven by snowmelt in late spring and early summer and by monsoonal-pattern rainfall in summer. We consider data from four gauge stations in ER watershed, two from the headwater catchments (Quigley and Rock Creek), one from a low-gradient meandering reach that aggregates several headwater catchments (Pumphouse), and a further downstream U.S. Geological Survey (USGS) gauge station 09112500 near the watershed outlet at Almont, Colorado. To support a large multi-institutional research project in the upper ER watershed (i.e., above the Pumphouse location), there is a strong interest to collect and/or synthesize best available streamflow records to support further biogeochemical modeling efforts. The Quigley catchment has about 2 years of streamflow records from 1 September 2014 to 13 October 2016, a total of 774 daily observations. Pumphouse has 3 years of data from 1 October 2014 to 30 September 2017, while Rock Creek has data from 31 August 2014 to 4 October 2017, a total of 1131 daily observations. Gauge station 09112500 has over 86 years of continuous daily streamflow records from 1 October 1934 till present. Besides the observed streamflow, we have catchment aggregated meteorological forcing data such as precipitation, daily maximum and minimum temperatures at the same daily time step with streamflow. The catchment aggregated meteorological data are calculated from the 1-km resolution Daymet dataset from 1981 to 2018 (Thornton et al. 2014).

ACT River basin covers the northeastern and east-central parts of Alabama, northeastern Georgia, and small parts of Tennessee (Fig. 3b). The basin receives an annual average of 1379 mm of precipitation primarily from rainfall with minimal influence of snow on runoff. Forest is the major land cover type in the ACT River basin, which results in high evapotranspiration ranging from 762 to 1067 mm (56%–78% of annual precipitation), generally increasing from north to south (Gangrade et al. 2020). We consider three gauge stations in ACT basin, USGS02387000, USGS02392000, and USGS02413300. For each gauge station, we consider 20 years daily data from 1998 to 2017.

In this study, we investigate the ability of LSTM to simulate discharge based on the meteorological observations (i.e., precipitation, maximum and minimum daily temperatures). We divide the data into three parts: training, validation, and test data, where the training and validation data are referred to as calibration data for LSTM model training and hyperparameter tuning in the calibration period and the test data are unseen data for diagnosing the model performance in the validation period. For the three short-record catchments in ER, we reserve the last 1 year of records for model diagnosis as unseen test data and use the remaining records as calibration data for LSTM training and hyperparameter tuning. Since Quigley has only 2 years of streamflow observations (i.e., 1 year of data for model training), we use this catchment to explore whether LSTM can learn hydrologic dynamics from only 1-yr dataset. Pumphouse aggregates several headwater catchments, whereas Rock Creek is a small headwater catchment with relatively small area (Fig. 3). As a result, Pumphouse is less sensitive to the meteorological forcing than Rock Creek. Pumphouse and Rock Creek have 3 years of data each. With one more year of data, we can investigate the influence of data size on LSTM performance, and test whether LSTM can learn local rainfall–runoff process from 2 years of calibration data and produce a good prediction in the following year. Additionally, year 2017 is a wet year with a relatively large amount of precipitation, but the high precipitation causes different streamflow responses in Pumphouse and Rock Creek. Specifically, Pumphouse has a similar hydrologic variability from 2015 to 2017, whereas the streamflow pattern of Rock Creek in 2017 is dramatically different from that in years 2015 and 2016. Thus, these two catchments can also be used to investigate the capability of LSTM in out-of-distribution prediction. Finally, the four long-record gauge stations (one in ER
Watershed and the other three in ACT basin) provide more validation samples to verify our proposed modeling approach designed for data-poor catchments.

b. Physics-based hydrologic model

The physics-based hydrologic model considered in this study is the USGS Precipitation-Runoff Modeling System (PRMS), a deterministic, distributed-parameter model developed to evaluate the response of various combinations of climate and land use on streamflow and general watershed hydrology (Markstrom et al. 2015). The PRMS model is set up by using digital elevation models, spatial database of land use land cover and soil characteristics derived from the National Hydrologic Model (NHM-PRMS) database prepared by Regan et al. (2018). Based on the derived parameters from NHM-PRMS, the watershed is divided into subareas called hydrologic research units (HRUs) in which different components of streamflow are computed. The PRMS model is driven by meteorological data such as precipitation and daily minimum and maximum temperature aggregated for each HRU from Daymet. To be consistent with LSTM, we used the same data in LSTM calibration period to calibrate PRMS and the same data in the validation period to evaluate the PRMS prediction performance. We selected dominant PRMS model parameters from the table (https://water.usgs.gov/water-resources/software/PRMS/PRMS_tables_5.1.0.pdf), and utilized the widely applied shuffled complex evolution algorithm (Duan et al. 1992) for calibration at a daily scale by maximizing the Nash–Sutcliffe efficiency (NSE). A total of 1000 trials were performed with six shuffling loops, and the early terminating criterion was set at a percentage change in root-mean-square error (RMSE) of less than 0.01 ft³ s⁻¹ within six loops.

c. Numerical experiments

For all the catchments, we train LSTM models separately in the following model configurations.

- Standard LSTM (S-LSTM): a standard LSTM learned from \( t \) preceding days of meteorological forcing to simulate streamflow discharge at the next day.
- Bayesian LSTM (B-LSTM): a Bayesian LSTM learned from \( t \) preceding days of meteorological forcing to simulate streamflow discharge at the next day. The B-LSTM used 100 MC samples and further increasing the sample size led to little change in performance.
- Hybrid LSTM (H-LSTM): a standard LSTM trained on \( t \) preceding days of input sequences to simulate streamflow discharge at the next day, where the input sequences include the meteorological features and the PRMS model simulated discharge. The simulated discharge is concatenated to the meteorological data at each time step as another set of inputs.
- Standard LSTM with regularization: different regularization techniques are applied to the S-LSTM when overfitting is detected in the training. The regularization can be applied on the LSTM model structure such as model simplification and dropout or on the LSTM training such as the L2 norm penalty.

LSTM model calibration involves several hyperparameters such as those related to network architecture, learning rate, and the length of input sequence \( t \). The values of these hyperparameters could affect the model simulation significantly and need to be adjusted in the calibration for a good prediction performance. We performed hyperparameter tuning in the following way. First, we used the 10% of the calibration sample size as the validation set and the other 90% as the training set. We then considered a set of hyperparameter values and analyzed the loss function decay on the validation data. Last, we chose the hyperparameter values giving the optimal performance on the validation. In the tuning, we considered two types of loss functions, mean absolute error (MAE) and mean squared error (MSE), and found that MSE gave a better performance for most cases except the USGS02387000 gauge station in the ACT basin. In the numerical experiments, we first worked on the Pumphouse catchment and found that a LSTM model with 20 nodes and a learning rate of 0.001 gave a good validation performance after 200 epochs. Then we applied this LSTM architecture to start the hyperparameter tuning in other catchments and observed overfitting in Quigley which has the shortest records and the smallest calibration data, as shown in Fig. 4a.

To address overfitting, we employed L2 norm penalty and dropout with different coefficient/rate for regularization. Figures 4b and 4c indicate that L2 norm penalty addresses the overfitting but meanwhile it also penalizes the training performance causing a relatively large loss function value. And the penalty of 0.05 or 0.2 does not make much difference on performance. Figures 4d–f show that the application of dropout can mitigate the overfitting but seems not be able to eliminate it completely as we see the loss function of validation slightly goes up when close to the end of training. And a larger dropout rate of 0.6 even causes a worse training and validation performance. This observation is consistent with the finding from Gal and Ghahramani (2016) which suggests that applying the dropout to the input-to-hidden connection of LSTM may still suffer from some overfitting. Finally, we used L2 norm penalty of 0.2 and dropout rate of 0.1 in the LSTM simulation. We will further explore the influence of overfitting and the regularization techniques on LSTM performance in section 4.

The input sequence length \( t \) is another important hyperparameter in LSTM, which determines the length of the sequence of the input meteorological data used to generate the model output (i.e., discharge). The \( t \) simulates the period over which the influence of meteorological inputs is taken into account to calculate the discharge. However, as a data-driven model, the \( t \) in LSTM cannot be determined a priori with physical basis but can be tuned using calibration data. By balancing the influence of the calibration sample size (a large sample size requires a small \( t \)) and the impact of the temporal meteorological information (a large \( t \) leads to more information learning) on discharge simulation, we tested a set of \( t \) (e.g., \( t = \{10, 30, 60, 90\} \)) in hyperparameter tuning and found \( t = 60 \) days in Quigley and Pumphouse and \( t = 30 \) days in Rock Creek giving a better performance on the loss function. After we determined the hyperparameters, we used all the calibration data (including the training and validation sets) for the
final LSTM model calibration and used the reserved last 1 year of unseen test data to evaluate the model simulation. This strategy was also suggested in Kratzert et al. (2018).

For the gauge station 09112500, which has over decades of daily streamflow observation, we designed two numerical experiments. Both experiments used the last 19 years of records as test data (i.e., from 1 January 2000 to 1 January 2018) whereas the first experiment used the preceding 2 years of data (i.e., years 1998 and 1999) for calibration and the second one used all the remaining 19 years in the calibration period. For the first 2-yr calibration case, we predicted a daily discharge from 60 preceding days and for the second 19-yr calibration case, we were able to use the preceding 240 days. In the first experiment, we considered different types of LSTM models whereas in the second one we considered the S-LSTM only. The aim of the first experiment is to analyze how well the network trained by small-sample data can generalize into a long-term discharge prediction under a changing meteorological condition. Additionally, this experiment provides a large set of test data and a long validation period to further confirm our findings in the three short-record catchments. The second experiment is designed to investigate whether the increase of training size can improve the simulation performance of S-LSTM.

For the three long-record gauge stations in the ACT basin, we used the first 2 years of data for calibration (i.e., 1998–99) and used the remaining 18 years of records (i.e., 2000–17) as unseen test data to evaluate the model performance. As the ACT basin has different rainfall–runoff processes from the ER Watershed, the study in ACT can investigate whether the findings from the snowmelt dominated watershed are applicable to other basins where the snow has little influence on runoff.

d. Model evaluation criteria

We consider three criteria to evaluate model performance. The NSE and RMSE-observations standard deviation ratio (RSR) (Moriasi et al. 2007) are used to quantify the inconsistency and bias of model simulations from the observations. Besides, we calculate the errors of different models in estimating the peak flow magnitude and peak flow timing for the out-of-distribution prediction.

NSE is a normalized statistic that determines the relative magnitude of the residual variance compared to the observation data variance. NSE indicates how well the plot of observed versus simulated data fits the 1:1 line. NSE is computed as

\[
NSE = 1 - \frac{\sum_{i=1}^{T} (Y_{i}^{obs} - Y_{i}^{sim})^2}{\sum_{i=1}^{T} (Y_{i}^{obs} - Y_{mean})^2},
\]

where \(Y_{i}^{obs}\) is the \(i\)th observation, \(Y_{i}^{sim}\) is the \(i\)th simulated value, and \(Y_{mean}\) is the mean of observation, and \(T\) is the total number of observations. NSE ranges from negative infinity to 1. A value of 1 corresponds to a perfect match of modeled discharge to observations, and a negative NSE indicates that the model performs worse than the observed mean.

RSR standardizes RMSE using the observations standard deviation and is calculated as...
RSR incorporates the benefits of RMSE metric and includes a normalization factor, so that the resulting statistic can apply to various quantities. RSR varies from the optimal value of 0 to a large positive value. The lower RSR and the better the model simulation performance.

4. Results and discussion

In this section, we evaluate LSTM model performance and investigate the effect of the three strategies (i.e., regularization, Bayesian inference, and hybrid modeling) on the performance in the validation period using the reserved unseen test data. We discuss the results using two types of plots, the NSE values in both calibration and validation periods and the comparison between observed and simulated hydrographs. We first analyze the results of the three short-record catchments and then move to the discussion of the four gauge stations with long records.

a. Catchment Quigley

Catchment Quigley only has 774 days of records. In determination of LSTM hyperparameters, we observed that the architecture with 20 nodes caused an overfitting for the small training set (Fig. 4), and the regularization techniques such as L2 norm penalty and dropout can mitigate the overfitting. Figures 5a–d demonstrate the influence of the overfitted S-LSTM model and some regularization methods on the test data performance in the validation period. Figure 5a shows that the overfitted model having a validation NSE first goes up, then drops down and finally ends up with a negative value at epoch 200, resulting a very poor performance. Figures 5b–d indicate that applying regularization can improve the validation performance. For example, a simpler model with 10 nodes leads to a NSE value of 0.33 close to the PRMS result. However, this performance is not as good as the S-LSTM with 20 nodes before the overfitting happens. This suggests that a simple S-LSTM might not be sophisticated enough to learn the underlying complex hydrologic dynamics. That is, it may cause an opposite problem, underfitting. On the other hand, although L2 norm penalty is able to achieve a better performance at the end of training, it has a slow convergence. This is because the penalty limits LSTM capacity in both training and validation. Dropout also improves the simulation performance but the validation NSE shows a decreasing tendency as close to the end of training. This is because dropout does not eliminate the overfitting problem completely as discussed in Fig. 4d.

Bayesian and hybrid LSTM models show a good performance even in use of the network architecture with 20 nodes. As shown in Figs. 5e and 5f, the validation NSEs of B-LSTM and H-LSTM gradually increase along epochs (although with some fluctuation) and have a similar trend with the calibration NSEs, demonstrating a fast convergence and reliable performance. Furthermore, B-LSTM provides uncertainty information.
being able to show the model prediction confidence. Figure 6b depicts the observed and B-LSTM simulated discharge in calibration and validation periods where the gray bars quantify the prediction uncertainty. We can see that in both periods the predicted hydrograph fits the general trends of the observation well and in the falling limb of the validation period where B-LSTM has relatively large prediction bias, the prediction uncertainty bound is also large capable of enclosing most of the observed values and meanwhile showing lower confidence in these biased predictions.

Figure 6a summarizes the final NSEs in the validation period for the LSTM models. S-LSTM with L2 norm penalty and dropout show a higher NSE than the PRMS model. B-LSTM has a better performance than the S-LSTM model with regularization, and H-LSTM has the best prediction with an NSE value over 0.8. These results of NSEs are consistent with the RSR metric, where the RSR values of PRMS, S-LSTM, B-LSTM, and H-LSTM are 0.82, 1.04, 0.51, and 0.44, respectively, which also indicates the relatively better performance of H-LSTM. The H-LSTM model, fed with the physics information from the PRMS simulation outputs, has additional data to mitigate overfitting. Additionally, the extra physical information enables H-LSTM to quickly reach a high NSE in both training and validation. Figure 7a shows the simulated hydrograph at the beginning of H-LSTM training with randomly initialized weights. We can see that even with an arbitrary set of parameters, H-LSTM is able to approximate the general trend of the observed hydrograph with the right period of low flows and peak flows and its simulation is similar to the PRMS results. On the other hand, if the PRMS simulation has a large bias, this would deteriorate the H-LSTM performance. As shown in Fig. 7b at the end of training, H-LSTM has relatively large errors in simulating the low flows after the falling limb where the PRMS model has a poor performance. In contrast, S-LSTM, without the physics information, produces a non-informative simulation with the randomly initialized weights (Fig. 7c) and due to overfitting, at the end of training the simulated hydrograph of S-LSTM in the validation period is similar to that in the calibration period with an overestimation in the peak flow (Fig. 7d).

In application to Quigley catchment, we have the following findings:

(i) Using 1 year of data for calibration, S-LSTM can suffer from overfitting resulting in a poor simulation performance.

(ii) Model simplification could lead to underfitting, and in this work L2 norm penalty and dropout demonstrate effectiveness of regularization to improve the performance, although the penalty requires a long training and dropout might not overcome the overfitting completely.

(iii) Bayesian LSTM provides uncertainty information as an additional insight to point estimation and improves prediction confidence.

(iv) Hybrid LSTM, by integrating physics information, accelerates model training and enhances model performance.

(v) Even with 1 year of calibration data, the S-LSTM with an effective regularization, B-LSTM, and H-LSTM can produce a comparable streamflow simulation as the PRMS model.

b. Catchment Pumphouse

Pumphouse has 3 years of data. After reserving 1 year of test data for validation, we have 2 years of data for calibration. Figure 8 summarizes the LSTM model simulation results. We want to highlight the following four points. First, with a 2-yr calibration data, S-LSTM produces a good and reliable simulation of discharge in Pumphouse. As shown in Fig. 8a, S-LSTM achieves a high NSE of 0.8 in validation after a few epochs and then gradually improves the performance till getting a NSE of 0.86 at the end of calibration. Additionally, Fig. 8d demonstrates that S-LSTM can simulate the complex rainfall–runoff dynamics pretty well with the simulated hydrograph closely matching the observations except a small bump at the beginning of the validation period. S-LSTM has a small validation RSR of 0.37 and it accurately simulates the peak flow with an absolute error of 3.4 m$^3$s$^{-1}$ and a 3-day difference in peak flow timing compared to the observation. Additionally, the extra physical information enables H-LSTM to quickly reach a high NSE in both training and validation. Figure 7a shows the simulated hydrograph at the beginning of H-LSTM training with randomly initialized weights. We can see that even with an arbitrary set of parameters, H-LSTM is able to approximate the general trend of the observed hydrograph with the right period of low flows and peak flows and its simulation is similar to the PRMS results. On the other hand, if the PRMS simulation has a large bias, this would deteriorate the H-LSTM performance. As shown in Fig. 7b at the end of training, H-LSTM has relatively large errors in simulating the low flows after the falling limb where the PRMS model has a poor performance. In contrast, S-LSTM, without the physics information, produces a non-informative simulation with the randomly initialized weights (Fig. 7c) and due to overfitting, at the end of training the simulated hydrograph of S-LSTM in the validation period is similar to that in the calibration period with an overestimation in the peak flow (Fig. 7d).
NSE of 0.4 and RSR of 0.77 (Fig. 8b) in the validation. Third, B-LSTM provides useful uncertainty estimation although it is not necessary to correct the simulation bias from the S-LSTM. The UQ measures how likely the model can simulate the observed discharge (Fig. 8c) and what possible simulated values could be due to data shortage, which provides more detailed information to improve our understanding of the model simulation compared to the S-LSTM that only gives point estimates. Moreover, the uncertainty adds credible information to the estimation which enhances the model prediction confidence. Specifically, for the accurately simulated discharge, the model uncertainty is low, whereas for the inaccurate estimation, the standard deviation of the prediction is relatively high (Fig. 8e). For example, B-LSTM shows a large bias in the bump area like S-LSTM, but the uncertainty bound in this area is also relatively large, suggesting the prediction with low confidence. Last, the performance of H-LSTM depends on the accuracy of the physics-based model. Figure 8f indicates that PRMS model has a relatively large error in simulating the falling limb of the hydrograph. H-LSTM picks up this bias in calibration and affects its prediction in this region, although it still performs better than the PRMS attributed to its effective learning from other input sequences.

**c. Catchment Rock Creek**

Catchment Rock Creek also has 3 years of data, but different from Pumphouse, the hydrograph in the validation period is dramatically distinct from that in the calibration period. This greatly increases the difficulties in LSTM simulation as the data-driven model needs to predict some patterns it has never seen before. Figure 9a shows that as epochs increase, the validation NSE of S-LSTM fluctuates around 0.2 with no signs of improvement while the calibration NSE keeps climbing up. This implies that the hydrologic dynamic learned from the calibration period is not generalized well to the different validation period and the S-LSTM model seems lack of the capability to improve the generalization. As shown in Fig. 9d, the observed peak flow in the validation period is much higher than that in the calibration period. Both S-LSTM and PRMS models are unable to predict the peak streamflow in the validation with a large underestimation, where the peak flow estimates of S-LSTM and PRMS are 0.68 and 0.5 m$^3$s$^{-1}$, respectively, compared to the observed peak flow of 2.48 m$^3$s$^{-1}$. B-LSTM improves the performance slightly with a little higher NSE of 0.31 than the value of 0.24 in S-LSTM (Fig. 9b), but B-LSTM still significantly underestimates the peak flow even with the uncertainty bound (Fig. 9e).

Furthermore, H-LSTM gives the best NSE in the out-of-distribution prediction. Figure 9c depicts that, at the beginning of the training, H-LSTM already has a validation NSE higher than that of PRMS and the NSE keeps increasing till hit the value of 0.5 at the end of training. Although the prediction performance of H-LSTM is still relatively low, the result is promising given the difficulty of the out-of-distribution prediction. Figure 9f shows the H-LSTM simulated hydrograph. In comparison to the PRMS simulation and S-LSTM simulation in the validation period, despite of the underprediction of the peak flow from all models, it seems that H-LSTM has less bias.
FIG. 8. (a) The NSE values in calibration and validation periods for the standard LSTM; (b) the final NSE values in the validation period for different model configurations; (c) the probabilistic prediction of discharge at a given day from Bayesian LSTM; and the simulated hydrographs from (d) standard LSTM, (e) Bayesian LSTM, and (f) hybrid LSTM at catchment Pumphouse.
FIG. 9. The NSE values in calibration and validation periods for (a) the standard LSTM and (c) the hybrid LSTM; (b) the final NSE values in the validation period for different model configurations; and the simulated hydrographs from (d) standard LSTM, (e) Bayesian LSTM, and (f) hybrid LSTM at catchment Rock Creek.
in estimating the bell-shape pattern by leveraging the simulation capability from the other two models.

In application to the catchments Pumphouse and Rock Creek with 3 years of data, we have the following findings:

(i) If the hydrologic variability in the validation period is similar to the calibration period (Pumphouse case), LSTM models can predict discharge pretty well, superior to the PRMS model.
(ii) When the hydrologic variability in the validation period is dramatically different from the calibration period (Rock Creek case), the hybrid model has better generalizability but its performance greatly relies on the physics-based model accuracy.
(iii) To enhance the performance in out-of-distribution prediction, the key is to collect more data and improve our physical understanding of the hydrologic system.

In section 4d below, we use four long-record gauge stations from two diverse basins to further demonstrate these points.

d. Catchment with long gauge records

The gauge station 09112500 in ER Watershed has decades of continuous streamflow observations. Figure 10a shows the simulation results of the first experiment where we used 2 years of data (1998–99) for calibration and the following 19 years of test data (2000–18) for validating the performance. Figure 10b quantifies the validation performance in each year for models calibrated in both experiments (i.e., calibrated by 2 years of data and by 19 years of data). Figure 11 analyses the model performance in the first experiment in detail where Fig. 11a demonstrates the predicted hydrograph in 2012 from S-LSTM and Fig. 11b shows the simulated discharge of B-LSTM in 2016.

Several insights can be gained from this long-record data analysis. First, LSTM models can simulate the 19 years of discharge well, although the models were calibrated with only 2 years of data. The overall NSE values of the 19 years is 0.71 of S-LSTM, 0.72 of B-LSTM, 0.85 of H-LSTM, and 0.64 of PRMS. Second, S-LSTM has relatively poor performance when the predicted hydrograph has different variability from that in the calibration period. As shown in Fig. 10, S-LSTM has negative NSEs in year 2002, 2012, and 2018 where the peak flow is much lower than that in calibration period of 1998–99. Third, although B-LSTM is not able to improve the performance in out-of-distribution prediction significantly, it provides useful uncertainty information (Fig. 11b). Four, H-LSTM

![Figure 10](https://example.com/image10.png)

**Fig. 10.** (a) The observed and simulated hydrographs from different models at the long-record gauge station 09112500, where the LSTM models are calibrated using 2 years (1998–99) of data and the calibrated model is used to simulate discharge in 2000–18 for validation. (b) An individual NSE value at each of the 19 years in the validation period; model simulations are calibrated from 2 years of data, except the one calibrated by 19 years of data as represented by the black line.
produces the best prediction and it greatly improves the performance in the out-of-distribution prediction showing a high NSE for all the 19 years. As shown in Fig. 11a when S-LSTM overestimates the peak streamflow in 2012 (where the observed peak flow arrives at the 146th day with a value of 16.1 m$^3$s$^{-1}$ and the S-LSTM estimated peak flow is 35.6 m$^3$s$^{-1}$ arriving at day 120), H-LSTM is able to estimate the peak flow with a close value of 12.2 m$^3$s$^{-1}$ at the similar day to the observation. Last, the relatively poor performance of S-LSTM in the three dry years is majorly caused by the limited calibration data and lack of physical knowledge. Imbedding physical information (e.g., using H-LSTM) or increasing calibration data size could improve the performance. For example, when we calibrated the S-LSTM using 19 years of data, we not only increased the overall prediction NSE from 0.71 to 0.87, but also improved the performance of each single year especially for the three dry years (the black line in Fig. 10b). These findings are consistent with the conclusions drawn from the three short-record catchments.

The three gauge stations in the ACT River basin have dramatically different hydrographs than those in the ER Watershed. In particular, the seasonality in ACT is less clear (Fig. 12) as the ACT basin receives precipitation primarily from rainfall with minimal influence of snow on runoff. Despite the differences in the hydrographs, the investigation in ACT shows similar findings as the four catchments in ER area. As demonstrated in Fig. 12, LSTM models can predict the discharge in 2000–17 well, even though the models were calibrated with only 2 years of data and the 18 years in the validation period have different hydrograph patterns than the calibration period. Both S-LSTM and H-LSTM show better results than PRMS alone, with better performance for most years and fewer catastrophic failures (NSE < 0). In addition, H-LSTM gives similar performance to S-LSTM in most years, but does not result in catastrophic failures. Several reasons could cause the catastrophic failures of S-LSTM prediction such as the unsuitable training loss, the small calibration data, and the difference of hydrological dynamics in the calibration and validation periods. In USGS02387000, we observed that the use of MSE as the training loss resulted in a negative NSE in 2007 while the use of MAE loss greatly improved the S-LSTM performance with all positive NSEs. Whereas, for the other two gauge stations, the choice of loss functions did not make much difference. The catastrophic failures of S-LSTM are majorly caused by the hydrological dynamics difference. For example, in USGS02392000, the year 2008 in the validation period has similar precipitation to 1999 in the calibration period; however, the discharge in 2008 is only about half of the discharge in 1999. Also, the year 2017 has slightly higher precipitation than that in 1998, but the discharge in 2017 is much lower than the discharge in 1998. All the other years in validation period have consistent hydrological patterns with the calibration period. These different patterns in 2008 and 2017 can weaken the S-LSTM learning capability and result in negative NSEs in these 2 years.

Increasing the calibration data size is expected to improve the prediction performance for all the LSTM models and PRMS model. However, this study specifically focus on investigating model prediction capabilities in poorly monitored watersheds with short observation records. Thus, the discussion and insights from these numerical experiments will have vital implications for streamflow simulation in data-scarce basins.

5. Conclusions

This work investigated the capability of LSTM in simulating discharge from meteorological observations with limited calibration data. The LSTM model is data hungry in nature. When applying it to data-scarce basins with short records of streamflow observations and in lack of detailed and reliable catchment attributes, we need to overcome three challenges: overfitting, UQ, and out-of-distribution prediction. We explored different regularization techniques to address overfitting, proposed a Bayesian LSTM model to estimate the prediction uncertainty, and designed a physics-informed hybrid LSTM model to improve out-of-distribution prediction.

We applied the methods to seven catchments, where four of them are snowmelt dominated and the other three are barely...
influenced by snow. The major findings are summarized as follows:

1) When using 1 year of data for calibration, standard LSTM can suffer from overfitting and results in a poor simulation. Regularization techniques such as L2 norm penalty and dropout can mitigate overfitting and improve simulations in some degrees. Bayesian and hybrid LSTM models do not suffer from overfitting and yield superior model performance.

2) Using 2 years of data for calibration, if the hydrologic variability in the prediction period is similar to that in the calibration period, LSTM models can predict discharge very well. Whereas if the hydrologic variability in the prediction and calibration periods is dramatically different,
only physics-informed hybrid LSTM model performs reasonably well in this out-of-distribution simulation.

3) Bayesian LSTM provides useful uncertainty information, which improves prediction understanding and credibility about how likely the model can predict the observed discharge and what possible predicted values could be due to data shortage.

4) Physics-informed hybrid model presents the best performance in terms of fast learning, high prediction accuracy and the capability of generalization, but its performance greatly relies on the physics-based hydrologic model accuracy.

5) In most numerical experiments, LSTM models are able to simulate discharge with accuracy comparable to or even higher than the physics-based PRMS model.

Note that we drew the above conclusions under the context of a single basin simulation with limited training data. These findings may not be fully applicable to regional modeling and simulations in ungauged basins. This study provides important insights not only from a scientific point of view, but also from a practical perspective. The vast majority of watersheds are poorly monitored with sparse and low-quality data, especially in developing regions/countries (Alipour and Kibler 2018).

Insights from this study have the potential to improve streamflow simulation in these data-scarce regions and enhance local water resources management.

Applying data-intensive LSTM for streamflow simulations in data-scarce basins has unavoidable barriers. In this work, we investigated three strategies to mitigate the problems. However, due to the data-driven nature of the LSTM, it will always strongly depend on the available observations for training. Thus, even if less data are required by using the regularization and Bayesian inference techniques, LSTM may still have some limitations compared to physics-based models especially in application to new conditions such as different climate and land use change. Adding physics or physical model simulations to the LSTM can enhance the simulation performance as demonstrated in this study. In the future, we will explore other possible ways in integration of physical modeling and machine learning, including the use of physics information to constrain the LSTM loss function and for the hybrid model construction.

Acknowledgments. This research was supported by the Oak Ridge National Laboratory (ORNL) AI initiative project and ExaSheds project supported by the Office of Biological and Environmental Research in the DOE Office of Science. ORNL is managed by UT-BATTELLE for DOE under Contract DE-AC05-00OR22725.

Data availability statement. The long-record data are openly available which can be downloaded from U.S. Geological Survey. The short records from headwater gauging stations in East River Watershed is available on ESS-DIVE (https://ess-dive.lbl.gov). The code is available at https://github.com/demiludan/LSTM_JHMpaper, which can be used to implement different LSTM model configurations and reproduce our results with the provided data.

REFERENCES


Lu, D., S. Liu, and D. Ricciuto, 2019: An efficient Bayesian method for advancing the application of deep learning in Earth science. 2019 Int. Conf. on Data Mining Workshops, Beijing, China, IEEE, 270–278, https://doi.org/10.1109/ICDMW.2019.00048.


