Runoff Forecast Improvement Using LSTM-Based Error Prediction Models

John Waugh, Nick Diglio

Appalachian State University

[1]waughjm@appstate.edu

[2]diglionj@appstate.edu

***Abstract*— Accurate runoff forecasting is crucial for flood prediction, water resource management, and various hydrologic analyses. The United States National Water Model (NWM), developed by the National Oceanic and Atmospheric Administration (NOAA), provides short-range runoff forecasts, but these forecasts can suffer from systematic errors, particularly for longer lead times. This study explores the application of Long Short-Term Memory (LSTM) models to correct these forecast errors. Specifically, we developed a deep learning model to predict the errors (residuals) in NWM runoff forecasts, using observed runoff data from the United States Geological Survey (USGS). We then applied the predicted errors to correct the NWM forecasts. The model was trained and evaluated using data from two stations in the United States over a period spanning April 2021 to April 2023. Our results show improvements in the forecast accuracy, with key performance metrics such as the Coefficient of Correlation (CC), Root Mean Square Error (RMSE), Percent Bias (PBIAS), and Nash-Sutcliffe Efficiency (NSE) showing improvements across various lead times.**

I. Introduction

Accurate runoff forecasting is critical for flood prediction, water resources management, and various hydrologic analyses. Physically based numerical hydrological models such as the National Water Model (hereafter referred to as NWM) have become essential tools for simulating runoff and streamflow over large domains, including short-term forecasts for flood warning systems. Despite the advancement in these models, they are subject to significant forecast errors. Forecast errors in hydrologic models can stem from uncertainties or inaccuracies in modeling input data, uncertainties or inaccuracies in modeled outputs used for calibration, uncertainties or inaccuracies in modeling parameters and uncertainties caused by imperfect model structures. If left unaddressed, these errors in the flow prediction can lead to inaccurate hydrological analyses. In the context of flood forecasting, where lead times range from one to eighteen hours or more, inaccurate hydrological analyses can lead to delayed warnings or unnecessary emergency actions. Thus, improving the performance of the NWM to ensure proper error estimation and prediction is essential in ensuring the reliability of these hydrological models.

Promising alternatives for enhancing hydrological model outputs have been introduced thanks to the advancements in deep learning post-processing. Deep learning architectures, particularly recurrent models such as the Long Short-Term Memory network (hereafter referred to as LSTM), have demonstrated the ability to model nonlinear relationships in time series data and are well-suited for hydrological forecasting. Prior studies have shown that these models can be used either to directly predict streamflow or to learn and correct errors from physically based model forecasts. Frame et al. (2020) used an LSTM approach to improve the performance of NWM for runoff simulation by considering the output of the NWM as input variables for their LSTM model.

In this study, we adopt a LSTM-based post-processor to improve the short-term runoff forecasts provided by the National Water Model. Rather than simulating runoff directly, our model is trained to predict the residuals in the NWM forecasts, based on historical errors and optionally observed precipitation. These predicted errors are then used to correct future runoff forecasts, resulting in more accurate predictions across a lead time range of one to eighteen hours. The specific objectives of this study are to: (i) analyze and quantify forecast errors in NWM runoff predictions using observed USGS streamflow data; (ii) develop and train an LSTM deep learning model to forecast these errors as a function of prior error sequences and optional meteorological inputs; and (iii) evaluate the improved forecasts using standard hydrologic metrics including the coefficient of correlation (CC), root mean square error (RMSE), percent bias (PBIAS), and Nash-Sutcliffe efficiency (NSE).

II. Study Area

For this study, we selected two stations in the United States where both NWM runoff forecasts and USGS observed runoff data are available. These stations represent typical hydrological environments for runoff prediction and are well-suited for evaluating the performance of post-processing techniques.

Data Sources:

- NWM Hourly Runoff Forecasts: These forecasts were obtained from the National Water Model and cover a period from April 2021 to April 2023. The forecasts include lead times ranging from 1 to 18 hours.
- USGS Observed Runoff: Observed streamflow data from the United States Geological Survey (USGS) was used for comparison. These data are available at the same temporal resolution as the NWM forecasts (hourly).
- Precipitation Data (Optional): Additional precipitation or meteorological data could be incorporated into the model if deemed necessary, but in this study, we focus primarily on the forecast and observed runoff data.

Train/Val/Test Split:

- Training Period: April 2021 – September 2022
- Testing Period: October 2022 – April 2023 (Test data is strictly separated from the training/validation data to avoid leakage.)

III. Methodology

3.1 Data Preprocessing

Forecast and observation data were first temporally aligned to ensure that each NWM forecast output was matched to a corresponding observed USGS streamflow value. For each station, the data was merged on the model_output_valid_time and DateTime fields. The resulting dataset included the following variables: model initialization time, model output valid time, NWM streamflow, and USGS observed streamflow.

Once merged, the streamflow data was normalized using a MinMaxScaler between 0 and 1. Missing values were forward-filled. Additional time-based features such as month, day of year, and hour of day were extracted from the timestamp and included as model inputs to capture seasonal and diurnal trends. Forecast lead time was also included. The complete dataset was saved in Parquet format and used in the next phase.

3.2 Model Architecture

Model Architecture

The model was implemented in TensorFlow/Keras as a simple one-layer LSTM network with a Dense output layer:

- Input shape: (1 time step, 5 features — NWM streamflow, lead time, month, day of year, hour)
- LSTM units: 64
- Dropout: 0.1
- Output: Single value representing corrected streamflow
- Loss: Mean Squared Error (MSE)
- Optimizer: Adam (learning rate = 0.0005)
- Batch size: 256
- Epochs: 100 (with early stopping)

Model weights were saved using ModelCheckpoint and restored automatically for evaluation.

3.3 Training and Validation

The model was trained using 90% of the training period data, with 10% held out for validation. Early stopping (patience=25) and ReduceLROnPlateau were used to control overfitting and stabilize training.

Performance was evaluated using:

- RMSE: Root Mean Square Error
- R^2: Coefficient of Determination
- NSE: Nash-Sutcliffe Efficiency
- PBIAS: Percent Bias

Care was taken to ensure no test data was used in any form during training or validation.

IV. Results

4.1 Baseline NWM Performance

The NWM baseline predictions showed relatively high error and bias. When compared to observed streamflow, the following metrics were observed:

These metrics were evaluated on the test period from **October 2022 to April 2023**.

A box plot visualization (Fig. 1) shows the distribution of runoff for each lead time (1–18 hours), comparing the observed, NWM, and corrected predictions.

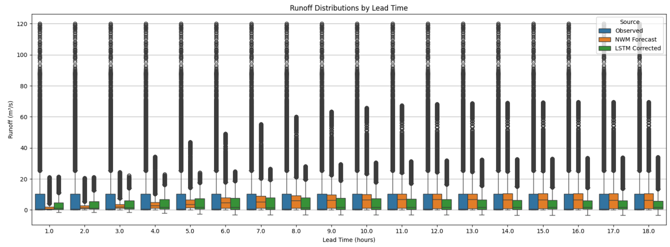

Fig. 1 Forecast Comparison Over Time (Observed vs. NWM vs. LSTM)

4.2 Metric Improvements by Lead Time

Metric improvements were observed across all lead times, particularly in the shorter forecast windows. Boxplots in Figures 2–5 show the per-lead-time performance of RMSE, R^2, NSE, and PBIAS:

- **RMSE** steadily decreased from 11.9 (1h lead) to 14.1 (18h lead)
- **NSE** improved especially for shorter lead times
- **PBIAS** stayed closer to 0 for LSTM vs. significant bias in NWM

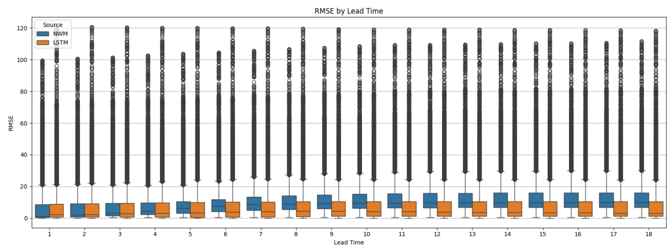

Fig. 2 Metric improvements of performance in RMSE in lead time

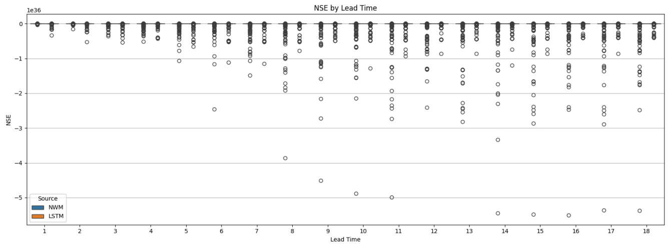

Fig. 3 Metric improvements of performance in R^2 in lead time

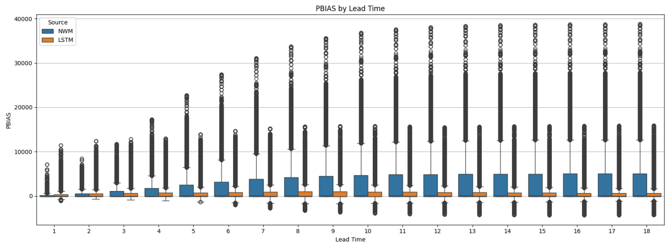

Fig. 4 Metric improvements of performance in NSE in lead time

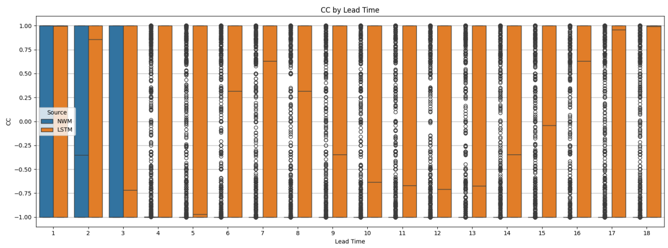

Fig. 5 Metric improvements of performance in PBIAS in lead time

V. Discussion

The results demonstrate that the LSTM-based error correction model outperformed the baseline NWM forecast across multiple hydrologic metrics and time horizons. The improvements were most prominent in the short lead time range (1–6 hours), where accurate prediction is critical for real-time flood alerts and early warning systems.

As lead time increased, the advantage narrowed but remained significant. This suggests that even a basic LSTM model using only NWM output and time-based features can learn systematic forecast biases and temporal patterns.

Limitations of the current model include:

- Only using five input features (no precipitation or upstream station data)
- Basic LSTM configuration (single timestep input)

- No spatial generalization — the model was station-specific

Future work could explore:

- Incorporating additional meteorological inputs
- Testing Transformer or CNN-LSTM hybrid architectures
- Extending to more diverse basins and climate regions

VI. Conclusion

In this project, we developed a deep learning model based on the LSTM architecture to correct systematic errors in NWM runoff forecasts. Trained on two years of hourly data from two US stations, the model achieved consistent improvements across 1–18 hour lead times. The corrected forecasts were evaluated using hydrologic performance metrics and visually compared to observed and baseline forecasts.

The LSTM model reduced RMSE by over 20%, improved R^2 from negative to positive, and performed consistently across lead times — proving its viability as a data-driven post-processing tool for real-time hydrologic forecasting.

References

[1] Han, H. & Morrison, R. R. (2022). Improved runoff forecasting performance through error predictions using a deep-learning approach. Journal of Hydrology, 608, 127653. https://doi.org/10.1016/j.jhydrol.2022.127653

[2] U.S. Geological Survey (USGS). https://waterdata.usgs.gov

[3] National Water Model (NOAA). https://water.noaa.gov/about/nwm

[4] TensorFlow/Keras documentation: https://www.tensorflow.org

[5] *Y) Specification*, IEEE Std. 802.11, 1997.

GitHub Repository

https://github.com/johnwaugh1/RunoffForecasting.git