Response to Comments of Reviewer 2

This manuscript presents a comprehensive large-sample study evaluating the impact of LSTM-based data integration (DI-LSTM) on streamflow simulation across hundreds of basins in the Western U.S., using both streamflow (Q) and snow water equivalent (SWE) as auxiliary inputs. The study is motivated by the operational challenges of hydrological forecasting in arid and snow-dominated regions and aims to improve short- and long- term forecasting using deep learning techniques. The authors highlight the advantages of DI over traditional data assimilation (DA) and provide an extensive experimental comparison across multiple timescales and input configurations. My detailed comments are as follows:

Major Comments

1. The manuscript title references "implications for forecasting in the Western U.S.," yet the experimental setup focuses solely on hindcasting using future observations (i.e., perfect knowledge of lagged Q or SWE). It would be better if the authors could clarify what specific implications for real-world forecasting are supported by their results, and how the proposed DI-LSTM might be adapted for settings where future information is unavailable or uncertain.

Response: We thank the reviewer for this insightful comment. We acknowledge that the original use of the term "implications" in the title may have been misleading, as the study does not directly demonstrate operational forecasting applications. To more accurately reflect the scope of the work, we have revised the title by replacing "implications" with "potential", emphasizing that this study explores the benefits of integrating lagged observations in retrospective experiments and exhibit the potential applicability of this approach in real-world forecasting contexts. This potential is discussed in Section 4.3 of the manuscript.

2. There is a risk that DI-LSTM overfits to future data, especially when lagged target variables (Q or SWE) are incorporated directly from observed time series. It would be better if the authors could clarify:

- Whether the lagged variables are drawn from observations or predicted recursively;
- How these variables are embedded into the model;
- And whether any form of future leakage occurs during training or evaluation.
- It would also be helpful if the authors could provide a clear schematic of the DI-LSTM architecture to illustrate how lagged information is integrated into the model.

Response: We recognized that the original Equations (1) and (2) might have been unclear or potentially misleading. We have revised them to explicitly show the inputs of LSTM and DI-LSTM. We have also added a new subplot (Figure 1c) to illustrate how lagged information is integrated into the DI-LSTM model. As shown in the revised Equation (2) and Figure 1c, the inputs of DI-LSTM at each time step include forcings and basin attributes at the current time step, along with observations lagged by N time steps before the current time step. For instance, to simulate streamflow at time t, the model directly receives the forcings and basin attributes at time

t, as well as lagged observations from time *t*-*N*. These lagged variables are directly from observations, appended to the original LSTM inputs, and processed using the same preprocessing procedures described in Appendix C. Therefore, at each time step *t*, DI-LSTM will only see historical observations with *N* lagged time steps. The model will iterate over time steps and output streamflow estimations at each time step. There is no future leakage during training or validation. Specifically,

- The lagged variables were drawn from observations and no recursively predicted values were used: "where *N* is the lag time step, and y^{t-N} is *N*-step lagged Q or SWE directly from observations"

- We have added a section in the Appendix C to describe data pre-processing procedures for LSTM and to show how all the lagged variables are embedded into the model.

"Standard pre-processing techniques, including normalization and standardization, were applied to ensure compatibility across different input types and to facilitate effective parameter optimization (See Appendix C for details). Lagged observations were directly appended to the original LSTM inputs and underwent the same preprocessing procedures."

"Appendix C: Data preprocessing for LSTM and DI-LSTM

During the iterations of the training process, basins from the entire dataset were randomly sampled to form a mini-batch each time to calculate the loss function. This batching method typically assumes that model errors are identically distributed among basins within the same mini-batch. Without data preprocessing or normalization, the loss function would inherently pay more attention to wetter and larger basins compared to drier or smaller basins. To prevent this imbalance, we applied standard pre-processing techniques, including normalization and standardization, following Feng et al. (2020).

First, we normalized the daily discharge by basin area and mean daily precipitation to obtain a dimensionless discharge value as the target variable.

Then we transformed the distributions of daily discharge and precipitation as close to Gaussian as possible, since these two typically have Gamma distributions, using the equation:

$$v^* = log_{10}(\sqrt{v} + 0.1) \quad \text{(A9)}$$

where v and v^* are the variables before and after transformation, respectively. 0.1 is added inside the log to avoid making the log of zero. Transforming the data to a Gaussian distribution enhances the stability and efficiency of gradient-based optimization methods in LSTM. Additionally, it reduces the impact of extreme peak values during model training, improving the model's representation of low-flow conditions.

Finally, standardization was applied to all input features (forcings, static basin attributes and lagged observations), as well as the output (discharge) by subtracting the mean value and then dividing by the standard deviation of training-period data."

- The revised Equation (1) and (2) are provided below to more clearly illustrate the inputs

of LSTM and DI-LSTM. For DI-LSTM, inputs at each time step consist of forcings and basin attributes at the current time step, along with N-step lagged historical observations. As the model relies solely on information available up to the current time step, no future data is included in the input data, ensuring that the framework is free from any form of future data leakage.

"Overall, we trained two types of LSTM models to assess the potential of leveraging lagged observations to improve streamflow estimation (Fig. 1b). The first type is a standard LSTM model that does not perform data integration (DI) and does not use any historical Q or SWE observations. It serves as a valuable benchmark for the comparison against DI-LSTM model. The inputs consist solely of forcings and basin attributes at the current time step and can be expressed as:

$$\boldsymbol{I}^t = [\boldsymbol{x}_0^t,\ A], \tag{1}$$

Where t is the current time step, $\boldsymbol{I^t}$ reprensents the raw input to the model (before data pre-processing), $\boldsymbol{x}_0^t$ stands for dynamic forcings, and A represents static basin attributes.

The second type of model is DI-LSTM, which refers to the incorporation of lagged observations (y) into the model (Fig. 1c). The inputs of DI-LSTM can be expressed as:

$$\boldsymbol{I}^t = [\boldsymbol{x}_0^t,\ A,\ y^{t-N}], \tag{2}$$

where N is the lag time step, and y^{t-N} is N-step lagged Q or SWE directly from observations."

- We have added a subplot (Figure 1c) to illustrate how the DI-LSTM model works with data integration of N-step lagged observation, which is shown below.

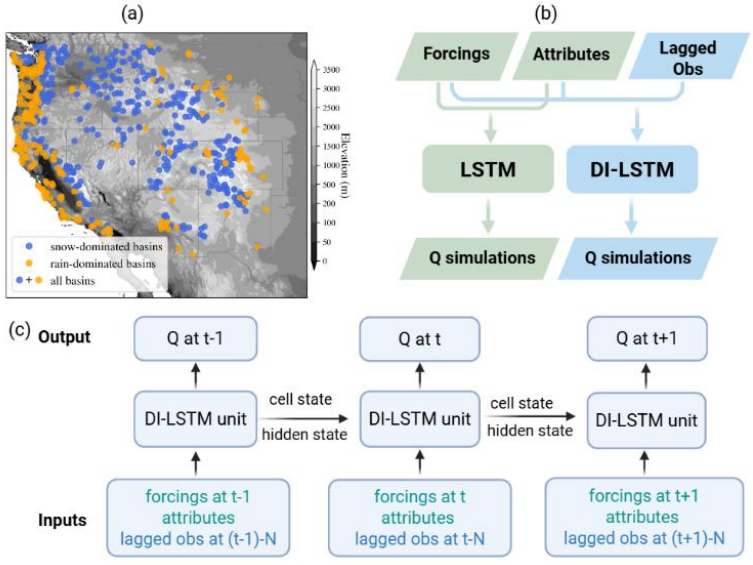

Figure 1. (a) Study basins: blue dots stand for snow-dominated basins, orange dots stand for rain-dominated basins. (b) models: LSTM vs. DI-LSTM model. (c) DI-LSTM with data integration of N-step lagged observations"

Minor Comments

1. Line 138: The typographic dash in DI-LSTM in the formula appears to be a mathematical minus sign. Please correct this to ensure clarity.

Response: It has been fixed in the revised manuscript.

2. The choice of using a 10-day lag for Q and a 6-month lag for SWE is not clearly justified. It would be better if the authors could explain the rationale behind these specific durations, either based on hydrological reasoning or exploratory experiments.

Response: The 10-day lag for daily scale and 6-month lag for monthly scale were selected to align with the practical considerations of operational forecasting. In general, short-term operational forecasts focus on lead time within 10 days, beyond which the uncertainty in forecasted forcings increases substantially, often resulting in streamflow forecasts that are of limited practical value. At the monthly scale, forecasting horizons ranging from 1 to 6 months are commonly used to inform broader water resource planning and management decision. We have added this clarification to the manuscript, as shown below.

"For the daily scale, lag times ranged from 1 to 10 days were considered, aligning with the focus of short-term operational forecasts, which typically target lead times within 10 days due to rapidly increasing uncertainty beyond this range. For the monthly scale, 1- to 6-month lags were chosen to reflect typical forecasting horizons used in broader water resource planning and management."

3. It would be better if the authors could discuss more thoroughly the phenomenon shown in Figure 10(a), particularly the performance degradation at 4–7 day lags in some snow-dominated basins.

Response: In the daily DI(SWE) experiments, we did not observe a consistent trend of performance improvement or degradation across different lag times. The fluctuations in KGE appear to be random rather than indicative of a meaningful benefit signal. We therefore interpret these variations as noise rather than evidence of (in)effective data integration.

4. Sensitivity to Random Initialization and Training Variability. It would be better if the authors could report how diverse the six randomly seeded training runs are. This would help clarify whether the models are sensitive to random initialization or the stochastic training process. Reporting variability across seeds would improve the robustness and reproducibility of the findings.

Response: Thank you for this suggestion. We have added a figure in the Appendix comparing the performance of the ensemble mean and individual random seed simulations for daily LSTM, DI(Q-1) and DI(SWE-1), respectively. The results highlight that randomness in the training process introduces some variability, and the ensemble mean provides a more reliable basis for model evaluation.

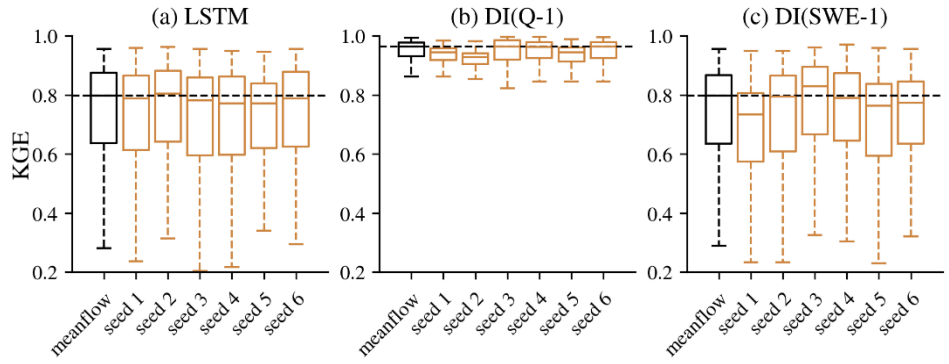

Figure E1. Performance comparison between the ensemble mean and individual random seed simulations across different experiments at the daily scale: (a) LSTM, (b) DI(Q-1), and (c) DI(SWE-1). "meanflow" refers to the ensemble mean derived from six simulations, while "seed 1" through "seed 6" represent the results from individual random seeds.

5. While Table C2 provides hyperparameters for model training, it would be better if the authors could briefly justify their selection or indicate whether any tuning or sensitivity analysis was performed. This would help assess the robustness of the model configuration and whether the selected architecture is optimal across diverse basin types.

Response: Thank you for the comment. We have added a brief description of the hyperparameter selection process to the manuscript, as shown below.

"Hyperparameters, such as the number of hidden/cell states and the length of the input sequence, were determined separately for daily and monthly scales. For the daily scale, hyperparameter combinations were inherited from our previous studies (Feng et al., 2020, Song et al., 2024, Yang et al., 2025). For the monthly scale, hyperparameters were determined through a simple grid search across a predefined range of values (Table E2). Final selections were based on analysis of training and validation RMSE learning curves, with the chosen settings minimizing validation RMSE while avoiding overfitting."

Table E2. Hyperparameters for the LSTM or DI-LSTM model

Hyperparameter	Daily Scale	Monthly Scale	
	Best value	Grid search	Best value
Length of training instances	365	12, 24, 36, 48	48
Mini-batching size	100	50, 100, 150, 200	50
LSTM dropout rate	0.5	0, 0.2, 0.5	0.5
LSTM hidden size	256	128, 256	256
Number of training epochs	300	[100, 600]	300
Number of stacked LSTM layer	1	1	1

Feng, D., Fang, K., and Shen, C.: Enhancing streamflow forecast and extracting insights using long-short term memory networks with data integration at continental scales, *Water Resour. Res.*, 56, e2019WR026793, https://doi.org/10.1029/2019WR026793, 2020.

Song, Y., Tsai, W.-P., Gluck, J., Rhoades, A., Zarzycki, C., McCrary, R., Lawson, K., and Shen, C.: LSTM-Based Data Integration to Improve Snow Water Equivalent Prediction and Diagnose Error Sources, *J. Hydrometeorol.*, 25, 223–237, https://doi.org/10.1175/JHM-D-22-0220.1, 2024.

Yang, Y., Feng, D., Beck, H. E., Hu, W., Abbas, A., Sengupta, A., Delle Monache, L., Hartman, R., Lin, P., Shen, C., and Pan, M.: Global Daily Discharge Estimation Based on Grid Long Short-Term Memory (LSTM) Model and River Routing, *Water Resour. Res.*, 61, e2024WR039764, https://doi.org/10.1029/2024WR039764, 2025.