# *Hydra-LSTM: a semi-shared machine learning architecture for prediction across watersheds*

# Hydra-LSTM: A Semi-Shared Machine Learning Architecture for Prediction across Watersheds

KARAN RUPARELL,[a,b] ROBERT J. MARKS,[a,b] ANDY WOOD,[c] KIERAN M. R. HUNT,[a,d] HANNAH L. CLOKE,[a,e] CHRISTEL PRUDHOMME,[b] FLORIAN PAPPENBERGER,[b] AND MATTHEW CHANTRY[b]

[a] *Department of Meteorology, University of Reading, Reading, United Kingdom*
[b] *ECMWF, Reading, United Kingdom*
[c] *National Center for Atmospheric Research, Boulder, Colorado*
[d] *National Centre for Atmospheric Science, Reading, United Kingdom*
[e] *Department of Geography and Environmental Science, University of Reading, Reading, United Kingdom*

(Manuscript received 26 October 2024, in final form 12 March 2025, accepted 10 June 2025)

ABSTRACT: Long short-term memory (LSTM) networks are used to build single models that predict river discharge across many catchments. These models offer greater accuracy than models trained on each catchment independently, if the same variables are used as inputs for each catchment. However, the same data are rarely available for all catchments. This prevents the use of variables available only in some catchments, such as historic river discharge or upstream discharge. The only existing method that allows for optional variables requires all variables to be in the initial training of the model, limiting its transferability to new catchments. To address this limitation, we develop the *Hydra*-LSTM. The *Hydra*-LSTM is able to use some variables across all catchments to make predictions and use further variables in other catchments where they are helpful and available. This allows general training and the use of catchment-specific data. The bulk of the model can be shared across catchments, maintaining the benefits of multicatchment models to generalize while also benefiting from using bespoke data. We apply this methodology to 2-day-ahead river discharge prediction in the western United States, a small enough time step to expect our models to be skillful and difficult enough to expect differences between models. We obtain more accurate quantile predictions than multicatchment and single-catchment LSTMs while allowing forecasters to introduce and remove variables from their prediction set. We test the ability of the *Hydra*-LSTM to incorporate catchment-specific data, introducing historical river discharge as a catchment-specific input, outperforming other commonly used models.

KEYWORDS: Hydrology; Probability forecasts/models/distribution; Short-range prediction; Hydrologic models; Artificial intelligence

## 1. The need for hydrological models dealing with variable input data

Accurate river discharge forecasts are vital for catchment managers to decide how much water to extract for agricultural usage, the production of hydroelectricity, or any actions to take to maintain the river's biodiversity. The needs of catchment managers and the relevant variables for forecasts can vary greatly between catchments (Li and Razavi 2024; Clerc-Schwarzenbach et al. 2024), making it vital that models are readily adaptable and expandable in a low-cost way to take advantage of local domain expertise (Fleming and Goodbody 2019; Fleming et al. 2021). In the current state of machine learning for river discharge forecasting, if a variable is essential in a catchment but not in the list of variables the model was initially trained for, it cannot be used without retraining the entire model. This creates a conflict in the training of machine learning models, as individual models perform better when trained across a range of catchments (Kratzert et al. 2019, 2024).

This forces forecasters into a choice: Should they choose a model that lets them benefit from knowledge gained from other catchments, or should they choose a model tailored to their catchment? The latter allows them to include information like historical river discharge, local forecast predictions, additional soil measurements, upstream observations, and other variables that are important in many catchments but only available in some. This underpins the need for a model that can take advantage of increased exposure to catchment variability while preserving the importance of bespoke catchment knowledge.

Numerous strides have been made using machine learning in river discharge prediction, addressing problems with inputs existing on multiple time scales (Gauch et al. 2021), limits of predictability in discharge prediction (Liu et al. 2024), or physics-guided machine learning models (Xie et al. 2021). Work has also focused on transferring models to catchments with no or very few river discharge observations (Yoon and Ahn 2024; Fang et al. 2022). However, there has so far been only one attempt to train a model with different data inputs between locations, which we believe is an important step toward uncompromised predictions at a global scale.

This method, proposed by Nearing et al. (2023), allows variables to be input into a single general model alongside a

Denotes content that is immediately available upon publication as open access.

*Corresponding author*: Karan Ruparell, k.ruparell2@pgr.reading.ac.uk

DOI: 10.1175/AIES-D-24-0103.1 e240103

| Architecture | Universally available data (ERA5 Reanalysis etc.) | Predefined catchment specific data (river discharge, soil type etc.) | Usability across catchments | Can add new variables (river temperature, upstream gauge information etc.) |
|---|---|---|---|---|
| Single-Catchment LSTMs | ✓ | ✓ | ✗ | ✗ |
| Multi-Catchment LSTM Without River Discharge | ✓ | ✗ | ✓ | ✗ |
| Multi-Catchment LSTM With River Discharge | ✓ | ✗ | ○ | ✗ |
| Flag LSTM | ✓ | ✓ | ✓ | ✗ |
| Hydra-LSTM | ✓ | ✓ | ✓ | ✓ |

FIG. 1. Comparison of different LSTMs for hydrological modeling regarding their data requirements, usability across catchments, and flexibility in adding new variables. The architectures include single-catchment LSTMs, multicatchment LSTM without river discharge, multicatchment LSTM with river discharge, Flag LSTM, and *Hydra*-LSTM. The checkmarks indicate the presence of a feature, the crosses indicate the absence of a feature, and the circles indicate partial usability.

corresponding flag variable that is 1 if the variable is available and 0 if not. Missing variables are replaced by some default value, with a 0 in the corresponding flag variable. This approach allows the same model to be used on different catchments, even if they have other data available. However, it is unusable for variables not already defined in the parameter set or for variables particular to a single catchment, such as information from an upstream gauge.

Here, we design a model that can combine the ability to train on many catchments with the flexibility to use whatever relevant hydrological information is available. We call this model the *Hydra*-long short-term memory (LSTM) (Fig. 2). The *Hydra*-LSTM uses an initial encoding LSTM, called the *Hydra* Body, to use variables available across all catchments, for example, globally available reanalysis datasets such as ERA5 (Hersbach et al. 2020) The *Hydra* Body transforms them into a set of variables that are more directly useful to river discharge prediction. The *Hydra* Body outputs are then combined with one of the *Hydra* Heads, each an LSTM itself, being passed to either a Multi-Catchment Head trained across all catchments or a Single-Catchment Head trained only for the corresponding catchment that is capable of accepting catchment-specific variables. The Single-Catchment Head can take the outputs of the Body alongside a time series of recent river discharge observations from that river gauge, and an upstream river gauge that the forecaster has decided is particularly

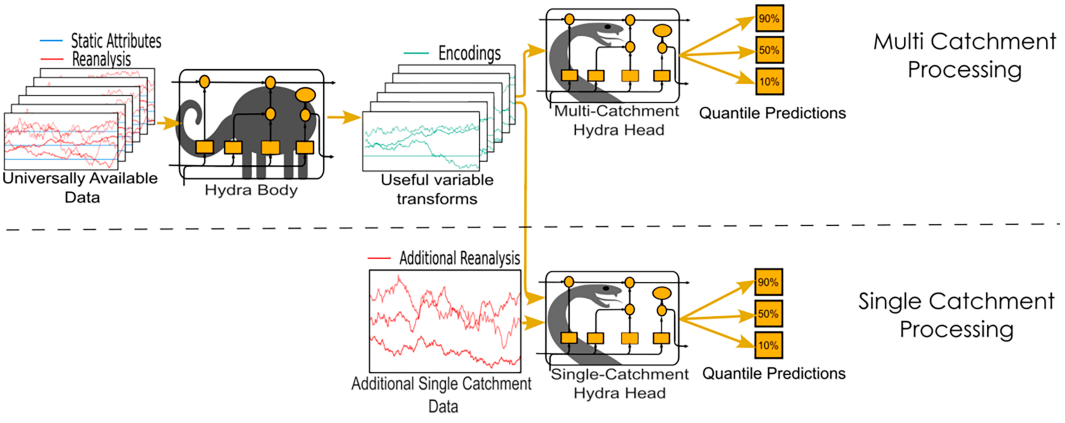


FIG. 2. Diagram of the *Hydra* Model Architecture. The leftmost box plots the time series data available at all catchments, including historical data, forecast data, and static catchment attributes. An encoding LSTM processes these, dubbed the *Hydra* Body, which produces a lower dimensional encoding of the information. If no further data are available, this encoding is passed to the Multi-Catchment Head, an LSTM that transforms the encoding to quantile discharge predictions. However, if further information is available for that catchment, it is passed to a Single-Catchment Head alongside the additional time series data, which are then combined to produce quantile discharge predictions. All model blocks, the *Hydra* Body and *Hydra* Heads, are LSTMs, whose parameters are different and learned in training.

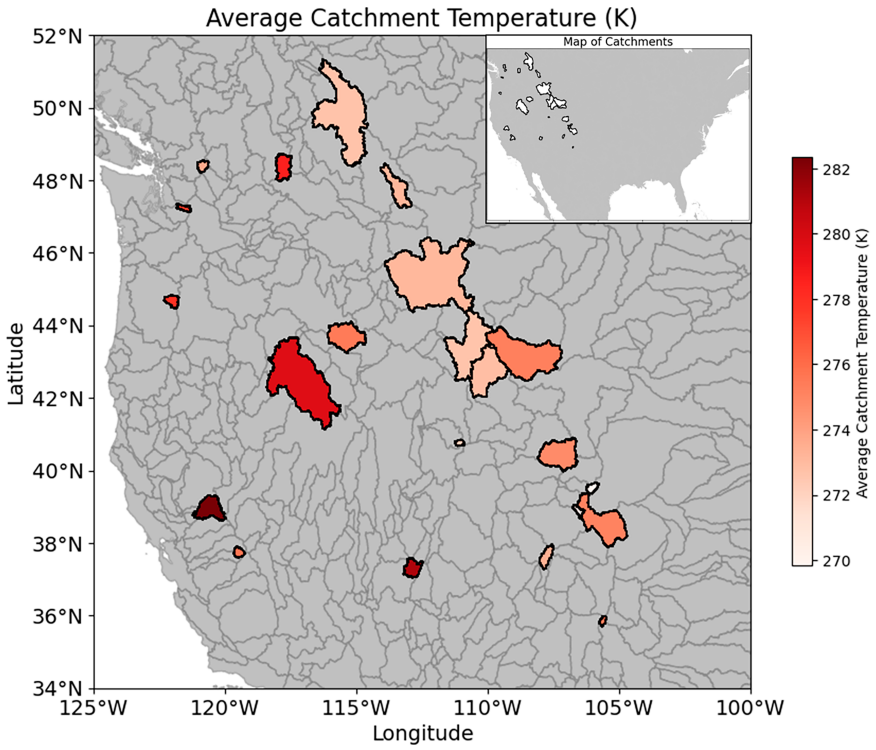


FIG. 3. Plot of catchment sites evaluated in this study, and the mean annual temperature over each catchment in Kelvin. All catchments are located in the western United States, and the plot shows the western United States with an inset showing a map of United States as a whole.

influential to the river flow at this river gauge. The Single-Catchment Head would then predict the 2-day-ahead river discharge at that river gauge, which has been tuned specifically to that river gauge. Conversely, the Multi-Catchment Head would only take the outputs of the *Hydra* Body to make its quantile predictions for 2-day-ahead river discharge. The differences between the models discussed is summarized in Fig. 1, and the range of catchments used in our study can be found in Fig. 3.

There are two requirements for the *Hydra*-LSTM to be beneficial. First, when the same data are available, it should perform just as well as other state-of-the-art methods. The main benefit of multicatchment models, such as the *Hydra*-LSTM, is their ability to be used out of the box, so a forecaster should be able to use the *Hydra*-LSTM in this setting without having to pay a penalty compared to the skill they could have achieved if they used another model. Second, the *Hydra*-LSTM should benefit from including catchment-specific data in an additional Single-Catchment Head. The unique advantage of the *Hydra*-LSTM is in its ability to include additional data not considered in the design of the *Hydra* Body and Multi-Catchment Head, and so for the *Hydra*-LSTM to be beneficial, the Single-Catchment Head should be a useful means of adding additional data in a way that improves the model's performance.

## 2. Experimental design

We test the *Hydra*-LSTM for 1-day-ahead river discharge prediction, as this is the foundation for prediction over longer lead times, and a number of other LSTM-based models have shown the potential of machine learning models at this lead time (Kratzert et al. 2018; Hunt et al. 2022; Nearing et al. 2023). We focus on creating forecasts of the 10%, 50%, and 90% quantile thresholds of 2-day-ahead river discharge, as it is useful for water resource managers to understand the range of uncertainty in predictions (Wang et al. 2023). Quantile estimation has been successfully applied in hydrological forecasting previously (Jahangir et al. 2023; Koenker 2005). Other papers have attempted to predict the continuous distribution of uncertainty in river discharge (Klotz et al. 2022). However, this requires assumptions to be made on the shape of discharge uncertainty, and so instead, we opt to predict the quantiles directly.

We perform two experiments. In the first experiment, we evaluate the *Hydra*-LSTM Body and Multi-Catchment Head in a setting where historical river discharge is not operationally available and compare it to other state-of-the-art methods, described in section 3. In this experiment, none of the models are given access to historical river discharge as a prediction input; however, it is used for training the model. In the second experiment, we provide all models with river discharge as an input. We provide river discharge into the *Hydra*-LSTM as an additional data source in Single-Catchment Heads for each catchment. If historic river discharge added this way can be used just as well as if it were fed as an input in the multi-catchment setting, then we expect the performance of the *Hydra*-LSTM to be at least as good as the performance of the

benchmarks. This would mean that we can add additional data to our *Hydra*-LSTM without retraining the entire model. This is not true for existing LSTMs used in river discharge prediction, where adding additional data to any of the machine learning benchmarks would require retraining us to retrain entire multicatchment model with that variable present.

*Performance benchmarks*

We compare our *Hydra*-LSTM with four different machine learning approaches already used for predicting daily river discharge, all LSTMs (Kratzert et al. 2018, 2019; Nearing et al. 2023). These models are the current best approaches to use a machine learning model for river discharge prediction in catchments and so are alternative models that might otherwise be used by a forecaster that could use the *Hydra*-LSTM.

The first of these modeling approaches is to use single-catchment LSTMs, training separate LSTMs for each catchment. The architecture is replicated as described in Kratzert et al. (2018). This allows all available data in each catchment to be used in training but means that each model is not trained on more than one catchment. This means it is less exposed to extremes, which can decrease its ability to extrapolate. This requires individual forecasters to set up the model from scratch, adding additional complexity in usability and means that the model has less training data to use, potentially decreasing performance and generalizability to different hydrological conditions. This is an approach that is likely to be used when a forecaster has some specific features they would wish to use in their model that is not present in any out-of-the-box multicatchment model and has enough data in their own catchment to train a suitable model. We are interested in seeing whether or not the Single-Catchment Head of the *Hydra*-LSTM can offer results comparable to or even outperform these single-catchment LSTMs.

Second, we train two multicatchment LSTMs, one with river discharge as a predictor and one without. The implementation follows the method described in Kratzert et al. (2019). These LSTMs are trained across all catchments in our dataset. The decision whether or not to use the multicatchment LSTM and whether to use it with or without river discharge usually depends on whether or not you have river discharge available operationally as an input, as many catchments will not. Multicatchment LSTM setups do not allow for any differences in data availability between uses, either always requiring a variable or having no way of using it. It has been shown that with enough data across catchments to train on, these models can outperform single-catchment LSTMs and are especially useful in cases when the gauge record is too short to train a single-catchment LSTM. They are also useful when no additional data would allow the forecaster to benefit from a tailored approach. Here, we are interested in whether or not the *Hydra*-LSTM can perform similarly to the multicatchment LSTM trained without river discharge when it also does not use river discharge, i.e., through the Multi-Catchment Head. We also compare the performance of the Single-Catchment Head, which has river discharge as an additional input, with that of the multicatchment LSTM, which has river discharge as an input. If it performs similarly in the setting where river discharge is available, then we will have shown that there is no relevant loss in skill by providing additional information through an additional head instead of retraining an entirely separate multicatchment LSTM with river discharge as an input. This would offer more flexibility in variable choice, allowing forecasters to train Single-Catchment Heads onto the *Hydra*-LSTM using just the data at their catchment instead of having to train a multicatchment model.

The final model we compare against can tackle data disparity between catchments, which we call a Flag LSTM and is a method used in Nearing et al. (2023). This model has different variables as potential inputs and, for each of these variables, a corresponding flag time series denoting whether the variable is available for a given day. If the data are missing, the flag time series contains a zero; otherwise, it contains a one. The corresponding variable time series has some placeholder numbers for days when they are not available. In our tests, river discharge is the additional variable that may be unavailable. This model is able to make predictions both when river discharge is and is not available, but it does not allow for any additional variables to be introduced after the initial design of the model without retraining the entire model. Our *Hydra*-LSTM, on the other hand, has a Single-Catchment Head that allows additional variables to be introduced by training only a smaller new section of the model, and so it is far easier to introduce new variables. We train the Flag LSTM, using a flag to specify whether or not river discharge is available at a catchment. In training, 50% of training examples are without river discharge. When comparing with this model, we wish to see how the method of adding additional data through an additional model head compares to using a binary flag to introduce potentially missing data. Even though the *Hydra*-LSTM is more flexible, allowing additional data to be used even when it is not considered at the model development stage, the additional flexibility the Flag LSTM provides when compared to the other benchmarks may be satisfactory in many cases if it performs significantly better than the *Hydra*-LSTM.

## 3. Hydra-LSTM architecture

The *Hydra* Model consists primarily of three blocks: the *Hydra* Body, the Multi-Catchment Head, and a Single-Catchment Head for each catchment wishing to incorporate additional data. In our experiments, each of these model blocks are LSTMs. LSTMs are a form of recurrent neural network and often used when making predictions with temporal inputs. However, this general architecture could be adapted to use an array of different architectures. As our implementation of the *Hydra* Model comprises LSTMs, it takes in data as a vector of time series. Static catchment variables, such as soil type or rock type, are then passed to a model as static time series.

The *Hydra* Body takes as inputs time series data that are available across all catchments, such as ERA5 reanalysis precipitation or catchment area, and transforms these into smaller and more informative time series of summary variables, known as encodings. These encodings are transformations of the initial variables learned by the *Hydra* Body in

training. If a water resource manager does not wish to use a bespoke Single-Catchment Head for their catchment, the time series of encodings is passed as inputs to the Multi-Catchment Head. The Multi-Catchment Head then returns a set of three predictions, predicting the 10%, 50%, and 90% quantile thresholds of the 2-day-ahead river discharge.

If a water resource manager does wish to use a Single-Catchment Head, however, then the outputted time series from the *Hydra* Body can be concatenated with additional catchment data and then be passed onto their catchment Single-Catchment Head. The Single-Catchment Head then outputs predictions of the 10%, 50%, and 90% quantile thresholds of the 2-day-ahead river discharge, just as the Multi-Catchment Head would. In our experiments, we assume that river discharge is not always operationally available and so does not include it as inputs into the *Hydra* Body. We also train a different Single-Catchment Head for each catchment, with river discharge as an additional input that is concatenated to the outputs of the *Hydra* Body. River discharge is important for predicting future river discharge. However, in many catchments, river discharge observations do not exist or are only available with a large time delay. We further analyze prediction quality at the Green River below Howard A Hanson Dam gauge.

## 4. Data

The Water Supply Forecast Rodeo was a competition held by the U.S. Bureau of Reclamation in 2023 focused on predicting river discharge across 27 catchments in the western United States for primarily agricultural and hydro-electrical purposes (DrivenData 2024). Because this work focuses on forecasting daily river discharge, we focus on 18 of those catchments for which daily river discharge observations are available. These catchments cover a wide range of different climatological conditions (Table 1) and static physical characteristics (Table 2).

Historical observations from the U.S. Geological Survey and reanalysis data from the fifth generation ECMWF atmospheric reanalysis land (ERA5-Land) are used as input to the models with 25 variables used in total Hersbach et al. (2020) (Table 1). Daily river discharge observations are extracted from the U.S. Geological Survey (2024). ERA5-Land forcing data are taken for the study region from October to July from 2001 to 2023 in hourly and 6-hourly intervals at a spatial resolution of 9 km. Historical daily river discharge observations were only used in some catchments to test how introducing river discharge in a Single-Catchment Head in the *Hydra*-LSTM compares to introducing it in the benchmark models. Catchment attributes are taken as static for the purpose of this study and are taken from the BasinAtlas dataset (Linke et al. 2019).

## 5. Training

To train a single model to create predictions, $\hat{y}$ of different quantile thresholds $\tau$ of 2-day-ahead river discharge, we need a loss function that is minimized when the model predicts the correct thresholds. The quantile loss, Eq. (1), is an ideal function for this Koenker (2005). A generalization of the mean absolute error (MAE), the $\tau$th quantile loss $L_\tau$, can be proven to be minimized only by predicting the $\tau$th quantile of the distribution associated with the predictand, conditional on the available predictors. This means that applying the 10% quantile loss to the model's prediction for the 10% quantile threshold $\hat{y}_{10}$ results in the loss being minimized if and only if it consistently predicts the value of river discharge $y_t$ that has a 10% chance of being exceeded given the data available to the model. The loss is shown in Eq. (1), where $y$ is the observed river discharge for the day the model is trying to make a prediction for and the $\hat{y}$ is the models' corresponding prediction:

TABLE 1. Summary statistics of each variable used in training and their range across catchments. (a) Time series variables, averaged along each catchment. The table shows the minimum, median, and maximum catchment averages of various forcing variables. All variables other than river discharge are derived from ERA5-Land (Hersbach et al. 2020). River discharge is derived from the USGS (2024). Catchment boundaries were provided by the Bureau of Reclamation (DrivenData 2024). (b) Static catchment attributes. The table shows the minimum, median, and maximum values found across the catchments used in this study, averaged along each catchment when necessary. Gauge elevation and area are provided by the U.S. Bureau of Reclamation (DrivenData 2024), and all other variables are derived from the BasinAtlas dataset (Linke et al. 2019).

| Variable | Minimum | Median | Maximum |
|---|---|---|---|
| (a) | | | |
| Precipitation (mm $yr^{-1}$) | 375.95 | 781.1 | 1708 |
| Evaporation (mm $yr^{-1}$) | 317.55 | 438 | 547.5 |
| 2-m mean annual temperature (°C) | −0.85 | 4.25 | 11.35 |
| Snow depth water equivalent (m) | 0.0127 | 0.0891 | 0.573 |
| Soil water volume ($m^3$) | 0.14 | 0.30 | 0.35 |
| River discharge ($m^3$ $s^{-1}$) | 0.0955 | 1.3696 | 21.372 |
| 10-m *U* component of wind (m $s^{-1}$) | −0.09 | 0.67 | 2.14 |
| 10-m *V* component of wind (m $s^{-1}$) | −0.25 | 0.39 | 1.11 |
| Surface net solar radiation (W $m^{-2}$) | 130 | 162 | 220 |
| Surface net thermal radiation (W $m^{-2}$) | −102 | −77.5 | −49.8 |
| (b) | | | |
| Gauge elevation (m) | 1700 | 3700 | 4300 |
| Area ($km^2$) | 420 | 3580 | 38 010 |
| Average slope (°) | 36 | 144 | 276 |
| Mean annual air temperature °C | −1.95 | 2.75 | 8.85 |
| Climate moisture index | −70 | −18 | 65 |
| Inundation extent (%) | 0.0 | 1.5 | 83.0 |
| Wetland extent (%) | 0.0 | 2.5 | 19.5 |
| Permafrost extent (%) | 0.0 | 0.5 | 16.6 |
| Snow-cover extent (%) | 16.7 | 42.6 | 54.5 |
| Degree of regulation | 0.0 | 48.3 | 817.5 |
| Lake area (%) | 0.0 | 5.9 | 31.6 |
| Grassland (%) | 0.0 | 0.4 | 24.6 |
| Forest (%) | 0.0 | 7.6 | 86.9 |
| Cropland (%) | 8.7 | 82.7 | 100.0 |
| Shrubland (%) | 0.00 | 2.1 | 15.6 |

TABLE 2. Summary statistics of four key catchment attributes for each catchment in the study. All summaries are taken from the ERA5 deterministic forecast from 2000 to 2023, with gauge elevation, drainage area, and catchment boundaries provided by the U.S. Bureau of Reclamation metadata.

| USGS id | USGS name | Mean precipitation (mm yr$^{-1}$) | Mean evaporation (mm yr$^{-1}$) | Drainage area (km$^2$) | Gauge elevation (m) |
|---|---|---|---|---|---|
| 12362500 | South Fork Flathead River nr Columbia Falls, Montana | 908.85 | 434.35 | 4320 | 930 |
| 13037500 | Snake River near Heise, Idaho | 792.05 | 441.65 | 14 900 | 1530 |
| 6054500 | Missouri River at Toston, Montana | 605.9 | 467.20 | 37 920 | 1190 |
| 9361500 | Animas River at Durango, Colorado | 824.9 | 485.45 | 1820 | 1980 |
| 9251000 | Yampa River near Maybell, Colorado | 613.2 | 459.9 | 8760 | 1800 |
| 12301933 | Kootenai River below Libby Dam, near Libby, Montana | 795.7 | 394.2 | 23 310 | 640 |
| 12105900 | Green River Below Howard A Hanson Dam, Washington | 1708.2 | 514.65 | 570 | 300 |
| 9109000 | Taylor River below Taylor Park Reservoir, Colorado | 609.55 | 405.15 | 660 | 2800 |
| 9050700 | Blue River below Dillon, Colorado | 693.5 | 423.4 | 870 | 2670 |
| 10128500 | Weber River near Oakley, Utah | 704.45 | 521.95 | 420 | 2020 |
| 11251000 | San Joaquin River below Friant, California | 1157.05 | 507.35 | 4340 | 90 |
| 11266500 | Merced River A Pohono Bridge near Yosemite, California | 1069.45 | 430.70 | 830 | 1180 |
| 12409000 | Colville River at Kettle Falls, Washington | 638.75 | 478.15 | 2610 | 430 |
| 12451000 | Stehekin River at Stehekin, Washington | 1657.10 | 357.70 | 830 | 340 |
| 14181500 | North Santiam River at Niagara, Oregon | 1602.35 | 459.90 | 1170 | 330 |
| 9406000 | Virgin River at Virgin, Utah | 419.75 | 317.55 | 2480 | 1070 |
| 8378500 | Pecos River Near Pecos, New Mexico | 766.5 | 547.50 | 490 | 2290 |
| 13183000 | Owyhee River below Owyhee Dam | 375.95 | 321.2 | 28 900 | 710 |

$$L_\tau(y, \hat{y}) = \begin{cases} (\tau - 1)(y - \hat{y}_\tau), & \text{if } y < \hat{y}_\tau, \\ \tau(y - \hat{y}_\tau), & \text{if } y \geq \hat{y}_\tau \end{cases}. \quad (1)$$

We calculate the loss using our model's corresponding quantile loss function for each threshold prediction. We then sum these losses to get a total loss for the set of quantiles, $L_{\text{tot}}$, in Eq. (2). This equally weights each of the thresholds in the loss, which is also the weighting chosen in the Water Supply Forecast Rodeo DrivenData (2024):

$$L_{\text{tot}}(y, \hat{y}) = \frac{1}{3}[L_{0.1}(y, \hat{y}_{0.1}) + L_{0.5}(y, \hat{y}_{0.5}) + L_{0.9}(y, \hat{y}_{0.9})]. \quad (2)$$

We normalize $L_{\text{tot}}$ by dividing it by the corresponding loss that would be obtained by a model that predicts the climatological quantile thresholds at each catchment, $L_{\text{clim}}$, which is analogous to the normalizing factor in the Nash–Sutcliffe efficiency (NSE). This is done so that the loss does not consider the size of the catchment in determining the magnitude of the error but rather how much the model improves upon the simple benchmark of climatology, in terms of the percentage reduction of error. For each calendar day (e.g., 1 July), the climatology model looks at all historical observations of river discharge on that day across all years in the training data. It then calculates specific quantiles from this distribution. For example, the 10% quantile prediction for 1 July represents the discharge value that was exceeded by 10% of all 1 July measurements in the historical record. We can then use this to define a cumulative quantile efficiency score (CQES), as $1 - (L_{\text{tot}}/L_{\text{clim}})$. A CQES value of 0 would respond to a model that is as skillful as climatology, and a CQES value of 1 would respond to a model that perfectly forecasts the observed values:

$$\text{CQES}(y, \hat{y}) = 1 - \frac{L_{\text{tot}}(y, \hat{y})}{L_{\text{clim}}(y, y_{\text{clim}})}. \quad (3)$$

As we wish the *Hydra*-LSTM to be useful to operational hydrologists, providing context into the methods by which machine learning models are trained is useful. To train machine learning models, we perform many stochastic updates to the model's parameters to minimize the loss of the model's predictions over a subset, hereafter "batch" of the examples in the training set. The gradient of the loss with respect to the model parameters is used in a process called stochastic gradient descent. We use a variation of this method that considers the gradients at previous batches, known as the Adam optimizer (Kingma and Ba 2014). This is a form of parameter optimization.

Each training example is defined by the date a forecast is made and the catchment it is made for. In each batch, we randomly decided which catchment to make predictions for and randomly sampled potential forecast dates from which the prediction was made without replacement, step 1 in diagram 4. For these catchments and dates, the relevant data are extracted in step 2 and fed into the model being trained on, outputting predictions for the different quantile thresholds in each training example in step 3. The predictions and actual observed values are then input into the loss function $L_{\text{prop}}$, step 4, and gradient descent is performed to update the model parameters, step 5. An epoch is complete when all forecast dates in the training set have been trained on, and we compute the loss for each epoch on a validation set. An early stopping check then takes the validation loss and decides whether or not to end the training procedure stopping the training loop if a new minimum validation loss has not been reached in the last 20 epochs, step 6. The early stopper does

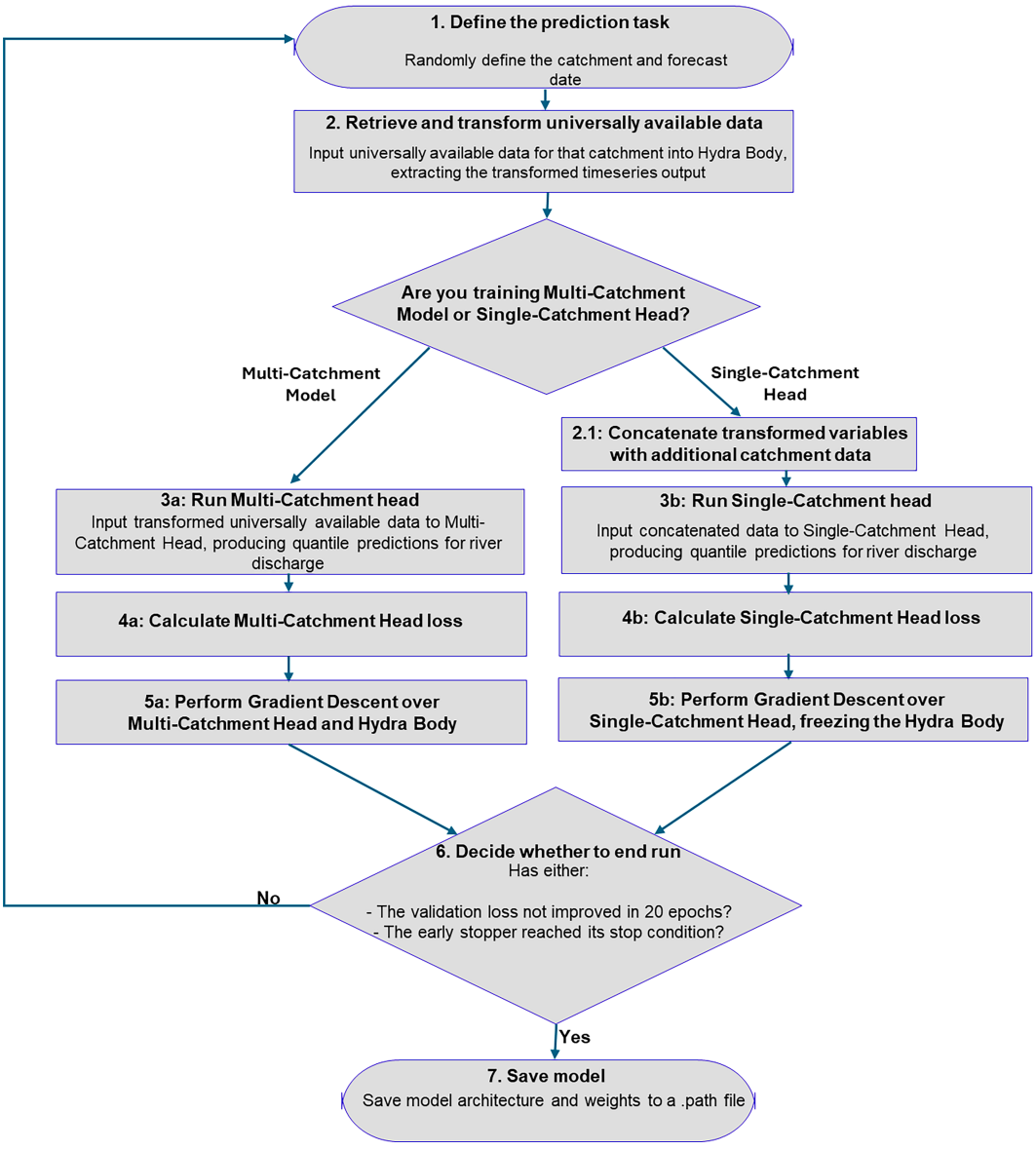


FIG. 4. Training flowchart for *Hydra*-LSTM. The Multi-Catchment Head and *Hydra* Body are trained initially, and then, the Single-Catchment Heads can be trained using the outputs from the *Hydra* Body as some of its inputs.

this depending on whether the maximum number of epochs has been reached or if the validation loss has started to plateau. If the early stopper does end the training run, the model and its parameter weights are saved to be used later.

For each model, we train it on a subset of 19 years and choose 2 years to validate. All multicatchment models are trained on all catchments. The performance reported in the results section is then computed on a final year withheld from the training and validation set. We do this with 11 different potential test years, a process known as 11-fold cross validation, or leave-one-out validation applied 11 times.

For the *Hydra*-LSTM in particular, the *Hydra* Body and Multi-Catchment Head are trained in tandem, and then, the Single-Catchment Heads are trained separately using the trained *Hydra* Body. This is because in operational usage, we expect a *Hydra* Body and Head to be made initially, and then, individual forecasters would train their own Single-Catchment Heads to incorporate their own additional data. This means that the Single-Catchment Head will have been trained separately from the Multi-Catchment parts, and so we want to test whether or not it will be able to satisfactorily combine the information from the *Hydra* Body with its additional data to make its prediction. The training of the *Hydra*-LSTM is summarized in Fig. 4.

For all models, there was a list of potential hyperparameters that we could have chosen relating to the features of each model architecture. To decide which were best, we trained all possible sets of hyperparameters from the list,

TABLE 3. Hyperparameters tested for each model architecture. The hyperparameters that were found to minimize the quantile loss in the validation dataset are in bold. The hidden size relates to the number of parameters in each layer of the LSTM, while the number of layers determines the number of processing steps required to transform the inputs into a satisfactory prediction. The learning rate is a determiner of how much to calibrate the parameters of the model given a particular set of training examples. Bidirectionality determines whether or not the model can use information on what happens in subsequent days to inform its representation of a previous day and can be useful in some complex problems. Finally, dropout is a process by which parameters in the model are randomly turned off and can often prevent an overreliance on particular features. Models are color coded throughout the figures as follows: Single-catchment LSTMs (light blue), multicatchment LSTM without river discharge (pink), multicatchment LSTM with river discharge (purple), Flag LSTM (orange), and *Hydra*-LSTM (yellow).

| Model | Hidden size | Number of layers | Learning rate | Bidirectional | Dropout |
|---|---|---|---|---|---|
| Single catchment | [16, 64, **128**] | [**1**, 2, 3] | [$\mathbf{1 \times 10^{-3}}$, $1 \times 10^{-5}$] | [**No**, Yes] | [**0**, 0.1, 0.4] |
| Multicatchment: without river discharge | [64, **128**, 256] | [1, **2**, 3] | [$\mathbf{1 \times 10^{-3}}$, $1 \times \mathbf{10^{-3}}$] | [**No**, Yes] | [0, **0.2**, 0.4] |
| Multicatchment: with river discharge | [64, **128**, 256] | [1, **2**, 3] | [$\mathbf{1 \times 10^{-3}}$, $1 \times 10^{-5}$] | [**No**, Yes] | [0, **0.2**, 0.4] |
| Flag LSTM | [64, **128**, 256] | [1, **2**, 3] | [$\mathbf{1 \times 10^{-3}}$, $1 \times 10^{-5}$] | [**No**, Yes] | [0, **0.2**, 0.4] |
| *Hydra*-LSTM | **Body:** [64, **128**, 256] **Head:** [16, **32**, 64] | **Body:** [1, **2**, 3] **Head:** [**1**, 2] | [$\mathbf{1 \times 10^{-3}}$, $1 \times 10^{-5}$] | [**No**, Yes] | [**0**, 0.2, 0.4] |

recorded the validation loss for the years 2020 and 2022, and chose the hyperparameter set that optimized the validation of the model for the rest of our analysis. Neither of these years is used as test years when evaluating our models. The hyperparameters tested are summarized in Table 3.

## 6. Results

### a. Models without river discharge available as input

The multicatchment LSTM trained without river discharge as an input performs nearly identically to the *Hydra*-LSTM with a Multi-Catchment Head, both having a CQES of 0.12 and having empirical 10% and 90% quantile thresholds within 1% of each other. The scores can be found in Table 4 and Fig. 5. This is expected, as the *Hydra* Body is trained only to minimize the results from the Multi-Catchment Head, and so using the Multi-Catchment Head is akin to using a multicatchment LSTM without river discharge. The Flag LSTM performs slightly worse than the other models in the ungauged setting, with a CQES of 0.09, and its 90% quantile threshold prediction being exceeded only 85% of the time compared to 87% and 88% of the time for the multicatchment LSTM and *Hydra*-LSTM. The distribution of CQES scores (Fig. 5) shows that the distribution of scores for all the models where river discharge is assumed to be unavailable is very similar, with all three curves mostly overlapping. Between a CQES of −0.5 and 0.0, we see the performance of the Flag LSTM being worse than that of the other models, with the curve lying left of the other models.

### b. Models with river discharge available as an input

When river discharge is available as an input to each of these models, introduced into the *Hydra*-LSTM through the use of Single-Catchment Heads, the average skill of each model is drastically increased, with the minimum CQES now being 0.61 compared to a previous maximum of 0.12. The scores in this setting can be found in Table 5 and cumulative CQES distribution plot can be found in Fig. 6. The *Hydra*-LSTM with Single-Catchment Heads has the highest CQES, and the Flag LSTM still has the worst score in the context where river discharge is a usable input. The single-catchment LSTMs produce an 80% confidence interval that is too narrow on average, with its 10% quantile threshold in actuality being exceeded 15% of the time and its 90% quantile threshold only being exceeded 84% of the time. All other models have empirical quantiles within 3% of their target. Overall, we find that the *Hydra*-LSTM with Single-Catchment Heads performs the best of all the models trained with river discharge as an available output. The distribution of average CQES scores for the *Hydra*-LSTM with Single-Catchment Head is the highest compared to all the other models, being the rightmost curve with a lowest CQES score of 0.5 compared to the next best lowest score of below 0.4 for the single-catchment LSTM. The *Hydra*-LSTM with Single-Catchment

TABLE 4. Comparison of models without river discharge as an available input. The best scores for each metric are shown in bold. Models are color coded throughout the figures as follows: Single-catchment LSTMs (light blue), multicatchment LSTM without river discharge (pink), Flag LSTM (orange), and *Hydra*-LSTM (yellow). Metrics shown are the CQES, the proportion of observations exceeding the models' predicted 10% quantile threshold, and the proportion of observations exceeding the models' predicted 90% quantile threshold.

| Model | CQES | Proportion of observations exceeding predicted 10% quantile threshold | Proportion of observations exceeding predicted 90% quantile threshold |
|---|---|---|---|
| Multicatchment: without river discharge | **0.12** | 0.13 | 0.87 |
| Flag | 0.09 | 0.13 | 0.85 |
| *Hydra*-LSTM: Multi-Catchment Head | **0.12** | **0.12** | **0.88** |

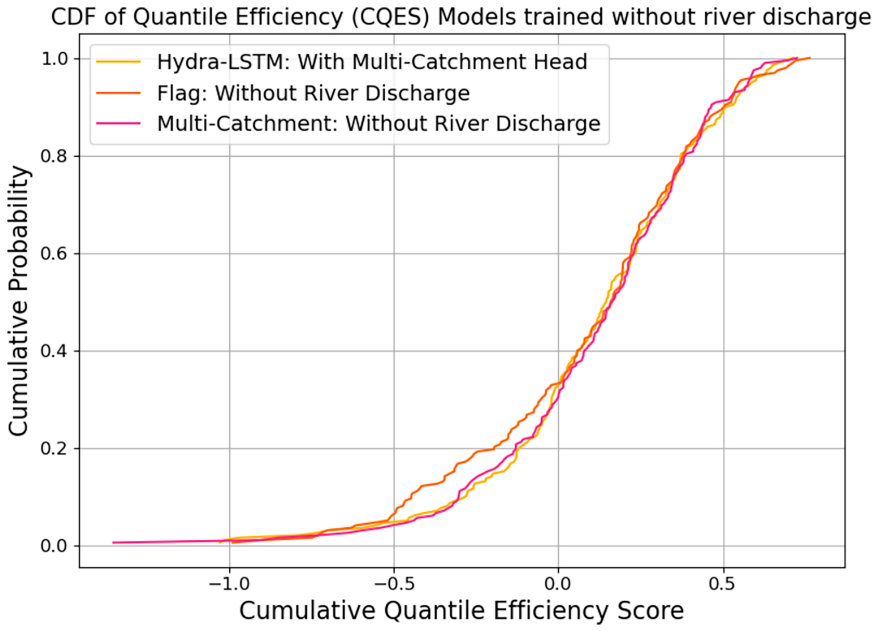


FIG. 5. Cumulative distribution plot showing the range of CQES 3, for each model trained without river discharge as an input. Each individual score is for a single year in a single basin.

Heads has an average CQES of 0.73 compared to 0.71 for the single-catchment LSTM; all other models score lower. The *Hydra*-LSTM with Single Catchment Heads also had the joint most accurate 10% quantile threshold estimate, alongside the Flag LSTM, as can be seen in Table 5.

### c. *Case study: Green river 2001*

In order for hydrologists to be able to reliably use a model, they require assurances that the model will not behave poorly in unseen, or out-of-distribution, conditions. In physical models, this can come from much of the physics being prescribed in model development, which is not true in purely statistical or machine learning models. Instead, to test the ability of our models to create skillful predictions in unseen conditions, we further analyze the January–July river flow in Green River, U.S. Geological Survey (USGS) ID 12105900, for 2001 (Fig. 7). Green River saw particularly low flow in 2001, with a peak flow roughly 3 times less than the average flow in other years. The *Hydra*-LSTM was best able to capture the potential for high flow being much lower than in other years, with the predicted 90% quantile thresholds being much lower than in other years. In comparison, the other models still had relatively high 90% quantile thresholds. The multicatchment LSTM trained without river discharge performed the worst at this catchment, with its 90% quantile threshold remaining much closer to the usual flow seen in that catchment in other years, of approximately 6000 cfs as opposed to the 2000 ft per second seen in that year.

The *Hydra*-LSTM with Single-Catchment Heads also had the best CQES score in the gauged setting for this scenario, as seen in the right column of Fig. 7. Again the Flag LSTM and multicatchment LSTM overestimated the 10% quantile threshold, and this is confirmed when looking at the proportion of 10% quantile thresholds that were exceeded in Table 5. The single-catchment LSTM, on the other hand, had an overly constrained 10% quantile threshold prediction, being exceeded 15% of the time. The corresponding hydrograph shows a high degree of stochasticity in its daily predictions, significantly less smooth than the true observations.

TABLE 5. Comparison of models with river discharge as an available input. The best scores for each metric are shown in bold. Models are color coded throughout the figures as follows: single-catchment LSTMs (light blue), multicatchment LSTM with river discharge (purple), Flag LSTM (orange), and *Hydra*-LSTM (yellow). Metrics shown are the CQES, the proportion of observations exceeding the models' predicted 10% quantile threshold, and the proportion of observations exceeding the models' predicted 90% quantile threshold.

| Model | CQES | Proportion of observations exceeding predicted 10% quantile threshold | Proportion of observations exceeding predicted 90% quantile threshold |
|---|---|---|---|
| Single catchment | 0.71 | 0.15 | 0.84 |
| Multicatchment: with river discharge | 0.66 | 0.07 | **0.91** |
| Flag | 0.61 | **0.09** | 0.92 |
| *Hydra*-LSTM: Single-Catchment Heads | **0.73** | **0.11** | 0.88 |

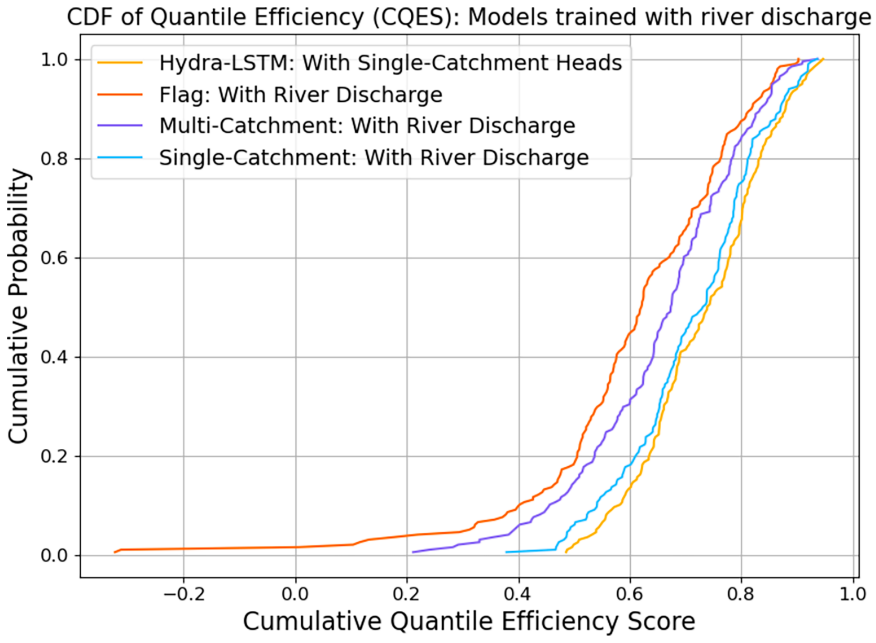


FIG. 6. Cumulative distribution plot showing the range of CQES 3, for each model trained with river discharge as an input. Each individual score is for a single year in a single basin. The CQES scores for the *Hydra*-LSTM are significantly higher than that of the other models ($p = 0.005$).

## 7. Comparison between models

### a. *Hydra-LSTM Multi-Catchment Head at least as skillful as other state-of-the-art machine learning models*

This paper develops a probabilistic model to match existing state-of-the-art architectures while allowing individual forecasters to introduce new inputs after the initial development of the model. We did this by developing a model with three key components: a *Hydra* Body to transform data available across all catchments into something more directly useful for river discharge prediction, a Multi-Catchment *Hydra* Head to take the transformations from the *Hydra* Body and make quantile predictions for river discharge when additional data are not available, and a suite of Single-Catchment *Hydra* Heads to take in additional data available only at select catchments and process these alongside the transforms produced by the *Hydra* Body to make more informed predictions when possible.

In the first setting, when no additional data are available, and only the *Hydra* Body and Multi-Catchment *Hydra* head are used, the performance of the *Hydra*-LSTM is on par with other state-of-the-art machine learning models used in river discharge prediction, namely, the multicatchment LSTM (Kratzert et al. 2019) and the Flag LSTM (Nearing et al. 2023). Not only does it have the best cumulative score, as defined in Eq. (3) and shown in Table 4 and Fig. 5, but it also has the most accurate quantile thresholds. The *Hydra*-LSTM in this setting is expected to have results nearly identical to the multicatchment LSTM, as the *Hydra* Body is trained only to minimize the loss from the Multi-Catchment *Hydra* Head. This means that the combination is the same as a two-layer LSTM with a different hidden size for each layer. It may also be expected that the Flag LSTM would perform slightly worse, as it attempts to create predictions both when historic river discharge is available as an input and when it is not, without having an auxiliary head as the *Hydra*-LSTM can have.

### b. *Hydra-LSTM Single-Catchment Heads at least as skillful and effective at using additional variables as if it were introduced in the initial set*

The training of the *Hydra*-LSTM Single-Catchment Heads is very efficient, as it needs only to be a single-layer LSTM, using the transformations learned by the *Hydra* Body as the main set of inputs. Using parallel processes on an A100 graphics processing unit (GPU), we could train 11 different folds of each of the models in under 2 h and train 11 different Single-Catchment Heads in under an hour. Because of its efficiency, we do not see the required resources to train an additional Single-Catchment *Hydra* Head as a significant computational burden for most users. The efficacy of the Single-Catchment *Hydra* Head is also evident from our results, with the distribution of scores lying strongly to the right, meaning higher scores than that of the other models, Fig. 6. It had a significantly higher CQES score of 0.73 than the Flag LSTM, which had the lowest score of 0.61 of all the models, and its quantile thresholds were within exceeded 2% of their intended values as seen in Table 5. We believe that the significant improvements compared to the Flag LSTM, which is the only other model that can be used with variable data available, show that it may be a better alternative in many cases, for example, when the additional data available are only

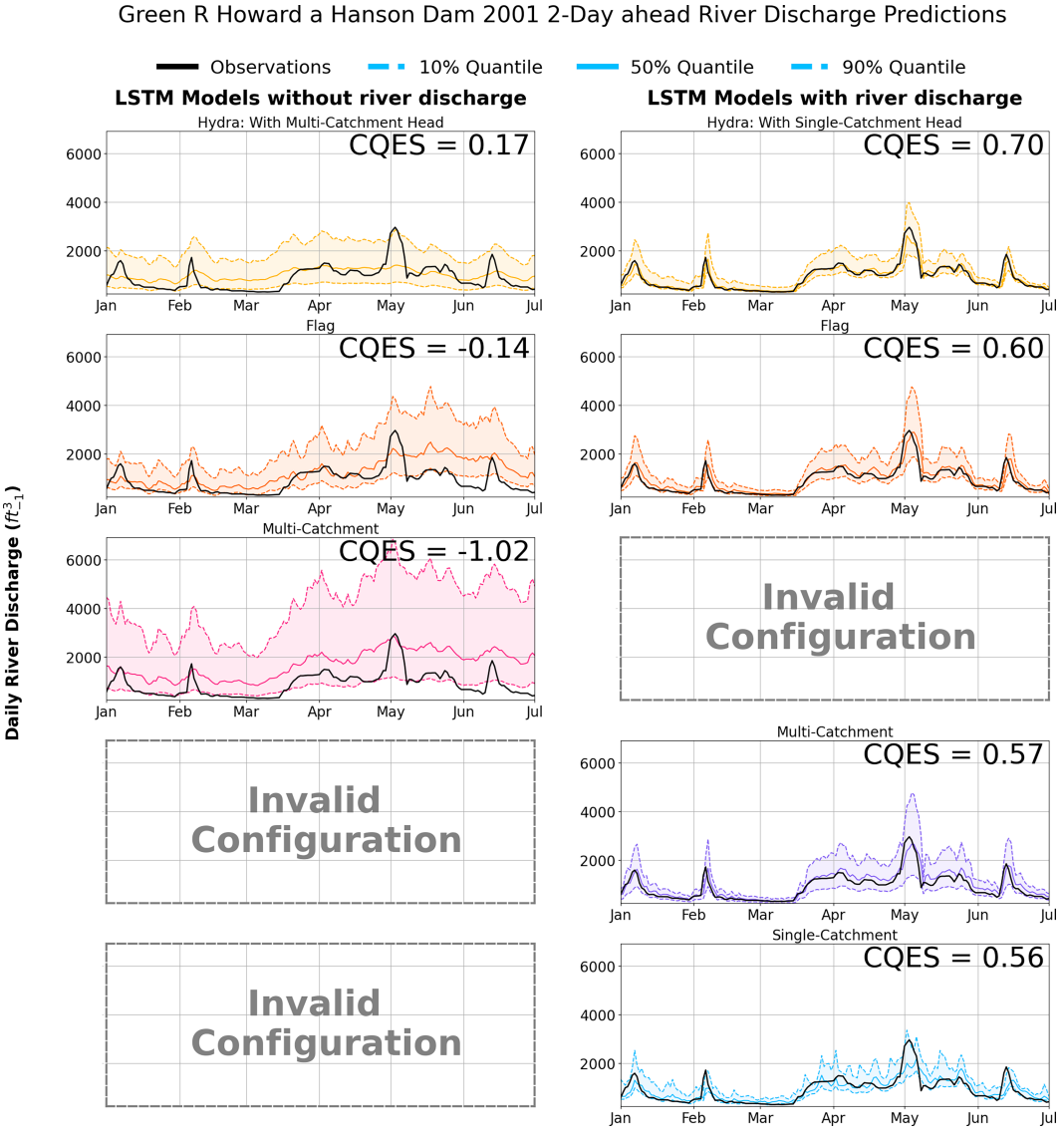


FIG. 7. Comparison of the hydrographs of all models for Green River Howard A Hanson Dam, USGS ID 12105900, in 2001. Not all models are usable with all potential datasets, and in this case, "invalid configuration" is written on the graph. (left) Each model performs without river discharge as a potential input, and (right) the models with river discharge as a potential input.

available for a single catchment. This is because the Flag LSTM and the multicatchment LSTM with river discharge were trained with river discharge across all catchments and so were trained on more data than might be available for another variable we might wish to introduce, such as radar measurements were taken only at a single catchment or upstream information for a particular catchment. It may be that changing the proportion of training devoted to a variable being available as an input and not available as an input might have improved the Flag LSTM; however, this is an additional complexity decreasing the usability of the Flag LSTM. Training a Flag LSTM or multicatchment would also require the forecaster to have access to and train the model on all the relevant data for all the catchments the model was trained on, whereas to train an additional Single-Catchment Head in the *Hydra*-LSTM, the forecaster would only need to train on the data for their own catchments, as the *Hydra* Body does not need to be retrained on each catchment.

We recalculated the CQES for each of these models during the high flow period exclusively (April–July) and found that there was no difference in the ordering of the performance of each of the models. The CQES was slightly increased by an average of 0.07 for the models with river discharge as an input, which we expect is because river discharge is an especially more useful input for prediction when in a high flow period.

### c. *Hydra-LSTM Single-Catchment Heads offer better performance than training a new single-catchment LSTM*

Not only does the *Hydra*-LSTM outperform the other models that are trained across multiple catchments, but we have also shown that it, on average, outperforms the use of different LSTMs at each catchment. Table 5 shows that the single-catchment LSTM, while having the second highest average CQES, tends to underestimate the uncertainty in its forecasts, with an average interval between its 10% and 90% quantile thresholds of 69%, compared to one of 77% for the *Hydra*-LSTM. The hydrographs in Fig. 7 suggest that single-catchment LSTMs may overrely on persistence. Its predictions for the 10% quantile threshold also seemed to be particularly erratic, having a high daily variability that was much more than seen in other models or in the actual flow in the catchment that is more than is seen in this catchment. This may suggest an inability to fully appreciate some of the underlying hydrological behavior, having been trained on the data on only one catchment as opposed to being driven by a model trained over many catchments, as the *Hydra* Body is.

Overall, this suggests that the *Hydra*-LSTM is just as skillful a method as the other models when there are no additional data available that a local forecaster would wish to add as a predictor for their catchment, and that if desired, a forecaster can introduce additional data in a Single-Catchment Head in a way that is just as effective as retraining a multicatchment LSTM.

### d. *Aspects for future work*

Our results have shown the ability of the *Hydra*-LSTM to introduce catchment-specific data, even when relatively few catchments are used in training. Kratzert et al. (2024) have shown that the performance of traditional LSTMs improves as more catchments are used, even when trained on hundreds of catchments globally, and in the future, we plan to test whether or not these benefits apply to the *Hydra*-LSTM too. One can also assess how much historical data are needed for the Single-Catchment Heads to learn from. The *Hydra* Body represents the bulk of the model parameters, and so it may allow for additional skill to be gained from additional data by the Single-Catchment Head that has too short a historical record to be used by a single-catchment LSTM.

There are also many other promising techniques and findings in machine learning hydrological forecasting that can be used alongside the *Hydra*-LSTM frameworks. These include focusing on the diversity of catchments used in training (Fang et al. 2022), looking at residual errors (Li et al. 2021) and other model changes such as using multistate vectors or self-training (Yin et al. 2021; Yoon and Ahn 2024). Now that we have provided evidence on the usefulness of *Hydra*-LSTM itself, exploring how we can combine it with other advances could prove beneficial to increasing the skill and usability of river discharge models. For example, for longer lead times, it may be useful to introduce predictions from atmospheric forecast models such as the ECMWF IFS (Persson and Grazzini 2007). This addition can be introduced into the *Hydra*-LSTM architecture, by using the hindcast–forecast LSTM developed by Nearing et al. (2023) as the model blocks.

It is also possible to use flag indicators in the *Hydra*-LSTM if variables might be best introduced in the *Hydra* Body with a flag indicator instead of as a separate head. Future research may wish to explore how these two means of introducing data optionally can be used together and when it is worth rebuilding the *Hydra* Body instead of introducing additional data in the Single-Catchment Head. Flags have the unique advantage of allowing us to use them still when data are only intermittently available. In contrast, the Single-Catchment Heads have the advantage of being able to introduce new variables without having to retrain the entire model. Testing the feasibility of adding flag time series into the *Hydra* Model would allow us to take full advantage of different data types at each catchment.

Another strength of the *Hydra*-LSTM that future work should explore is that it can have multiple, partially specialized, Multi-Catchment Heads. It is possible, for instance, to train one other Multi-Catchment Head that combined the outputs from the *Hydra* Body with historic river discharge and was trained over all catchments that had river discharge. Or one could train a different Multi-Catchment Head for each country, using the respective met-office models from each country as additional inputs. This may be worth doing when not all the catchments we want our models to be useful at have a particular variable available, but enough do that it is worth creating an alternate head and training it on all relevant data across catchments.

## 8. Conclusions

Before this paper, no method existed that allowed for the easy introduction of new variables outside of an initial set. In this paper, we have developed the *Hydra*-LSTM, a new machine learning architecture for predicting river discharge in a multicatchment setting. This method allows organizations to create large-scale models to make predictions across multiple catchments without restricting them to any particular subset of data to be used as predictors. The *Hydra*-LSTM is equivalent in skill to other state-of-the-art architectures when a catchment manager does not need to introduce their own variables, and the Single-Catchment Heads of the *Hydra*-LSTM can introduce new variables into the predictor set in a way that is more effective than introducing in the base set of predictands in other models. The Single-Catchment Heads being a more effective means of introducing additional variables is a bonus; it is the only realistic option for a local forecaster to add their own variables into a model. We encourage future work to test the *Hydra*-LSTM to introduce new variables that are specific to individual regions, to increase the number of catchments the *Hydra*-LSTM is trained on to assess its scaling properties, and to test in cases for which available data are limited.

*Acknowledgments.* This work was supported by the Advanced Frontiers for Earth System Prediction Doctoral Training Programme, funded by the University of Reading.

*Data availability statement.* All data and code required to reproduce the figures in this manuscript can be found in the following GitHub branch: https://github.com/KarRups/Hydra_Code/tree/Paper_Submission. Catchment averages of ERA5 reanalysis from Hersbach et al. (2020) are computed using the shapefiles provided by the U.S. Bureau of Reclamation DrivenData (2024), with the gridded data stored on the ECMWF's HPC systems and available upon request, or can be gathered using the Copernicus Climate Data Stores (CDS). Catchment attributes are acquired from the BasinAtlas dataset (Linke et al. 2019).

## REFERENCES


Clerc-Schwarzenbach, F., G. Selleri, M. Neri, E. Toth, I. van Meerveld, and J. Seibert, 2024: Large-sample hydrology—A few camels or a whole caravan? *Hydrol. Earth Syst. Sci.*, **28**, 4219–4237, https://doi.org/10.5194/hess-28-4219-2024.

DrivenData, 2024: Reclamation water supply forecast challenge. Accessed 4 September 2024, https://www.drivendata.org/competitions/254/reclamation-water-supply-forecast-dev/.

Fang, K., D. Kifer, K. Lawson, D. Feng, and C. Shen, 2022: The data synergy effects of time-series deep learning models in hydrology. *Water Resour. Res.*, **58**, e2021WR029583, https://doi.org/10.1029/2021WR029583.

Fleming, S. W., and A. G. Goodbody, 2019: A machine learning metasystem for robust probabilistic nonlinear regression-based forecasting of seasonal water availability in the US west. *IEEE Access*, **7**, 119 943–119 964, https://doi.org/10.1109/ACCESS.2019.2936989.

Gauch, M., F. Kratzert, D. Klotz, G. Nearing, J. Lin, and S. Hochreiter, 2021: Rainfall–runoff prediction at multiple timescales with a single long short-term memory network. *Hydrol. Earth Syst. Sci.*, **25**, 2045–2062, https://doi.org/10.5194/hess-25-2045-2021.

Fleming, S. W., D. C. Garen, A. G. Goodbody, C. S. McCarthy, and L. C. Landers, 2021: Assessing the new Natural Resources Conservation Service water supply forecast model for the American West: A challenging test of explainable, automated, ensemble artificial intelligence. *J. Hydrol.*, **602**, 126782, https://doi.org/10.1016/j.jhydrol.2021.126782.

Hersbach, H., and Coauthors, 2020: The ERA5 global reanalysis. *Quart. J. Roy. Meteor. Soc.*, **146**, 1999–2049, https://doi.org/10.1002/qj.3803.

Hunt, K. M., G. R. Matthews, F. Pappenberger, and C. Prudhomme, 2022: Using a Long Short-Term Memory (LSTM) neural network to boost river streamflow forecasts over the Western United States. *Hydrol. Earth Syst. Sci.*, **26**, 5449–5472, https://doi.org/10.5194/hess-26-5449-2022.

Jahangir, M. S., J. You, and J. Quilty, 2023: A quantile-based encoder-decoder framework for multi-step ahead runoff forecasting. *J. Hydrol.*, **619**, 129269, https://doi.org/10.1016/j.jhydrol.2023.129269.

Kingma, D. P., and J. Ba, 2014: Adam: A method for stochastic optimization. arXiv, 1412.6980v9, https://doi.org/10.48550/arXiv.1412.6980.

Klotz, D., F. Kratzert, M. Gauch, A. Keefe Sampson, J. Brandstetter, G. Klambauer, S. Hochreiter, and G. Nearing, 2022: Uncertainty estimation with deep learning for rainfall–runoff modeling. *Hydrol. Earth Syst. Sci.*, **26**, 1673–1693, https://doi.org/10.5194/hess-26-1673-2022.

Koenker, R., 2005: *Quantile Regression*. Vol. 38. Cambridge University Press, 376 pp.

Kratzert, F., D. Klotz, C. Brenner, K. Schulz, and M. Herrnegger, 2018: Rainfall–runoff modelling Using Long Short-Term Memory (LSTM) networks. *Hydrol. Earth Syst. Sci.*, **22**, 6005–6022, https://doi.org/10.5194/hess-22-6005-2018.

——, ——, M. Herrnegger, A. K. Sampson, S. Hochreiter, and G. S. Nearing, 2019: Toward improved predictions in ungauged basins: Exploiting the power of machine learning. *Water Resour. Res.*, **55**, 11 344–11 354, https://doi.org/10.1029/2019WR026065.

——, M. Gauch, D. Klotz, and G. Nearing, 2024: HESS opinions: Never train a long short-term memory (LSTM) network on a single basin. *Hydrol. Earth Syst. Sci.*, **28**, 4187–4201, https://doi.org/10.5194/hess-28-4187-2024.

Li, D., L. Marshall, Z. Liang, A. Sharma, and Y. Zhou, 2021: Characterizing distributed hydrological model residual errors using a probabilistic long short-term memory network. *J. Hydrol.*, **603**, 126888, https://doi.org/10.1016/j.jhydrol.2021.126888.

Li, K, and S. Razavi, 2024: What controls hydrology? an assessment across the contiguous united states through an interpretable machine learning approach. *J. Hydrol.*, **642**, 131835, https://doi.org/10.1016/j.jhydrol.2024.131835.

Linke, S., B. Lehner, C. Ouellet Dallaire, J. Ariwi, G. Grill, M. Anand, and P. Beames, 2019: Global hydro-environmental Sub-basin and river reach characteristics at high spatial resolution. *Scientific Data*, **6**, 283–283.

Liu, J., Y. Bian, K. Lawson, and C. Shen, 2024: Probing the limit of hydrologic predictability with the transformer network. *J. Hydrol.*, **637**, 131389, https://doi.org/10.1016/j.jhydrol.2024.131389.

Nearing, G., and Coauthors, 2023: AI increases global access to reliable flood forecasts. arXiv, 2307.16104v4, https://doi.org/10.48550/arXiv.2307.16104.

Persson, A., and F. Grazzini, 2007: User guide to ECMWF forecast products. *Meteor. Bull. 3.2*, 153 pp.

U.S. Geological Survey, 2024: National water information system data available on the world wide web (water data for the nation). Accessed 15 August 2024, https://waterdata.usgs.gov/nwis/.

Wang, M., Yu. Zhang, Yan. Lu, Li. Gao, and L. Wang, 2023: Attribution analysis of streamflow changes based on large-scale hydrological modeling with uncertainties. *Water Resour. Manage.*, **37**, 713–730, https://doi.org/10.1007/s11269-022-03396-7.

Xie, K., Pan. Liu, J. Zhang, D. Han, G. Wang, and C. Shen, 2021: Physics-guided deep learning for rainfall-runoff modeling by considering extreme events and monotonic relationships. *J. Hydrol.*, **603**, 127043, https://doi.org/10.1016/j.jhydrol.2021.127043.

Yin, H., X. Zhang, F. Wang, Y. Zhang, R. Xia, and J. Jin, 2021: Rainfall-runoff modeling using LSTM-based multi-state-vector sequence-to-sequence model. *J. Hydrol.*, **598**, 126378, https://doi.org/10.1016/j.jhydrol.2021.126378.

Yoon, S., and K.-H. Ahn, 2024: Self-training approach to improve the predictability of data-driven rainfall-runoff model in hydrological data-sparse regions. *J. Hydrol.*, **632**, 130862, https://doi.org/10.1016/j.jhydrol.2024.130862.