ECWS-9 2025 Conference

The 9th International Electronic Conference on Water Sciences

11–14 November 2025 | Online

MDPI

Short-Term River Discharge Forecasting Using an XGBoost-Based Regression Model

Sujoy Dey[1*]

[1]Postgraduate Student, Department of Water Resources Engineering, Bangladesh University of Engineering and Technology, Dhaka 1000, Bangladesh.

[*]Author to whom correspondence should be addressed.

INTRODUCTION & AIM

Accurate short-term discharge forecasting is essential in the effective management of river basins, the prevention of floods, and the planning of water resources [1]. Effective river flow condition forecasting has the potential to significantly enhance decision-making in operational hydrology, especially in cases involving prompt responses to extreme weather or changing hydrologic conditions. Some conventional hydrological models have physical process bases, but they are highly data-intensive in terms of calibration and catchment data, which may not be easily accessible or updated. In the past few years, data-driven approaches have surfaced as possible substitutes or complementary alternatives to physically-based models [2]. Machine learning methods, among others, have been shown to possess strong capabilities in mimicking complex, nonlinear input-to-output relations [3]. Gradient boosting algorithms, such as Extreme Gradient Boosting (XGBoost), are well-suited for such purposes based on their high predictive accuracy, immunity to overfitting, and ability to work with varied data types. In this study, the application of a multi-output regression model with XG-Boost for up to six-hour-ahead river discharge prediction is explored. The strategy of the model employs historical discharge data along with engineered temporal features like lagged flows and time-based indicators to extract temporal patterns in flow behavior. The study area is centered on a U.S. Geological Survey (USGS) monitoring station at Hurricane Mills, Tennessee (site ID: 03603000), where a high-resolution time series record of hourly discharge is available. The process pipeline involves in-depth data preprocessing, including imputation of missing values and resampling, as well as systematic feature engineering to derive useful predictors from the discharge time series. A stepwise multi-output strategy is adopted, where each model is trained to forecast each lead time individually. The performance of the models is then evaluated using conventional statistical metrics for establishing their predictive accuracy and reliability.

METHOD

Data Preparation (Preprocess hourly discharge DataFrame)

Step	Feature Engineering
Create Lag Features	For **i = 1** to **N_steps − 1**, generate columns **discharge_lag_i** by shifting the discharge series by **i** periods to represent recent historical discharge values **(e.g., Q_{t-1}, Q_{t-2}, …)**.
Create Short-Term Difference Features	For **i = 0** to **N_steps − 1**, create **discharge_diff_i** columns by computing one-step differences **($Q_t - Q_{t-1}$)** and lagging them to capture short-term rate of change.
Create 24-Hour Difference Features	Similarly, compute 24-hour (daily) discharge differences **($Q_t - Q_{t-24}$)** and lag them to model daily persistence or cyclical patterns.
Add Time-of-Day Feature	Extract the hour of the day (0–23) from the time index and store it as a new feature **(hour_of_day)** to capture diurnal effects.
Generate Future Targets	For **i = 0** to **N_predict − 1**, create target columns **discharge_future_i** by shifting the discharge series backward by **i+1** periods, representing future discharges **(Q_{t+1}, Q_{t+2}, …, Q_{t+6})**.
Remove NaN Rows	Drop all rows containing **NaN** values created by lagging and shifting operations to ensure feature–target alignment.
Split into Features and Targets	Separate the resulting dataset into: Features **(X)** — all predictor columns, and Targets **(Y)** — the future discharge columns.

Data Split
Training Data: From January 1, 2016, to July 1, 2020
Validation Data: From July 1, 2020, to January 1, 2024

Model Training

- **Initialize Model Dictionary:** Create an empty dictionary to store models.
- **Loop Over Forecast Steps:** Iterate through each future time step.
- **Set Model Parameters:** Use XGBoost with 1000 trees and max depth 3.
- **Train Each Model:** Fit each model on training data for each future target.
- **Validation & Early Stopping:** Monitor performance with MAE;
- Stop after 50 rounds without improvement.
- **Save Trained Models:** Store each model in the dictionary by its forecast label.

Feature Importance Visualization → **Prediction and Evaluation** → **Metrics Calculation** Performance metrics (MAE, MSE, RMSE, MAPE, Huber Loss, KGE, NSE, MBE, R^2) → **Visualization** Creating comprehensive metric plots

Figure 1. Methodological flowchart.

STUDY AREA

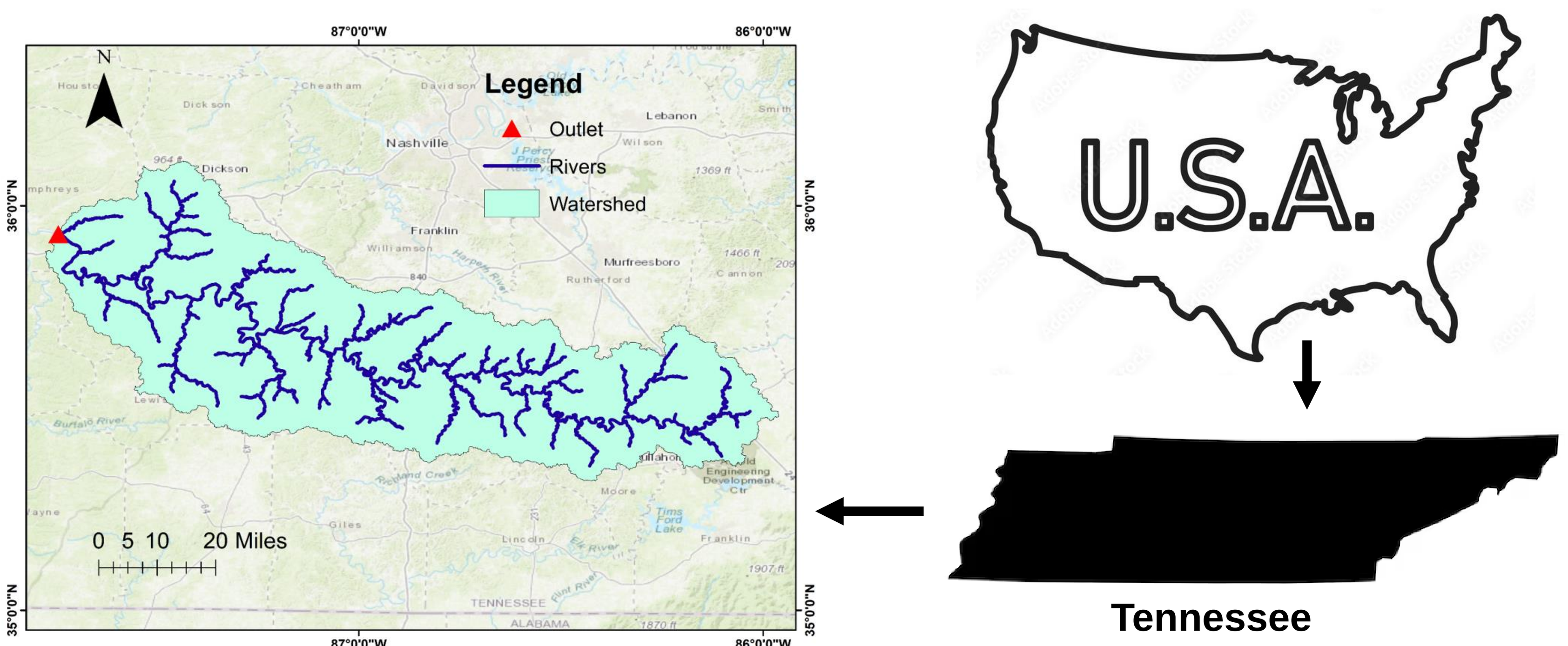

Figure 2. Duck River and Duck River Watershed, Tennessee, USA, with the selected outlet marked in red on the map (USGS 03603000).

RESULTS & DISCUSSION

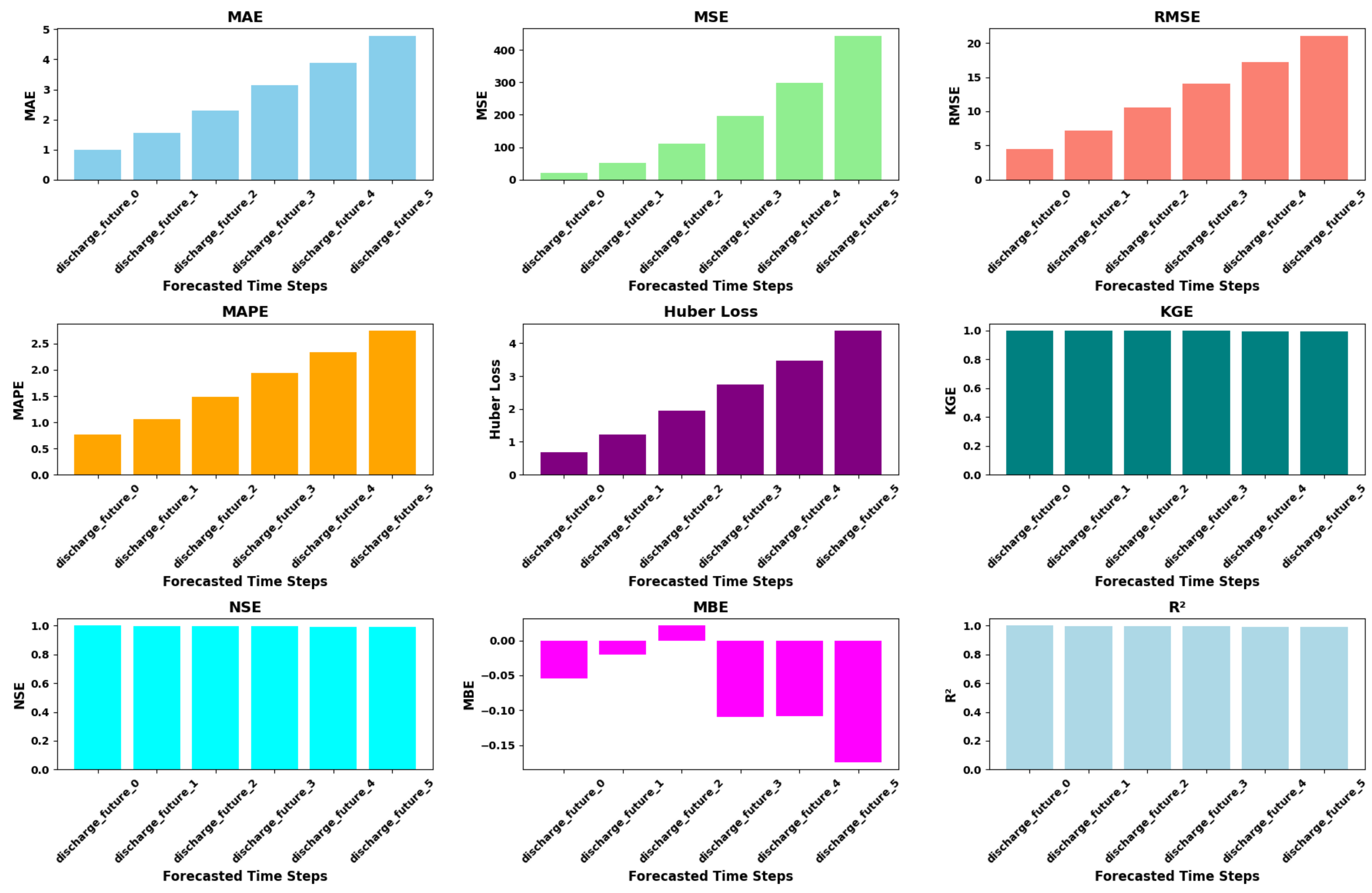

Figure 3. Model performance metrics for multi-hour lead time forecasts (1 to 6 hour lead times).

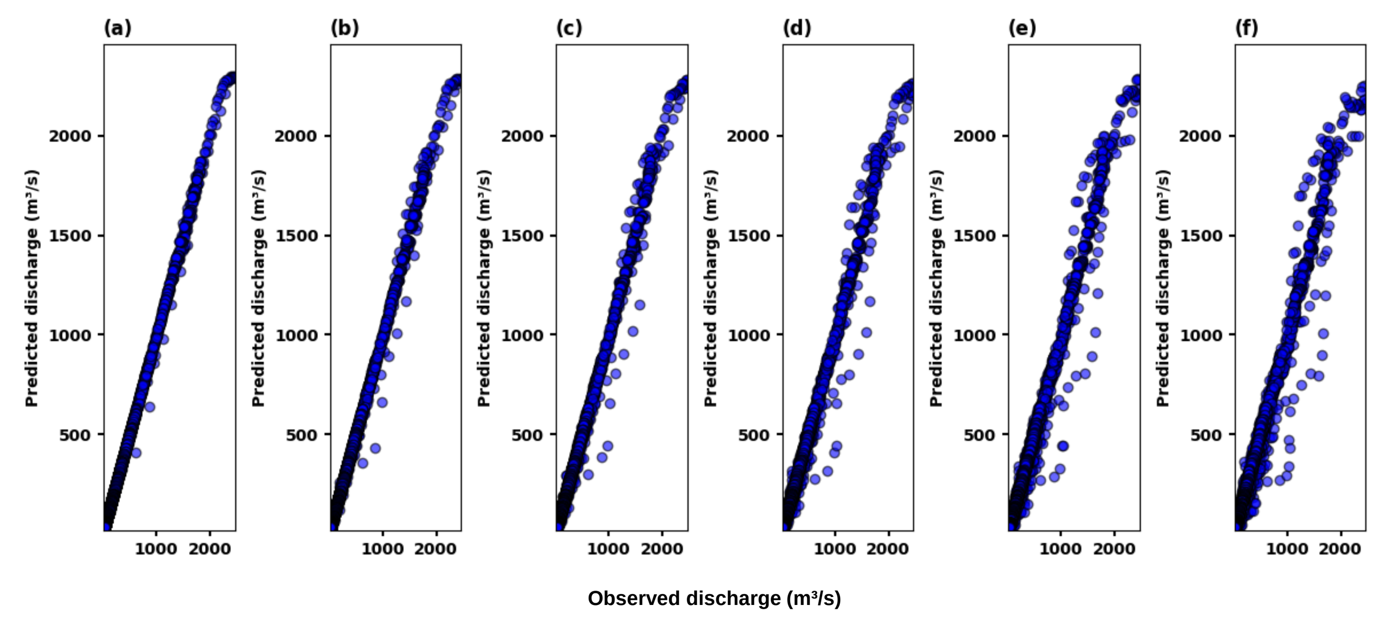

Figure 4. Scatter plots for (a) 1-hour lead time, (b) 2-hour lead time, (c) 3-hour lead time, (d) 4-hour lead time, (e) 5-hour lead time, (f) 6-hour lead time.

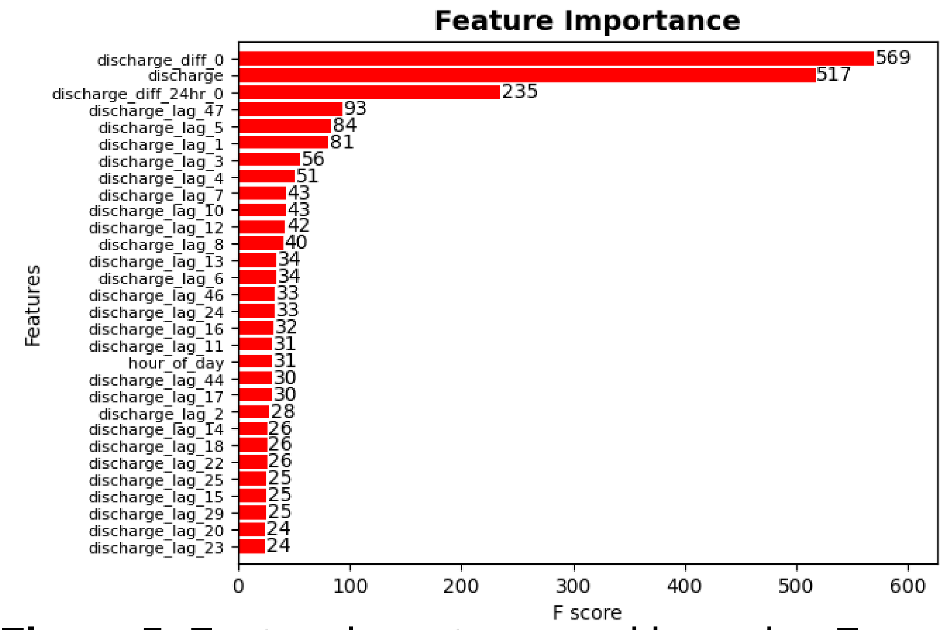

Figure 5. Feature importance ranking using F-score.

The XGBoost-based model achieved high accuracy, with MAE ranging from 0.99 to 4.79 m³/s and RMSE ranging from 4.44 to 21.06 m³/s for lead times of 1–6 hours. MAPE remained below 3% across all horizons, and efficiency metrics (KGE, NSE, R^2) exceeded 0.98, indicating strong agreement with observed discharge. Feature importance analysis highlighted recent discharge lags and short-term differences as key predictors.

CONCLUSION & FUTURE WORK

Future improvements in this study could focus on enhancing feature engineering by incorporating additional time features (e.g., weekday, rolling statistics) and external factors, such as weather conditions or upstream flow. The model's performance can be enhanced through systematic optimization of hyperparameters, ensemble learning methods, and the use of time-series cross-validation for improved generalization. Exploring deep learning methods, such as LSTM, may capture subtle temporal relationships, while probabilistic forecasting methods would offer valuable insights into prediction uncertainty.

REFERENCES

[1] Niazkar, M.; Menapace, A.; Brentan, B.; Piraei, R.; Jimenez, D.; Dhawan, P.; Righetti, M. Applications of XGBoost in Water Resources Engineering: A Systematic Literature Review (Dec 2018–May 2023). Environmental Modelling & Software 2024, 174, 105971, doi:10.1016/j.envsoft.2024.105971.

[2] Ni, L.; Wang, D.; Wu, J.; Wang, Y.; Tao, Y.; Zhang, J.; Liu, J. Streamflow Forecasting Using Extreme Gradient Boosting Model Coupled with Gaussian Mixture Model. Journal of Hydrology 2020, 586, 124901, doi:10.1016/j.jhydrol.2020.124901.

[3] Sanders, W.; Li, D.; Li, W.; Fang, Z. Data-Driven Flood Alert System (FAS) Using Extreme Gradient Boosting (XGBoost) to Forecast Flood Stages. Water 2022, 14, 747, doi:10.3390/w14050747.