Inductive Predictions of Extreme Hydrologic Events in The Wabash River Watershed

Nicholas Majeske[*], **Bidisha Abesh**[†], **Chen Zhu**[†], **Ariful Azad**[*]
nmajeske@iu.edu, bidabesh@iu.edu, chenzhu@iu.edu, azad@iu.edu
[*]Department of Intelligent Systems Engineering, [†]Department of Earth and Atmospheric Sciences
Indiana University Bloomington, IN, USA

Abstract

We present a machine learning method to predict extreme hydrologic events from spatially and temporally varying hydrological and meteorological data. We used a timestep reduction technique to reduce the computational and memory requirements and trained a bidirection LSTM network to predict soil water and stream flow from time series data observed and simulated over eighty years in the Wabash River Watershed. We show that our simple model can be trained much faster than complex attention networks such as GeoMAN without sacrificing accuracy. Based on the predicted values of soil water and stream flow, we predict the occurrence and severity of extreme hydrologic events such as droughts. We also demonstrate that extreme events can be predicted in geographical locations separate from locations observed during the training process. This spatially-inductive setting enables us to predict extreme events in other areas in the US and other parts of the world using our model trained with the Wabash Basin data.

1 Introduction

Extreme hydrologic events such as floods and droughts are among the most severe weather-related disasters that cause widespread damage in agriculture, wildlife habitats, and human properties. For example, the 2012 drought in the continental United States resulted in an estimated $30 billion in mostly agricultural losses [12]. Hence, predicting the occurrence, frequency, and severity of extreme hydrologic events is extremely important for societal well-beings [11].

In this paper, we use machine learning (ML) models to predict extreme hydrologic events related to droughts and floods. A drought is defined as prolonged periods of below-average water availability [6] and a flood is defined as a sudden increase of water level. The severity of droughts and floods can be quantitatively measured by soil water content and stream flows. Thus, predicted soil water (SW) and stream flow (SF) values can be used to accurately forecast future droughts and floods. Here, we use bidirectional Long Short-term Memory (BLSTM) networks to predict SW and SF from historic SW, SF, precipitation, temperature, and other records. Based on the predicted values of SW and SF, we forecast the occurrence and severity of extreme events.

We used both observed and simulated data spanning over 84 years from the Wabash River Watershed located in Midwest USA. We trained BLSTM networks– one for each subbasin– to predict SW and SF from time reduced historic data. These simple BLSTM networks, when used with a time reduction technique, outperform autoreggressive methods such as ARIMA [4] and VAR [13] and perform comparably with complex models such as GeoMAN [9]. To this end, we considered two settings: (a) *spatially-transductive* learning where a BLSTM network is trained and tested on data from the same subbasin and (b) *spatially-inductive* learning where a BLSTM network is trained on data from one subbasin and tested on data from a different subbasin. The BLSTM networks perform well for both settings.

2 Materials and Methods

2.1 Wabash River Watershed Time-Series

The Wabash River Watershed time-series consists of daily hydrological and meteorological records from 1929-01-01 to 2013-12-31 for a total $|T| = 31,046$ timesteps. The basin is subdivided into $|S| = 1,276$ distinct subbasins each of which record $|F| = 5$ features; 2 hydrological (stream flow and soil water content) and 3 meteorological (precipitation, minimum and maximum temperature). In our implementation and for the purposes of this paper, the complete time-series is a 3^{rd}-order tensor of dimensions $|T| \cdot |S| \cdot |F|$. Hydrological features of the time-series were obtained from simulation using the Soil and Water Assessment Tool (SWAT) hydrological model [1, 2]. Meteorological features are observed records used in-part to parameterize SWAT and were obtained from the University of Notre Dame [3].

34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.

2.2 Timestep Reduction

The daily timestep granularity of the Wabash River Watershed time-series is precise but computationally cumbersome and superfluous in modeling extreme events such as drought which span numerous timesteps. We apply timestep reduction to reduce timestep granularity and significantly decrease memory, computational, and modeling complexity with minimal cost to model accuracy. To introduce this algorithm, let $|T|$ be the timestep count of the original time-series, w be reduction window size (7: weekly), and s be reduction window stride. Timestep reduction is trivially applied using $s = w$ but we generalize this to any s such that s evenly divides w allowing control over timestep distinctiveness (varying s) in addition to granularity (varying w). The common sliding window method for extracting window samples requires timestep contiguity. To ensure contiguity persists through timestep reduction, we create $\frac{w}{s}$ reduced time-series channels. Each channel is created by applying the simple $s = w$ reduction at an offset unique to that channel defined $c_{\text{off}} = c_{id} \cdot s$ where $c_{id} \in \{0, 1, ..., \frac{w}{s} - 1\}$. Using the sliding window method, we then extract input/output window samples from each channel separately and pool them to create the final window sample sets X and Y.

Table 1 briefly analyzes the effect of w and s on computational complexity and model generalization measured by testing set normalized root mean squared error (NRMSE) defined equation in (1). We find increasing w decreases epoch operation count at the cost of window sample count and decreasing s reduces window sample count loss at the cost of window sample distinctiveness. Ultimately, timestep reduction to weekly units with $w = 7$ and stride $s = 1$ provides an optimal balance between computational complexity and model generalization.

w	s	Model Timesteps	Testing NRMSE SW	Testing NRMSE SF	Operation Ratio
1	1	84 → 28	0.0831	0.0657	1
7	1	12 → 4	0.0815	0.0682	1/7
7	7	12 → 4	0.0887	0.0701	1/49
14	1	6 → 2	0.0828	0.0711	1/14
14	14	6 → 2	0.0917	0.0780	1/196
28	1	3 → 1	0.0801	0.0806	1/28
28	28	3 → 1	0.0889	0.0894	1/784

Table 1: Effect of parameters w and s on model generalization and computational complexity.

2.3 Feature Standardization

To facilitate model optimization, we standardize all predictor and response features. The Wabash River Watershed time-series presents an interesting challenge for standardization since mean and standard deviation vary across both temporal (T) and spatial (S) domains. Across T, we see a cyclical change in mean and standard deviation coinciding with seasonality. Across S we see a wide range of mean and standard deviations coinciding with low (potentially upstream) and high (potentially downstream) yield subbasins. In order to correctly standardize our features we calculate mean and standard deviation specific to the day of year and subbasin resulting in 3^{rd}-order tensors of dimensions $366 \cdot |S| \cdot |F|$.

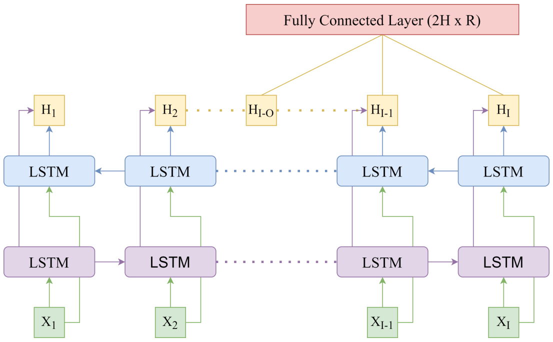

Figure 1: The Bidirectional LSTM network architecture.

2.4 Bidirectional LSTM Networks

The LSTM network proposed in [8] receives as input a series of I timesteps each containing P predictors and recurrently transforms them into a set of I hidden states $H_1, ..., H_I$ each containing h hidden units. Each hidden state H_t encodes information observed in all previous and current inputs $X_1, ..., X_t$ and all previous hidden states $H_1, ..., H_{t-1}$. A significant limitation of this architecture is the constraint of any hidden state H_t to encode only information observed in previous and current timesteps $1, ..., t$ while valuable predictor information for timestep t may exist in future timesteps $t + 1, ..., I$. Specifically, if observing I input timesteps and predicting O output timesteps, this architecture is constrained to use only the first $I - O + t$ timesteps when predict timestep t of the output series. For non-trivial timestep mappings ($I \geq O > 1$ and $I - O = \epsilon$) a significant $I - t$ timesteps are omitted in the prediction of timestep t though these timesteps share greater temporal proximity with timestep t. For this reason we use bidirectional LSTM networks which remove the constraint by processing information both forwards and backwards through time. Thus, hidden state H_t may now encode information from all input timesteps $X_1, ..., X_t, ...X_I$ and all other hidden states $H_1, ..., H_{t-1}, H_{t+1}, ..., H_I$.

3 Results

3.1 Experiment Settings

We implemented our BLSTM in PyTorch 1.5.0 [10] and executed all experiments on an Acer Predator G3-710 equipped with an Intel Core i7-7700 at 3.6GHz, 32GB RAM, and Nvidia's GeForce 1070 (1920 CUDA

	Training		Testing (spatially transductive)			Testing (spatially inductive)		
Model	Subbasin	Time (sec)	Subbasin	SW	SF	Subbasin	SW	SF
Naive Last-Timestep	1	N/A	1	0.0950	0.0867	2	0.0783	0.1898
ARIMA	1	0.58	1	0.2201	0.0871	2	0.1898	0.1968
GeoMAN	1	232.6	1	0.0815	0.0677	2	0.0734	0.1565
BLSTM	1	51.4	1	0.0815	0.0682	2	0.0768	0.1599
Naive Last-Timestep	141	N/A	141	0.0920	0.1275	142	0.0916	0.1092
ARIMA	141	0.72	141	0.2060	0.1272	142	0.2000	0.1063
GeoMAN	141	230.5	141	0.0787	0.0961	142	0.0803	0.0814
BLSTM	141	51.7	141	0.0806	0.0985	142	0.0810	0.0845
Naive Last-Timestep	1275	N/A	1275	0.0977	0.1220	1276	0.1040	0.1240
ARIMA	1275	0.53	1275	0.1746	0.1786	1276	0.1942	0.1322
GeoMAN	1275	231.4	1275	0.0908	0.0842	1276	0.0892	0.1202
BLSTM	1275	52.1	1275	0.0808	0.0849	1276	0.0860	0.1435

Table 2: Model performances (normalized RMSE scores) when predicting soil water and stream flow.

cores, 1.5GHz, 8GB memory). The entire time-series was split into training, validation, and testing sets containing all timesteps over 1929/10/01-1997/09/30, 1997/10/01-2005/09/30, and 2005/10/01-2013/09/30 respectively. To further evaluate the efficacy of our model, we established 3 baselines for comparison:

- **Naive Last-Timestep**: A non-parametric model predicting for all timesteps O observation X_I.
- **ARIMA** [4]: A classical auto-regressive model for time-series forecasting.
- **GeoMAN** [9]: A complex LSTM network with temporal and spatial attention.

We evaluated both the spatially transductive and inductive capability of our BLSTM and all 3 baselines for subbasin pairs $\{s_{trans}, s_{ind}\} \in \{\{1,2\},\{141,142\},\{1275,1276\}\}$. Each model was optimized to the training set of the spatially transductive subbasin $s_{trans} \in \{1, 141, 1275\}$ saving the model state with minimum MSE loss over validation set. Our best testing set NRMSE for BLSTM ($I = 12$ weeks, $O = 4$ weeks, $R = |\{SoilWater, StreamFlow\}| = 2$) was acheived using $P = 6$ predictors including {Date, Minimum Temperature, Maximum Temperature, Precipitation, Soil Water, Stream Flow}, $h = 256$ hidden units, mini-batch stochastic gradient descent (mini-batch size 128), learning rate 0.01 (and learning rate decay 0.001), 0.001 l_2 regularization, Xavier initialization, and 50 epochs of training.

$$\text{NRMSE} = \frac{\sqrt{E(\hat{y} - y)^2)}}{y_{max} - y_{min}} = \frac{\text{RMSE}}{y_{max} - y_{min}} \tag{1}$$

3.2 Transductive and Inductive Predictions of Soil Water

Table 2 shows testing set NRMSE on SW for our BLSTM and all 3 baselines. ARIMA performs significantly worse in predicting SW. Upon further investigation of the plotted predictions, ARIMA is predicting for 4 weeks an approximate mean over the 12 observed weeks. SW varies significantly at the week-to-week level causing this type of prediction to systematically over and under project. In all cases, GeoMAN and BLSTM outperform Naive Last-Timestep but we see either comparable NRMSE between GeoMAN and BLSTM or one model outperform the other. GeoMAN outperforms BLSTM for subbasin pairs $\{1, 2\}$ and $\{141, 142\}$ but BLSTM significantly outperforms GeoMAN for subbasin pair $\{1275, 1276\}$. The otherwise even performance between GeoMAN and BLSTM is interesting considering the discrepancy in model complexity. This is an advantage to BLSTM, which sees a $4.5\times$ speed-up in training runtime over GeoMAN given an identical training regime.

Moving from spatially transductive to inductive prediction shows either reduced, approximately unchanged, or increased NRMSE. For subbasin pair $\{1, 2\}$ spatial induction outperforms transduction while performance is approximately equal in subbasin pair $\{141, 142\}$. Performance is somewhat degraded when moving to inductive prediction for subbasin pair $\{1275, 1276\}$ for BLSTM but marginally improved for GeoMAN. The left plot of Fig. 2 shows spatially transductive and inductive SW predictions for subbasin pair $\{1, 2\}$ using BLSTM, where both settings accurately predict ground truth values.

3.3 Transductive and Inductive Predictions of Stream Flow

Table 2 additionally shows testing set NRMSE on SF for our BLSTM and all 3 baselines. ARIMA performs on-par with Naive Last-Timestep for all subbasin pairs with the exception of $\{1275, 1276\}$ in which performance is significantly worse. GeoMAN and BLSTM show approximately equal performance for spatially transductive subbasins but GeoMAN systematically outperforms BLSTM for spatially inductive subbasins. This is likely due to the superior complexity of GeoMAN relative to BLSTM allowing for better fit to the high frequency of extreme values we observe in SF. The right plot of Figure 2 overlays spatially transductive and inductive SF predictions for subbasin pair $\{1, 2\}$. This plot confirms BLSTM's trouble in predicting extreme values in SF which accounts for the higher NRMSE we observe.

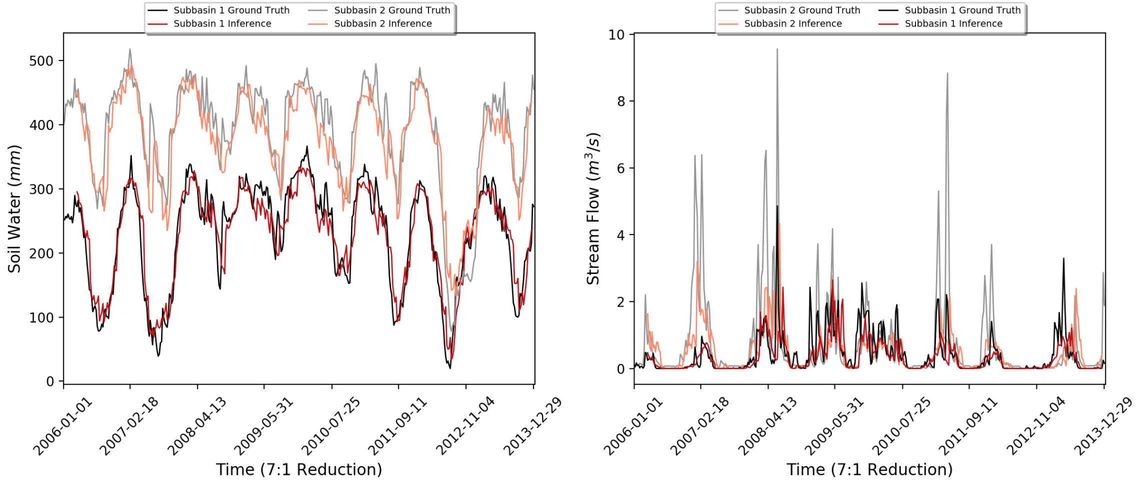

Figure 2: Testing set inference for spatially transductive and inductive settings on (left) soil water and (right) stream flow using subbasins 1 and 2.

3.4 Extreme Event Prediction

Each timestep of the ground truth and predicted curves are classified according to a standardized index simplified from the drought index presented in [5]. Specifically, each predicted timestep is converted to its z-score z and classified based on the following intervals: Normal: $z \in [-1, 1]$, Moderate: $z \in [-1.5, -1)$ *or* $z \in (1, 1.5]$, Severe: $z \in [-2, -1.5)$ *or* $z \in (1.5, 2]$, and Extreme: $z \in (-\infty, -2)$ *or* $z \in (2, \infty)$.

Fig. 3 shows a subsection of the testing set for subbasin 1 with extreme events labeled for ground truth and prediction on soil water. The blue regions enclose normal, moderate, and severe intervals with extreme events ($|z| > 2$) falling outside of all regions. Currently, our BLSTM tends to under project soil water values causing missclassification of some events to less extreme events. This phenomena is even more prevalent in stream flow prediction where extreme values are more frequent and of greater magnitude.

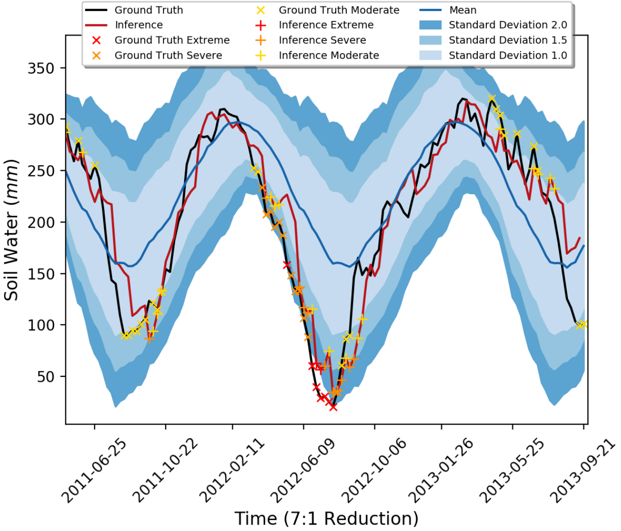

Figure 3: Extreme event classification on soil water for subbasin 1.

4 Concluding Remarks

In this paper, we apply BLSTM networks and 3 baseline models to predict soil water and stream flow in both spatially transductive and inductive settings. We apply timestep reduction to decrease memory, computational, and model complexity and train a BLSTM to comparable generalization w.r.t. the highly complex GeoMAN [9] at a fraction of the runtime. Using our soil water and stream flow predictions, we additionally perform extreme event classification for each timestep. Our experiments suggest (1) a relatively simple model can perform comparably well against complex models, (2) spatial induction can be performed with reliable accuracy for soil water but is currently unreliable for stream flow and (3) extreme event classification can be done with reasonable accuracy. In future work we plan to incorporate spatial information into the BLSTM architecture and scale the number of training subbasins. To handle the increased memory, computational, and model complexity we will also implement batch parallelism as described in [7].

References

[1] J. G. Arnold, R. Srinivasan, R. S. Muttiah, and J. R. Williams. Large area hydrologic modeling and assessment part i: model development 1. *JAWRA Journal of the American Water Resources Association*, 34(1):73–89, 1998.

[2] J. G. Arnold, J. R. Williams, and D. R. Maidment. Continuous-time water and sediment-routing model for large basins. *Journal of Hydraulic engineering*, 121(2):171–183, 1995.

[3] C. Boryan, Z. Yang, R. Mueller, and M. Craig. Monitoring US agriculture: the US department of agriculture, national agricultural statistics service, cropland data layer program. *Geocarto International*, 26(5):341–358, 2011.

[4] G. E. Box and D. A. Pierce. Distribution of residual autocorrelations in autoregressive-integrated moving average time series models. *Journal of the American statistical Association*, 65(332):1509–1526, 1970.

[5] J. R. Dierauer and C. Zhu. Drought in the twenty-first century in a water-rich region: Modeling study of the wabash river watershed, usa. *Water*, 12(1):181, 2020.

[6] J. A. Dracup, K. S. Lee, and E. G. Paulson Jr. On the definition of droughts. *Water resources research*, 16(2):297–302, 1980.

[7] A. Gholami, A. Azad, P. Jin, K. Keutzer, and A. Buluc. Integrated model, batch, and domain parallelism in training neural networks. In *Proceedings of the 30th on Symposium on Parallelism in Algorithms and Architectures*, pages 77–86, 2018.

[8] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[9] Y. Liang, S. Ke, J. Zhang, X. Yi, and Y. Zheng. GeoMAN: Multi-level attention networks for geo-sensory time series prediction. In *IJCAI*, pages 3428–3434, 2018.

[10] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. dAlché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

[11] M. Reichstein, G. Camps-Valls, B. Stevens, M. Jung, J. Denzler, N. Carvalhais, et al. Deep learning and process understanding for data-driven earth system science. *Nature*, 566(7743):195–204, 2019.

[12] B. R. Rippey. The US drought of 2012. *Weather and Climate Extremes*, 10:57–64, 2015.

[13] E. Zivot and J. Wang. Vector autoregressive models for multivariate time series. *Modeling financial time series with S-PLUS®*, pages 385–429, 2006.