https://doi.org/10.5194/egusphere-2024-3649
Preprint. Discussion started: 21 February 2025

Spatially Resolved Rainfall Streamflow Modeling in Central Europe

Marc Aurel Vischer[1], Noelia Otero[1], and Jackie Ma[1]

[1]Fraunhofer Heinrich-Hertz Institute, Applied Machine Learning Group, 10553 Berlin, Germany

Correspondence: Marc Aurel Vischer (marc.aurel.vischer@hhi.fraunhofer.de) and Jackie Ma (jackie.ma@hhi.fraunhofer.de)

Abstract. Climate change increases the risk of disastrous floods and makes intelligent fresh water management an ever more important issue for society. A central prerequisite is the ability to accurately predict the water level in rivers from a range of predictors, mainly meteorological forecasts. The field of rainfall runoff modeling has seen neural network models surge in popularity over the last few years, but a lot of this early research on model design has been conducted on catchments with smaller size and a low degree of human impact to ensure optimal conditions. Here we present a pipeline that extends the previous neural network approaches in order to better suit the requirements of larger catchments or those characterized by human activity. Unlike previous studies, we do not aggregate the inputs per catchment, but train a neural network to predict local runoff spatially resolved on a regular grid. In a second stage, another neural network routes these quantities into and along entire river networks. The whole pipeline is trained end-to-end, exclusively on empirical data. We show that this architecture is able to capture spatial variation and model large catchments accurately, while increasing data efficiency. Furthermore, it offers the possibility to interpret and influence internal states due to its simple design. Our contribution helps to make neural networks more operations-ready in this field and opens up new possibilities to more explicitly account for human activity in the water cycle.

1 Introduction

As one of the most frequent and destructive natural disasters, floods are expected to become more common due to climate change (Bevacqua et al., 2021) and more hazardous as the worldwide population in high risk areas is likely to increase (Kam et al., 2021). Heavy precipitation is expected to become more frequent, which will increase flooding risks (Gründemann et al., 2022). Europe is becoming increasingly vulnerable to flooding due to large-scale atmospheric patterns that lead to widespread precipitation extremes (Bevacqua et al., 2021) and certain landscape properties. This study focuses on river floods in central Europe, where heavy precipitation and snowmelt are driving the expansion of flood-impacted areas (Fang et al., 2024). Accurate prediction of such events is the foundation for creating resilience and preventing material damages, displacement of people and loss of human lives. The field of hydrological research concerned with predicting river levels from meteorological variables is called rainfall streamflow modeling[1]. The aim is to capture the process how precipitation feeds into rivers and other bodies of water. Predicting runoff, i.e. the amount of excess precipitation being drained away on the surface, requires modeling

[1]Another commonly used term is rainfall runoff modeling. As this paper aims to predict stream flow in rivers, we decided to use the more specific term rainfall streamflow modeling, but we will use rainfall runoff modeling to refer to the general literature.

https://doi.org/10.5194/egusphere-2024-3649
Preprint. Discussion started: 21 February 2025

different processes that take place inside or right above the ground, such as evaporation and seepage. It hinges on keeping some record of the state of the surface, e.g. the amount of precipitation in the last days or how much water is stored as snow during winter season. These processes are highly localized, and as a next step, the resulting local runoff needs to be converted into streamflow along a network of rivers. This modeling step is called routing. There is a large body of research employing (conceptually simplified) physical models for both these tasks (Beven, 2012). Furthermore, models based on neural networks have increasingly been proposed in recent years, e.g. Kratzert et al. (2018); Nearing et al. (2024). We build upon this line of work by introducing a neural network that performs both local rainfall-runoff modeling *on a regular grid* and routing along the river network to predict streamflow measured at river gauges. All of this is learned in a data-driven, end-to-end fashion. Here, we present a spatially distributed modeling framework, more versatile than other neural networks as it captures the spatial co-variability of the input features, enhancing prediction accuracy in larger basins (Yu et al., 2024). This approach allows for controllability and scientific discovery, and it is ready to scale to higher spatial and temporal resolution.

The following Section 2 discusses in detail which types of neural networks have been considered for rainfall runoff prediction and routing, and explain our contribution to this ongoing field of research. In Section 3 we describe our model architecture and introduce a novel, publicly available dataset for spatially resolved rainfall streamflow modeling in five river basins in Germany and neighboring countries. Section 4 presents the main results from the experiments. Section 5 concludes with a brief outlook onto future directions, highlighting the influence of human activity.

2 Related Work

We start this section by introducing a classification scheme for rainfall streamflow models. This scheme will allow us to keep an overview as we subsequently present previous work on neural networks in rainfall runoff modeling in general, and spatially resolved processing and routing in particular. We carve out how our approach is different and end this section with an overview over this paper's contributions.

2.1 Typology of Rainfall Runoff Models

We adapt a classification scheme for rainfall streamflow models originally introduced by Sitterson et al. (2018): Depending on the level of abstraction, models are said to be *empirical* (also *detailed* or *physical*) if they involve physical equations of the involved processes (Horton et al., 2022). *Conceptual* (or physically inspired) models make some substantial simplifications, but still contain (abstract) subsystems or quantities that can be identified with physical entities. Finally, *statistical* models refrain from explicit modeling of anything physical and instead focus on the statistical relationship between inputs and outputs exclusively. Our approach is based on neural networks and falls into the latter category, while most operational models such as LISFLOOD (Van Der Knijff et al., 2010) fall under the conceptual category.

Another criterion for classifying models is the way space is represented in the model: *Lumped* models aggregate all variables (temporal or static) across a station's catchment area before modeling starts. Not representing the spatial extension of a catchment can be a reasonable modeling assumption for small catchments, but it implies losing the opportunity to model

https://doi.org/10.5194/egusphere-2024-3649
Preprint. Discussion started: 21 February 2025

spatial co-variance within the catchment, such as the different effect of heavy rainfall over a forested area versus on sealed soil. *Distributed* models on the other hand explicitly model local processes, usually on a grid, less commonly on an irregular mesh vector-based, e.g. Hitokoto and Sakuraba (2020); Sun et al. (2022). Most physical or conceptual models fall under the latter category, as the underlying formulas are local and it is straightforward to resolve them on a regular grid for computation. Neural networks in this domain on the other hand started to be developed as lumped models for a combination of historical and technical reasons which we discuss in the next Subsection 2.2.1. In between sits a class of models called *semi-distributed*, where some sort of sub-structure is modeled. Many routing models fall under this category. An example of a neural network based routing model is Nearing et al. (2024), where a network of gauging stations is modeled with high temporal resolution, but not the processes inside each station's catchment area. As we detail below, our model first predicts runoff fully distributed in space, then mapping these runoffs onto the river network in a second stage.

2.2 Neural Networks in Rainfall Runoff Modeling

Neural networks have been used for rainfall runoff modeling since the 1990s (Smith and Eli, 1995), but have surged in popularity since Kratzert et al. (2018), when long short term memory (LSTM) layers (Hochreiter and Schmidhuber, 1997) were employed for the first time. Kratzert et al. (2019c) then described the beneficial effects of adding static information about the locations to the meteorological inputs, albeit in an aggregated manner. This type of model has since been demonstrated to predicting streamflow more accurately than models not based on neural networks across a variety of locations and experimental setups (Lees et al., 2021; Mai et al., 2022; Clark et al., 2024). It also transfers more readily to ungauged basins (Kratzert et al., 2019b). Calibrating physical or conceptual rainfall runoff models usually requires hand-crafting ancillary input features to support the meteorological forcing variables, such as catchments' climate type or hydrological signature (see Beven (2012) for an overview). Sometimes, the dataset needs to be partitioned into hydrologically homogeneous subsets, on which separate parameters are then calibrated (Beven, 2012). Neural networks do not require such human labor and in contrast profit from processing all catchments indiscriminately and with a single model (Kratzert et al., 2024). They are capable of extracting task-relevant information from a large array of potentially informative, raw static features (Kratzert et al., 2019c). These data sources can include categorical information such as land cover or soil classes, which can not be readily integrated into physical formulas. Neural networks can also leverage entirely new types of input data, such as large-scale remote sensing data (Zhu et al., 2023), concentrations of isotopes (Smith et al., 2023) and chemical compounds (Sterle et al., 2024). As we demonstrate in our study, neural networks can be stacked flexibly into a pipeline designed for a specific task and trained end-to-end without any manual calibration or intermediate steps. Additionally, research on the explainability of neural networks has been conducted by Kratzert et al. (2019a) and Lees et al. (2022), who focused on identifying hydrological quantities and concepts in neural networks. Similarly, using an explainability framework Cheng et al. (2023) extracted hydrological signatures from networks in a data driven fashion. Furthermore, as shown in Jiang et al. (2022) the use of explainability methods can provide a better understanding about the dominant flooding mechanisms across different catchments. Explainability is crucial for reliable operations in real-life applications because it allows for controlling of risk, as well as for scientific discovery (Shen et al., 2018). In summary, LSTM-based models have been firmly established as state of the art in rainfall runoff modeling with a

https://doi.org/10.5194/egusphere-2024-3649
Preprint. Discussion started: 21 February 2025

combination of consistently superior performance and addressing the most pressing concerns regarding reliability, even though many questions remain to be answered. Due to their flexibility, they are primary candidates for entirely novel approaches that will become more relevant as climate change gives rise to questions of human influence and multi-factor disasters.

2.2.1 Spatially Resolved Processing

Smith and Eli (1995), the first study on neural networks in rainfall runoff modeling trained a simple, non-recurrent neural network on a five-by-five grid of synthetic data as a proof of concept. Another early example of semi-distributed processing is Hu et al. (2007). The authors evaluate the effect of lumping, but use only five rain gauging stations as input instead of a full grid, and a single catchment as a target. Xiang and Demir (2022), unfortunately not peer-reviewed, presents an architecture closely resembling the first stage of our model, which they call GNRRM-TS: Inputs are processed separately on a regular grid before being aggregated using a manually computed flow direction map. Here, too, the scope is limited to a single station and the only inputs are precipitation and drainage area of each grid cell. Xie et al. (2022) use LSTMs in a gridded fashion to estimate monthly baseflow instead of daily runoff. But similar to us, they also include static information as inputs and train their model on hand-selected subgroups of catchments. Muhebwa et al. (2024) propose a nuanced semi-distributed strategy, which instead of aggregating entire catchments, aggregates regions within a catchment that are similarly far upstream. The resulting set of input features for each region group are concatenated and jointly processed by a LSTM model. Hitokoto and Sakuraba (2020) is an interesting example of using an irregular vector-mesh rather than a regular grid. For each node, a conceptual model provides estimates of local runoff that are then aggregated by iteratively simplifying a mesh using a technique that is inspired by particle filters. Once coarsened to 96 nodes, they use a relatively simple four layer fully connected neural network for routing. When considering only the portion of the pipeline managed by the neural network—i.e., after the conceptual model's outputs have been coarsened—this approach can be classified as semi-distributed. Sun et al. (2022) also act on an irregular mesh, this time training a graph neural network on the outputs of a conceptual model. After this pre-training, they fine-tune the neural network on streamflow observations - an elegant way to deal with sparse empirical data in this context. However, their study is also limited to a single smaller basin in the western United States, and they have to rely on graph coarsening to scale this approach up to another, larger basin, as their model consists of complexly interleaved graph and time convolution layers. Yu et al. (2024) propose to apply a LSTM model on the catchment level, then use a conceptual model to route the predicted streamflows along the gauges in the river network.

2.2.2 Routing

Routing refers to modeling the flow of water between gauging stations in a river network. Neural networks have been successfully employed for this task as well. Within this context, streamflow at a given station is predicted from the streamflows of upstream stations alone, typically at an hourly resolution. At this temporal resolution, routing within the river system can ignore slower process like runoff generation or baseflow, and instead focus entirely on the movement of runoff along the river network. Since the stations within a river system can be conceptualized as nodes of a directed acyclic graph, it seems natural to model this data with a graph neural network, although this term is fairly broad (Bronstein et al., 2021). Example of this

https://doi.org/10.5194/egusphere-2024-3649
Preprint. Discussion started: 21 February 2025

approach include Moshe et al. (2020), Sit et al. (2021), Sun et al. (2021), Sun et al. (2022), Nevo et al. (2022) and Nearing et al. (2024) all of which demonstrated excellent performance in this setting. In comparison, the design of the routing stage in our model as detailed below is much more minimal in order to give the user more fine-grained control and interpretability.

Another line of research investigates models that act on networks of rain gauges instead of a regular grid of inputs. These models can be considered semi-distributed as well. The general focus here seems to lie more on finding suitable architectures for this task, combining self-attention, LSTM, convolution and more complex graph convolution layers. For example, Chen et al. (2023) intricately stack several LSTM layers to represent the river network structure, while Zhou et al. (2023) propose a mixture of self-attention, recurrency and convolution to build a graph neural network for this routing task. Zhu et al. (2023) aggregate remote-sensing rain data within each catchment in a data-driven fashion by training separate convolutional neural networks for every input product. They then concatenate each sub-basin's lumped information with rain gauge data for further routing in a semi-distributed scheme. The convolutional approach amounts to a more sophisticated form of lumping, as integrating the different data sources as well as temporal modeling happens at a later stage, once all spatial data has been fully aggregated. Hu et al. (2024) partially work on gridded data, namely remote-sensing measurements of rainfall. Each sub-basin is aggregated separately using a convolutional LSTM to produce a spatially aggregated timeseries of rainfall in the sub-basin, concatenate it with static information and recent runoff and continue processing this information in a semi-distributed fashion. The crucial difference here is that the convolutional LSTM serves as a data-driven aggregation mechanism for the gridded rainfall input data, but hydrological modeling again takes place in the semi-distributed domain.

2.2.3 CAMELS-type datasets

Apart from the studies which we just discussed, featuring individual or a few select catchments, large-scale rainfall runoff modeling with neural networks has extensively featured the CAMELS dataset, based on Newman et al. (2015) and extended to its current form by Addor et al. (2017). It contains meteorological time series and ancillary data for 671 catchments located within the contiguous United States, manually selected for minimal human impact. This implies that the catchments are relatively small, but on the other hand ensures "laboratory conditions" for hydrological modeling. The downside of this is the limited applicability of findings generated with this data to areas of the world where human influence contributes significantly to streamflow, such as central Europe. But as the dataset covers the contiguous US homogeneously, spans a large area, contains many catchments and a wide variety of different climates, it offers optimal conditions for training neural networks. And so CAMELS rose to popularity together with the neural network approach in rainfall runoff modeling. Since then, similar public datasets were introduced that cover other parts of the world: Chile (Alvarez-Garreton et al., 2018), Great Britain (Coxon et al., 2020), Brazil (Chagas et al., 2020), Australia (Fowler et al., 2021), the upper Danube basin (Klingler et al., 2021), France (Delaigue et al., 2022), Switzerland (Höge et al., 2023), Denmark (Liu et al., 2024) and Germany (Loritz et al., 2024).

2.3 Contributions

The first or *local* stage of our model, detailed in Subsection 3.5, largely follows the architecture presented in Kratzert et al. (2019b), but instead of using lumped basins as input, we apply a single neural network in parallel to a regular grid of meteoro-

https://doi.org/10.5194/egusphere-2024-3649
Preprint. Discussion started: 21 February 2025

logical and ancillary inputs. This novel kind of spatially resolved processing is enabled by modern GPUs with large memory. We show that this finer spatial resolution allows us to capture co-variances and provide regularization, benefiting especially larger catchments. This approach is natural for physical or conceptual models, which have to solve local equations at a certain level of rasterization. Yet, no one to our knowledge has applied a neural network directly to the grid of inputs in a way that scales up to entire river basins.

The right panel of Figure 6 visualizes the output of the local stage in an exemplary fashion. These local runoff quantities are then mapped onto a river network in the second or *routing* stage described in Subection 3.6. It consists of two simple network layers without any nonlinearity. We show that the river network connectivity graph can be used as inductive bias to constrain the model to reproduce the river's natural layout. This increases data efficiency and allows for better interpretability. Below we discuss that in principle, although this has yet to be shown, the model can be controlled interactively: Extracting or injecting quantities of water can simulate human influence such as industrial, agricultural or hydroelectric energy generation activity, which significantly contributes to streamflow but is independent of the modeled hydrological processes. Both stages are trained jointly in an end-to-end fashion on the entire dataset, rendering any kind of expert knowledge obsolete. This also means that the model is fitted exclusively on empirical data, enabling scientific discovery from raw data.

The lumped datasets discussed above are unsuited for our approach, as it requires non-aggregated information for spatial processing as well as entire basins for routing. Hence, for this study we compiled gridded meteorological and static data for five entire basins in central Europe, characterized by an overall high level of human activity, compared to the CAMELS dataset. The data is publicly available.

3 Data and Methods

As discussed above, previously released datasets for rainfall streamflow modeling are unsuited for spatially resolved processing, so we compiled a new, publicly available dataset, referenced in the data availability statement. We present the data sources, preprocessing steps and how to handle the files in more detail in a separate publication (Vischer et al., 2025, under review). The river discharge data that we use as targets for training and the catchment information from which we derive the river connectivity information is publicly available for download from the original provider. We provide code that processes and combines it with the input data after manual download. The following subsections give an overview of the data, followed by an introduction of our neural network.

3.1 Study Area, Study Period and Resolution

The focus of our study is on five river basins covering Germany and parts of neighboring countries: Elbe, Oder, Weser, Rhine as well as the upstream part of the Danube river up to Bratislava (see Figure 1, right panel). Due to the sparser coverage of gauging stations in the lower reaches of the Danube basin, we decided to focus on the upper reaches where the station network is more homogeneous. Additionally, the placement of river gauging stations varies across countries, as each follows distinct policies for station location. From a machine learning perspective, this results in diverse sampling strategies across the river network.

https://doi.org/10.5194/egusphere-2024-3649
Preprint. Discussion started: 21 February 2025

We restricted our study to this region to prevent variations in sampling distributions from affecting the assessment of routing performance. We decided to limit our study to this region so as not to confound the performance of routing with such different sample distributions. The total study area covers a contiguous 570.581 km^2 area of Germany and neighboring countries. Figure 1 visualizes exemplary features in the study area with boundaries of the river basins and Germany for geographic reference. Based on the consistent availability of streamflow data, we decided to conduct our experiments on the water years 1981-2011. A water year lasts from October 1st of the previous year to September 30th. Due to data availability, we homogenized all input data to daily temporal resolution and regular grid covering the earth's surface with a spatial resolution of $0.1° \times 0.1°$ or roughly $9km \times 9km$.

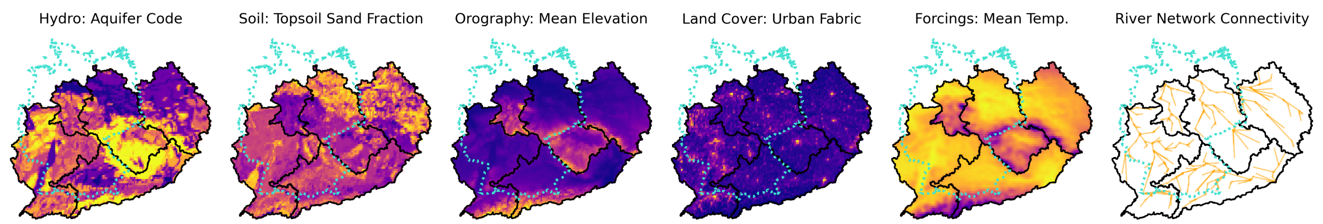

Figure 1. Overview of study area and visualizations for an example feature of each type. Basins are outlined in black, the boundaries of Germany are shown in turqoise. The right hand panel shows additional river network information as yellow arrows.

3.2 Dynamic Input Data

Runoff is primarily driven by precipitation, but to properly capture processes like evaporation or snow dynamics, temperature and solar radiation need to be taken into account as well. Our meteorological input variables, also called *forcings* are thus daily minimum, average and maximum temperature, daily sum and standard deviation of precipitation and average potential evaporation - a score computed from radiation, temperature, air pressure and humidity. This set of variables widely used in previous studies (Kratzert et al., 2018, 2019b) was retrieved from ERA5-Land (Muñoz Sabater, 2019). The variables, downloaded at three hour intervals, were aggregated to a daily time step to match the time resolution of the target. These time series were obtained by aggregating daily the values contained in the ERA5 database[2] (CopernicusClimateChangeService, 2022). We amend these six meteorological input dimensions with two more sine-cosine embeddings of day of the week and day of the year, which can be considered as a coarse proxy to human activity (Otero et al., 2023).

3.3 Static Input Data

Following the insights from Kratzert et al. (2019c) and Shalev et al. (2019), we include static data, also termed ancillary data, to enable genuinely data-driven and transferable models, trained jointly on all locations. Specifically, we include hydrogeo-

[2]The dataset was downloaded from the Copernicus Climate Change Service (2022). The results contain modified Copernicus Climate Change Service information 2020. Neither the European Commission nor ECMWF is responsible for any use that may be made of the Copernicus information or data it contains.

logical properties, soil class, land cover, and orographic features derived from a digital elevation map for a total of 46 feature dimensions. Please refer to (Vischer et al., 2025, under review) for a detailed description of the origin and preprocessing of all input features. Cheng et al. (2023) demonstrated that relevance propagation could help to streamline the model's inputs by identifying non-task relevant features.

3.4 Target Streamflow Time Series, Station Information and River Networks

Target time series of streamflow at each station were obtained from the Global Runoff Data Center (GRDC) data portal. Together with the streamflow data, the GRDC offers a catalog of station information. We considered all stations in the station catalog, but excluded stations that had ten or more values missing in the time series for the selected study period. Furthermore, initial experiments showed that including stations with less than 500 km^2 drainage area in training decreased performance, even when evaluating exclusively on larger stations. We decided to exclude these small catchments, and discuss this decision in Section 5. Another natural limitation on the spatial scope of this approach is that it only captures rainfall-streamflow dynamics in locations contained in the drainage area of a gauging station. In coastal areas, runoff might directly enter the sea through smaller streams that are not gauged. Hence, our study area usually starts several kilometers inland from the sea. The following number of stations resulted in each basin: 62 in upper Danube, 34 in Elbe, 36 in Oder, 78 in Rhine and 29 in Weser basins, for a total of 239 stations. For comparison, the CAMELS dataset contains 671 catchments. Further visualizations of the river networks can be inspected in the preprocessing scripts, along with all details on how the metadata was processed.

3.5 Local Stage (1) - Modeling Spatially Resolved Runoff

Figure 2 visualizes the simple network architecture used in the local stage to predict locally generated runoff on a regular grid. As mentioned before, following Kratzert et al. (2019b) we add static information to the meteorological inputs. This model was originally conceived to model aggregated catchments, but we adapt the design and apply it to all grid locations in parallel, which are then fed into a routing layer. Our pipeline features 46 static input dimensions, which we reduce to ten by adding a simple, fully connected embedding layer. The resulting static features are concatenated with another ten dimensions of meteorological forcings. The input dimensionality to the LSTM layer is thus 20 in our case, compared to 32 in the case of Kratzert et al. (2019b). Their LSTM layer consists of 256 units, ours of 250. However, we reduce the 250 output values of the LSTM layer by using two regression layers instead of one. We do not employ a nonlinearity after the second readout layer, meaning that the network's outputs are not confined to the range of e.g. $[-1,1]$, but rather live in the range of actual, physical quantities. The model itself does not have any predictive capabilities, but produces a forecast of streamflow if the inputs lie in the future, i.e. are taken from a meteorological forecast.

In an exploratory experiment, we compared concatenating static inputs to dynamic inputs before and after the LSTM layer. Feeding the static inputs through the LSTM together with the dynamic inputs resulted in substantially better performance. This is consistent with the findings of Kratzert et al. (2019b) and can be explained by the static inputs helping the LSTM to better adapt to the hydrological dynamics of a location. This is similar to training separate models for different climatic zones, but in a data-driven fashion. Indeed, Cheng et al. (2023) show that clustering on the relevance values of the different inputs

https://doi.org/10.5194/egusphere-2024-3649
Preprint. Discussion started: 21 February 2025

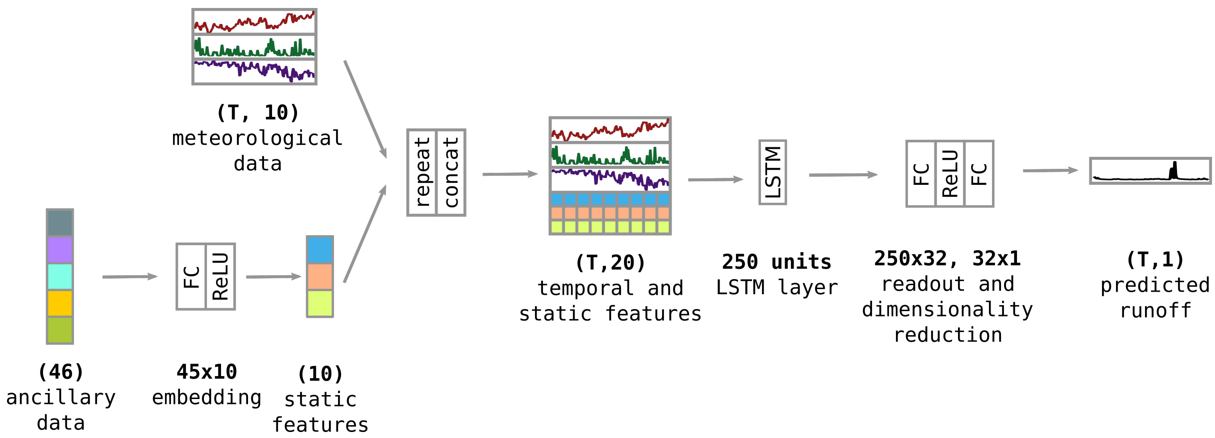

Figure 2. Overview of the local stage of the network architecture, acting on the regular grid input. Each grid cell's static inputs are reduced to ten features by feeding them through a simple, fully connected embedding layer and applying a nonlinearity function. The resulting feature vector is repeated for every time step $\mathbf{T}$ of the meteorological forcing time series and concatenated for a total of 20 features fed into the LSTM layer consisting of 250 parallel units. The recurrent layer's output is then reduced to the 1D output time series by sequentially applying two fully connected readout layers plus a nonlinearity in between. Numbers in parentheses signify feature vectors at a given stage in the model pipeline, numbers without signify the size of the weight matrix of a given layer.

results in hydrologically plausible clusters. Concatenating all features before feeding them through the LSTM layer requires more parameters in this layer, which due to its intricate inner workings is particularly expensive to train in terms of data and compute. We also used gated recurrent units (GRU) (Cho et al., 2014) instead of LSTMs as a backend, which mitigates this problem a little because they are computationally more efficient. We found that they do not affect performance, but decided to stick with LSTM as our main backbone as it is more popular in the literature. Nevertheless, using GRUs could be another way to further optimize the model.

3.6 Routing Stage (2) - Mapping Local Runoff to the River Nework and Routing

The task of the routing stage of our model is to map the locally generated runoff to a station's catchment area, and then routing the runoff along the river network to predict streamflow time series for every station in the basin. Figure 3 visualizes the layout. Within a given river basin, we concatenate the predicted runoff time series of all grid cells. The network learns a simple, strictly linear mapping consisting of two layers: First, a fully connected layer without nonlinearity maps all grid cells $\mathbf{G}$ to their respective stations $\mathbf{S}$. Since no nonlinearity is added, this layer can be translated into a weighted, time dependent average of all grid cells within a catchment. Location information from the station catalog, described below, can be used as an inductive bias to constrain this layer so as to only route water in a physically plausible way Then, a 1D-convolution layer (Kiranyaz et al., 2021) performs time convolution on each station's time series to combine information inside the river network over the last nine days. Separate kernel values can be learned for each day, but the same kernel is applied jointly over the entire time series. The value of nine days was chosen as a conservative estimate of the maximum time that water would be

https://doi.org/10.5194/egusphere-2024-3649
Preprint. Discussion started: 21 February 2025

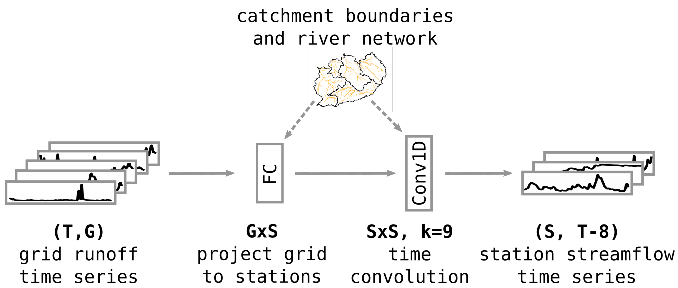

Figure 3. Runoff time series for every grid cell within a given basins are concatenated and fed through a fully connected layer that projects them down onto the number of stations. The weight matrix in this layer is constrained to be non-zero only when a grid cell lies in the catchment of a given station. The stations are then time-convolved with a kernel length of 9 time steps. Each kernel is also constrained to be only non-zero if a given station is directly upstream of another station. This stage yields time series for all stations in the network. It does not include any nonlinearity, so all activation values can be interpreted as streamflow quantities. The basin map and dotted arrows indicate that the river connectivity information serves as inductive bias on these two layers, constraining the activations to replicate the real river network.

running inside the river networks considered in this study, but of course awaits empirical validation and further optimization. The kernels in this layer are constrained by the connectivity of the river network in order to be physically plausible as we explain in the following subsection. Crucially, this stage does not involve any nonlinearity. Hence, both the fully connected as well as the time convolution layer are linear functions and as such can be chained to form yet another linear function. This means that the quantities of water "flowing" through this pipeline are physically interpretable, or put differently, that input quantities can be added or subtracted meaningfully to and from the input. A practical application example of this, which we plan to further investigate, is the injection or extraction of water in between two stations as a result of agricultural, industrial or hydroelectric human activity. This sets our approach apart from previous routing approaches. We also investigated the effect of not constraining the weight matrices with the connectivity matrices, which leads to slightly poorer performance when data is scarce (see Subsections 3.9 and 4.5).

3.7 Structural Bias

Among many other kinds of information, the station catalog contain polygons describing each station's catchment area. From this, we can derive two important kinds of information: first, for every grid cell we can determine in which station's catchment it is located; second, for each station, we can determine how it is connected to upstream and downstream stations.

3.7.1 Catchment Matrices

Mapping grid cells to stations is important to ensure that the runoff predicted at a given location ends up at the only station that is physically plausible. Since a given station's catchment area is contained within all the downstream stations' catchment areas, we need to make sure that we select the one where the generated runoff first enters the river network. To do so, we select the station with the smallest catchment area that contains a given grid cell. For each river network, we represent this information

https://doi.org/10.5194/egusphere-2024-3649
Preprint. Discussion started: 21 February 2025

conveniently in a one-hot matrix with grid cells as rows and stations as columns. This matrix is then used to constrain the fully connected layer in the routing stage of the network. This is achieved by multiplying the one-hot matrix point-wise with the freshly initialized weight matrix of this layer before training begins. Weights corresponding to physically impossible connections are set to zero from the start. Zero weights can not contribute to gradients, and will remain zero throughout the training. All other weights can be freely optimized.

3.7.2 Connectivity Matrices

From the catchment area polygons, a graph representing the connectivity between stations in the river network can be derived. Each node in the graph represents a station, a directed edge exists between a station A and B if A is directly upstream of B. We determine this by verifying if the catchment area of A is contained in the catchment area of B, ensuring that there are no intermediary stations in between, i.e. contained by B and containing A. Note that this automatically leads to a directed, acyclic graph. This fits our approach well, as it does not require us to apply any model of routing iteratively in order to capture cyclic movements within the graph. The graph is represented by connectivity matrix, i.e. a square matrix with rows (input) and columns (output) corresponding to stations, where the entries are 1 if a directed edge exists and 0 otherwise. This matrix is used to constrain the time convolution layer in a manner similar to the catchment matrix. After initialization of the weight matrix, the connectivity matrix is multiplied point-wise, preserving the weights where a connection exists and setting them to zero where no connection is present. The only difference is that the connectivity matrix needs to be repeated by the depth of the temporal convolution, nine times in our case for the nine days of past information that we convolve.

3.8 Training and Metrics

We split the data into three parts, all containing entire water years: A training set from water years 1981 to 2005, a validation set for model selection from 2005 to 2008, and a test set to report the final performance from 2008 to 2011. We also created two special training datasets to illustrate how the models perform on less training data: A medium length training set comprising ranging from 1991 to 2005 and a short training set from 1999 to 2005. Regardless of the length of the training set, we divide it into chunks of 400 days that partially overlap. The first 30 days are used as a warm up period for the LSTM. During this time no gradients are computed and the LSTM can stabilize into an operating regime before starting the learning process. The value of 400 was chosen to accommodate an entire year plus the warm up period. It would not be detrimental to use even longer time series, but this cutoff is dictated by GPU memory space. When calculating scores, no gradients are computed, resulting in a significantly smaller memory footprint. This allows us to calculate all scores on uninterrupted time series in a single model forward pass. In all experiments, we trained for 2000 epochs of the training data. This number of epochs is generous for all models to converge. Since the purpose of this study is to provide a proof of concept of this type of spatially resolved processing, we decided to not conduct an extensive hyperparameter optimization or perform input feature ablation. Instead, we included all potentially relevant static data and ran hyperparameter tuning experiments only on a limited set of values for a few key hyperparameters, summarized in Table A2. We trained ten different random seeds for every setting, and report the best seed in terms of station-wise median Nash-Sutcliffe Efficiency (NSE) score on the validation period.

https://doi.org/10.5194/egusphere-2024-3649
Preprint. Discussion started: 21 February 2025

We use the widespread NSE metric (Nash and Sutcliffe, 1970) both as a loss function in training as well as a score to quantify performance. The NSE normalizes the square loss of each station by the standard deviation of the station's values in the training period, so as to count each station equitably towards the loss or performance, regardless of the magnitude of the river at this point. We do not split training and test partitions geographically, as breaking up basins would make routing impossible. We also refer to Klotz et al. (2024) for a word of caution when combining NSE values that are calculated on partitions of a dataset. Unless noted otherwise, the scores we report were calculated on the test period. The median NSE over all stations serves as a robust point estimate of performance, but for the interested reader we provide mean NSE scores in Table A1, which we found to correlate strongly with the median NSE for all seeds. Likewise, we report the percentage of stations with a NSE score below zero, which indicates predictive performance worse than simply using the average value of a station's runoff for prediction, i.e. chance level. The popular Kling-Gupta efficiency (KGE) metric (Gupta et al., 2009) was developed in the context of univariate, convex optimization - both these assumptions do not hold in the case of training a neural network. However for the sake of comparability, we also report the KGE values in the appendix as well.

As stated before, in this study we refrained from extensive hyperparameter optimization to maximize the performance. A few exploratory experiments to calibrate our pipeline seemed necessary nonetheless: Dropout (0, 10%, 30%, 50% separately in recurrent and readout layers of the routing stage) did not increase performance, so we removed it entirely. 250 units in the LSTM layer (out of 150, 200, 250 and 300) yielded the best results. We use an automatic learning rate scheduler, the ReduceLROnPlateau scheduler provided by Pytorch (Ansel et al., 2024) with threshold 1e-3 and patience 10, so the pipeline trains robustly with regard to the initial learning rate (1e-4, 5e-4, 1e-5). But as the baselines in the experiments have vastly different number of parameters, we decided to continue the experiments by always trying out both of the lower values. Unless explicity mentioned, we report the performance resulting from the better value for every condition, table A2 lists the results in detail.

3.9 Baselines

We introduce two baselines in order to evaluate the performance gains of the two central aspects of our pipeline: spatially resolved processing in the local stage and the inclusion of structural bias in the routing stage. In the first baseline experiment, we aggregate all spatial information within a catchment and feed it through the same architecture as used in the local stage. Normally, this stage of the model processes individual grid cells, but here it processes entire, aggregated catchments, and the output is a prediction of the runoff measured at the corresponding station. This aggregated baseline does not require any further routing and is virtually identical to the approach in Kratzert et al. (2019b). We will use this as baseline and coin it *aggregated*, whereas our default model is referred to as *spatially resolved*.

The second baseline leaves the local stage unaltered, while in the routing stage the weight matrices are not constrained. Instead, we use two fully connected time convolution layers with a kernel size of nine and three days, respectively, and a nonlinearity in between, as simple and more conventional neural network approach. We will refer to this baseline as *naive routing*, and to our default model as *structured routing*.

https://doi.org/10.5194/egusphere-2024-3649
Preprint. Discussion started: 21 February 2025

4 Results and Discussion

We start this section by presenting and contextualizing the general performance of our model. We then show that our model excels in modeling large catchments, is less prone to overfitting and learns from the data more efficiently than the baselines. Inductive bias does not make a big difference to performance. We end this section by showcasing that our model is, unlike most neural networks, not an entirely black box model, and that capturing human influence seems to be the biggest challenge in our study area.

4.1 Model Performance

Our model reaches a median NSE performance on the test dataset of 0.77. This is on par with the median NSE of 0.74 reported by Kratzert et al. (2019b) and 0.73 reported by Shalev et al. (2019), both on the CAMELS dataset. Training our model on a spatially extended version of CAMELS would allow for more direct comparability with other approaches, but this has to be left for future research. The aggregated baseline only achieves a median NSE of 0.69. This is a first indicator that our spatially resolved modeling pipeline is able to compensate our dataset being smaller than CAMELS and probably containing a larger human signal. Despite the data being different and not allowing for a straightforward comparison, this shows that runoff generation and routing can be learned end-to-end by a single model pipeline and without additional data along the way. Moreover, the simplicity of the routing module's design, along with the possibilities it offers, does not come at a significant performance cost. An example in case is that the internal activations inside the network between local and routing stage appear to be hydrologically plausible, as Figure 6 illustrates, despite not enforcing this property during training.

4.2 Modeling Large Catchments

Figure 4 compares our model and the aggregation baseline, depending on the catchment area and separately for training and testing periods. We focus on the test results first (right panel): For aggregated processing, we observe a negative trend between performance and catchment size, which does not exist for spatially resolved processing. This is expected, as the benefits of spatially resolved processing are more pronounced for larger catchments, which tend to be more heterogeneous. Our results suggest that spatially resolved processing should be considered in such situations.

4.3 Inherent Regularization

A comparison between the left and right panel of Figure 4 reveals a substantial performance drop between train and test period. To a certain degree, this is expected, but the effect is much more pronounced for the aggregated than for the spatially resolved model. It stands to reason that the neural network in the aggregated baseline with its associated reduction of data severely overfits. This also becomes apparent when looking at the median NSE values in training and test datasets. While overall performance drops from 0.86 to 0.69 in the case of aggregated processing, spatially resolved processing deteriorates more gracefully from 0.90 to 0.77. Spatially resolved processing, with its shared local stage and overall much more data, seems to have an intrinsic regularization effect.

https://doi.org/10.5194/egusphere-2024-3649
Preprint. Discussion started: 21 February 2025

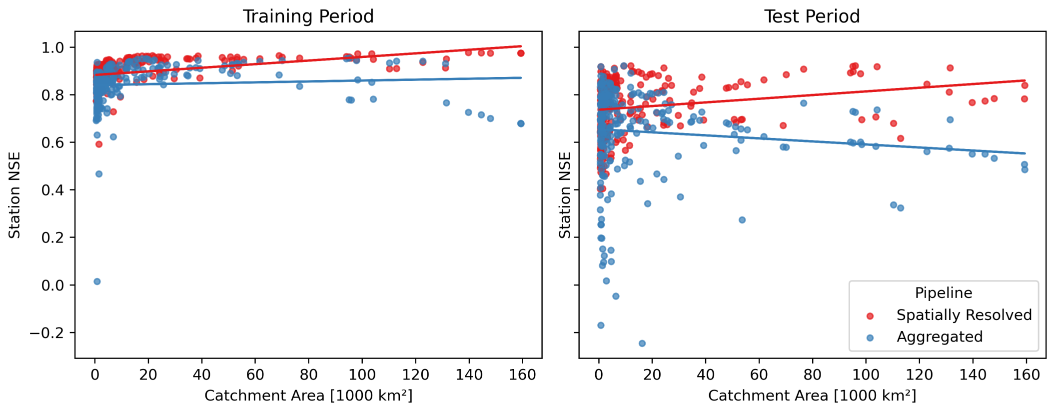

Figure 4. Station-wise performance vs. size of catchment, for aggregated baseline (blue) and spatially resolved pipeline (red) on training (left panel) and test (right panel) period. Aggregated processing impairs performance, especially in larger catchments. The trend is more pronounced on the test dataset, indicating overfitting on the training dataset. Spatially resolved processing is less prone to overfitting and manages to handle large catchments accurately in a low data setting.

4.4 Data Efficiency

The positive effect of spatially resolved training, especially for large catchments, becomes even more pronounced when looking at modeling in an environment with limited available training data. Figure 5 visualizes the difference in NSE score between spatial and aggregated processing on a per-catchment level, when training on 25, 15 and six years of training data. While differences are positive across all sizes of catchments, meaning spatial processing on average performs better regardless of the catchment size, the positive trend becomes stronger as data becomes more scarce. Spatially resolved processing utilizes the available data more efficiently.

4.5 Inductive Bias

Including inductive bias for what we call structured routing leads to slightly better performance than naive routing without the additional real-world information, with a median NSE of 0.77 compared to 0.72. Figure A1 contains more detailed results, but also shows that the performance gain is small. The point we want to make here is that the practical benefit of being able to simulate the injection or extraction of quantities of water in routing process does not come at the cost of lowering performance. As we mentioned before, naive routing on the other hand does remain an important tool in modeling basins where catchment delineation information is unavailable or unreliable, or where lateral transport of water inside the bedrock layer across catchment boundaries is suspected. Whether or not we use inductive bias in the routing layer, our networks are extremely simple compared to other networks proposed in the literature for routing that we discusses above. Certainly, we demonstrate that routing modules do not need to be complex, and river network extraction algorithms are not necessary for end-to-end routing, e.g. when no catchment boundary information is available.

https://doi.org/10.5194/egusphere-2024-3649
Preprint. Discussion started: 21 February 2025

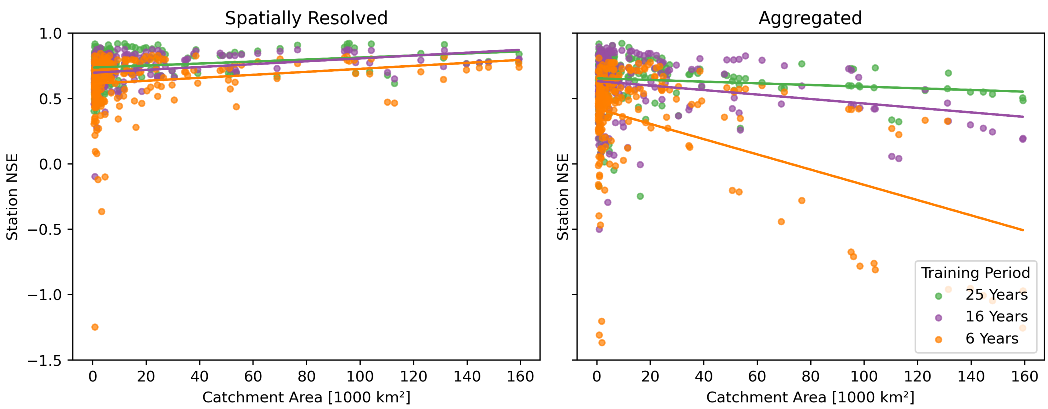

Figure 5. Station-wise differences in NSE performance between spatial and aggregated training, plotted against the size of the station's catchment. Positive values indicate that spatial processing performs better. The individual data points are fitted linearly. Colors correspond to 25 years of training data (green), 15 years (purple), and 6 years (orange). Differences are positive across all catchment sizes and different amount of training data. But as data becomes scarcer, the trend of spatial processing outperforming aggregated processing in large catchments becomes more pronounced.

4.6 Interpretable Internal States

The simple architecture of our model, particularly the shared recurrent layer in the local stage and inductive bias and linearity of the routing stage, lead to interpretable internal states within the network. As an illustrating example, Figure 6 displays the activation values after the local stage and before the routing stage for two exemplary days in spring and summer. The spatio-temporal correlation seems to suggest that in the example day in spring, runoff is primarily driven by snow melt in low mountain ranges, whereas in summer, it seems to be driven by heavy precipitation events. We want to make the point that these activations are hydrologically plausible, yet we did not enforce this property during the training process, e.g. by providing additional target information or training a special readout layer. It is a purely data-driven, emergent behavior, resulting from both the end-to-end training process and the model's parsimonious design. Unlike more complex neural network designs which are generally considered black box models, this suggests that our model naturally allows for a certain degree of internal control by manipulating the internal states (e.g. subtracting or adding quantities of water to the natural runoff), as well as enabling further scientific discovery from large quantities of data.

4.7 On Human Influence

We conclude this section by discussing a specific negative outlier in terms of station-wise NSE. As explained above, a negative NSE values indicates performance below chance level. The only two stations that yield negative NSE values after training - consistently across all ten random seeds - are Spremberg and Boxberg (GRDC numbers 6340800 and 6340810) located along

[3]The country boundary information was downloaded from simplemaps.

https://doi.org/10.5194/egusphere-2024-3649
Preprint. Discussion started: 21 February 2025

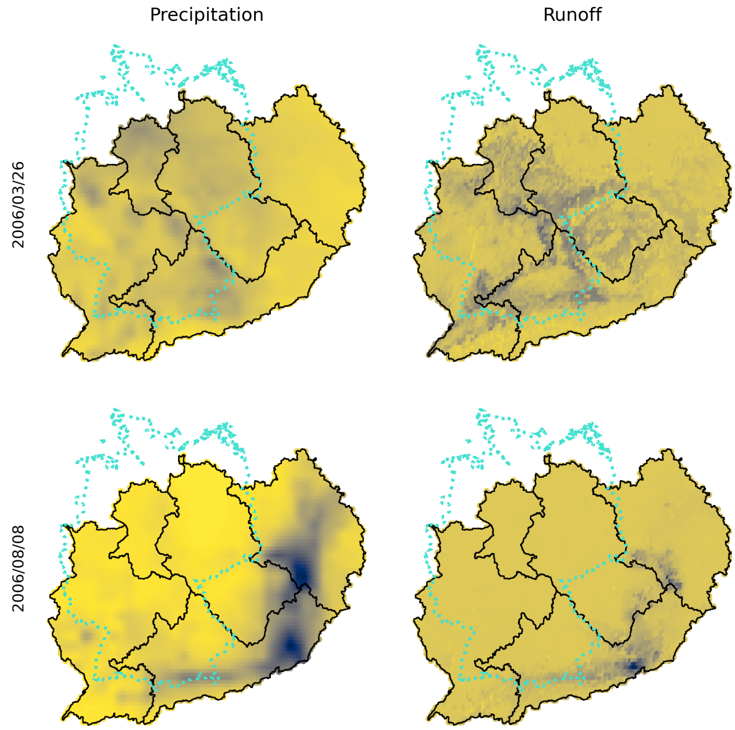

Figure 6. Input feature precipitation (left column) and activation values after the local stage of the model (right column) for two days in March (top row) and August 2006 (bottom row), manually selected for illustration purposes. Values are displayed in arbitrary units, with blue signifying more runoff or precipitation. We included the outline of Germany [3]in dotted lines for geographic reference. For the day in march (top row), a low spatial correlation between precipitation and runoff together with a pattern of high runoff values in low mountain ranges suggests that runoff on this day is primarily driven by snow melt in medium-high altitudes. For the day in August on the other hand, we see runoff that is driven by two clusters of heavy precipitation in the East of our study area.

small rivers next to large surface mining operations in the Lusatia region. A potential explanation for these extreme outliers is that those mining operations have an influence on the overall water balance that is relatively large compared to the hydrological processes in such catchments. This seems to support our assumption that human influence is one of the main obstacles to be overcome by rainfall streamflow models in the densely populated areas of central Europe. We want to emphasize that the two stations in question only performed very poorly in the validation period, but not in the test period. We hypothesize that operations might have changed in the meantime or the test period simply lacks substantial events of human influence by chance. We plan to investigate this phenomenon more closely and explore potential solutions in future work.

https://doi.org/10.5194/egusphere-2024-3649
Preprint. Discussion started: 21 February 2025

5 Summary and Future Directions

We have successfully trained a neural network in an end-to-end fashion to capture runoff generation in a spatially resolved manner and on a large scale and shown the advantage of this approach especially in large catchments. We have also managed to simultaneously train a very simple neural network to perform routing in five river basins in central Europe. Not only does this approach mitigate overfitting and increase data efficiency, but the simplicity of the design and the ability to integrate inductive bias opens new possibilities to control the inner workings of the model, which is unusual for neural networks.

Our model reaches a level of performance comparable to that of other benchmark models, both conceptual and statistical. In future work, we plan to compare our model to other state-of-the art models that are used in science and operations. The former requires a spatially resolved version of the popular CAMELS dataset to allow for direct comparability with much of the neural networks literature. A direct comparison of our overall pipeline to that of an operational system like LISFLOOD would be similarly interesting. Another question that needs to be addressed in the future is how the performance of our model decreases with the forecast horizon. As we discussed before, our model requires a suitable meteorological forecast as input to generate a forecast of streamflow quantities, and the quality of the model's predictions depends on the quality of the forecast meteorological input. Quantifying this effect is important for real-world applications.

In this study, we excluded catchments where we suspected human influence is too strong based on a simple catchment area heuristic. Unlike much research in this area, we do not train our model exclusively on catchments with little human influence. But a more sophisticated strategy, inspired e.g. by Loritz et al. (2024) or Tursun et al. (2024) is needed to identify catchments with a large human footprint, and thus being able to properly disentangle the effects of catchment area and human influence.

Another big limitation in terms of data availability is the temporal and spatial resolution: The relatively small size of the recurrent layer enables our model to process time series at a higher than daily temporal resolution. Because the local stage, where most computation happens, is applied in parallel to all input locations, the number of parameters is independent of the number of inputs, and computational demand grows linearly with the number of locations. The weight matrices in the routing stage grow quadratically, but are much smaller in the first place.

Another important aspect that we will address in future work is discussed in Klotz et al. (2022), where the authors extend a model similar to ours by using the outputs as parameters of a distribution. Such distributional predictions could be obtained from our model in the same way, which is a relevant feature for many real-world applications. Producing genuinely probabilistic forecasts and warnings in this fashion is theoretically more sound than training an ensemble of more or less different models and combining their predictions.

As would be expected from the high degree of human activity in our study area, we found evidence that the effect of human influence is the central obstacle to further improving model performance in such an environment. Yet, a comprehensive investigation of the extent and impact of this phenomenon is still required. Future research could demonstrate that our neural network is capable of incorporating simulated human activity, such as water extraction or diversion, into the modeling of hydrological processes.

https://doi.org/10.5194/egusphere-2024-3649
Preprint. Discussion started: 21 February 2025

Code and data availability. The data used in this study is publicly available under CC BY-NC-SA license at hydroshare, the code used to preprocess it is available under Clear BSD license at our repository.

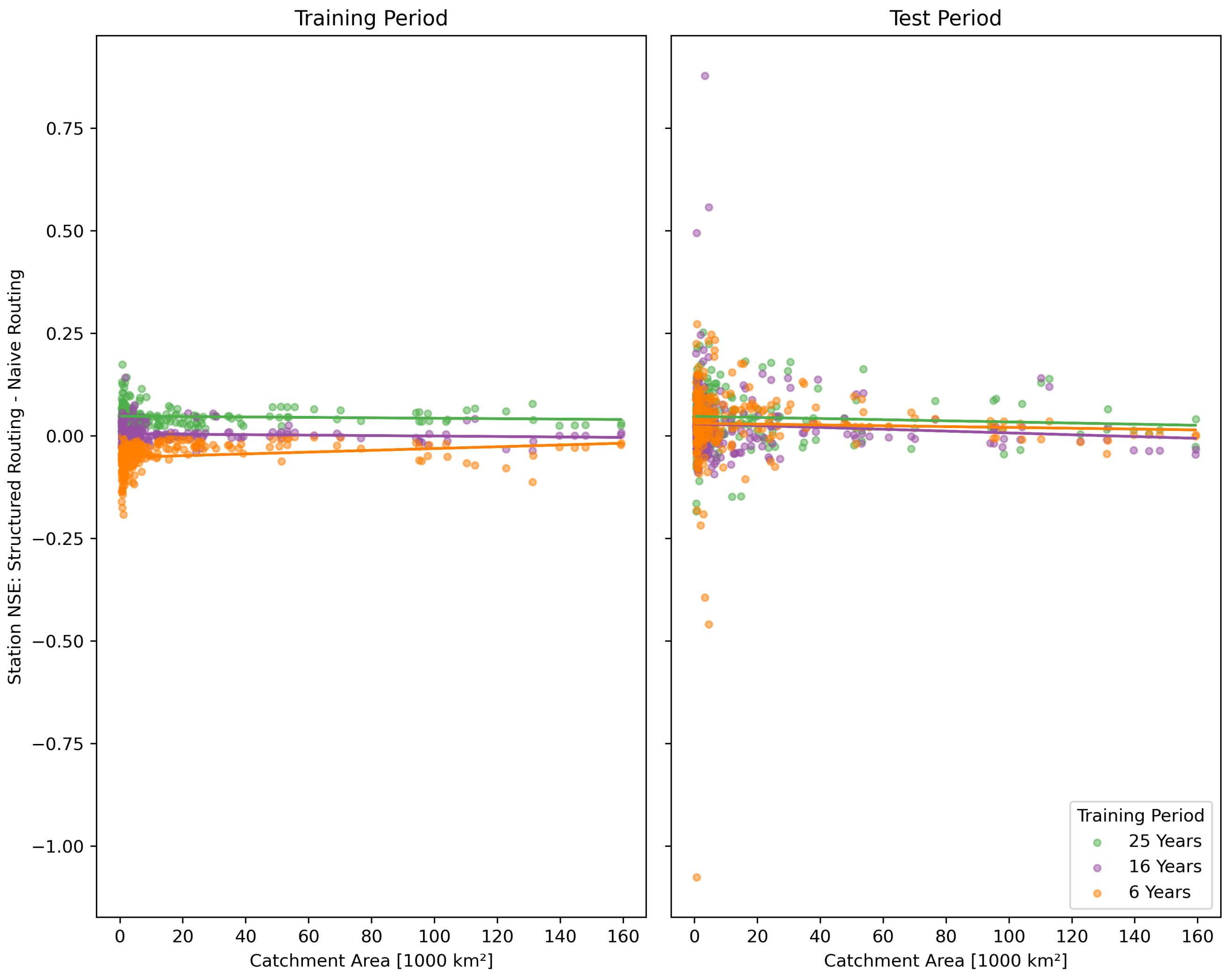

Figure A1. Station-wise differences in NSE performance between structured and naive routing, plotted against the size of the station's catchment on training (left panel) and test dataset (right panel). Positive values indicate that structured routing performs better. The individual data points are fitted linearly. Structured routing marginally outperforms naive routing on the test dataset (indicated by values greater zero). On the training dataset, naive routing performs better when data is scarce (values below zero for short training period in green), indicating overfitting.

https://doi.org/10.5194/egusphere-2024-3649
Preprint. Discussion started: 21 February 2025

Input Processing	Routing	Train Period	median NSE	mean NSE	NSE < 0	median KGE	mean KGE
spatially res.	structured	long	0.773	0.751	0.000	−0.077	−0.092
spatially res.	naive	long	0.719	0.706	0.000	−0.022	−0.035
aggregated	-	long	0.691	0.643	0.013	−0.065	−0.078
spatially res.	structured	med.	0.739	0.717	0.004	−0.051	−0.062
spatially res.	naive	med.	0.735	0.692	0.008	−0.052	−0.055
aggregated	-	med.	0.642	0.603	0.013	−0.072	−0.097
spatially res.	structured	short	0.687	0.633	0.017	−0.004	−0.030
spatially res.	naive	short	0.653	0.605	0.004	−0.023	−0.033
aggregated	-	short	0.485	0.318	0.126	−0.056	−0.138

Table A1. Various performance metrics for the experiments presented in section 4. The metrics were calculated after picking the best out of ten random seeds for each condition in terms of median NSE.

Input Processing	Routing	initial LR	median NSE
spatially res.	structured	5e-4	0.738
spatially res.	naive	5e-4	0.724
aggregated	-	5e-4	0.691
spatially res.	structured	1e-3	0.745
spatially res.	naive	1e-3	0.721
aggregated	-	1e-3	0.704

Table A2. Median NSE over 20 seeds for exploratory experiments on the optimal initial learning rate. Performance was evaluated on the *validation* period, as this is part of the model selection process.

https://doi.org/10.5194/egusphere-2024-3649
Preprint. Discussion started: 21 February 2025

Author contributions. M.A.V. and J.M. designed the experiments with crucial suggestions from N.O. M.A.V. prepared the data, implemented the model, performed experiments, and analyzed the results. M.A.V. prepared the manuscript with significant contributions from J.M and N.O. All authors reviewed the manuscript.

Competing interests. The authors declare no competing interests.

Acknowledgements. This work was supported by the Federal Ministry for Economic Affairs and Climate Action (BMWK) as grant DAKI-FWS (01MK21009A).

References

Addor, N., Newman, A. J., Mizukami, N., and Clark, M. P.: The CAMELS Data Set: Catchment Attributes and Meteorology for Large-Sample Studies, Hydrol. Earth Syst. Sci., p. 21, 2017.

Alvarez-Garreton, C., Mendoza, P. A., Boisier, J. P., Addor, N., Galleguillos, M., Zambrano-Bigiarini, M., Lara, A., Puelma, C., Cortes, G., Garreaud, R., McPhee, J., and Ayala, A.: The CAMELS-CL Dataset: Catchment Attributes and Meteorology for Large Sample Studies – Chile Dataset, Hydrology and Earth System Sciences, 22, 5817–5846, https://doi.org/10.5194/hess-22-5817-2018, 2018.

Ansel, J., Yang, E., He, H., Gimelshein, N., Jain, A., Voznesensky, M., Bao, B., Bell, P., Berard, D., Burovski, E., Chauhan, G., Chourdia, A., Constable, W., Desmaison, A., DeVito, Z., Ellison, E., Feng, W., Gong, J., Gschwind, M., Hirsh, B., Huang, S., Kalambarkar, K., Kirsch, L., Lazos, M., Lezcano, M., Liang, Y., Liang, J., Lu, Y., Luk, C. K., Maher, B., Pan, Y., Puhrsch, C., Reso, M., Saroufim, M., Siraichi, M. Y., Suk, H., Zhang, S., Suo, M., Tillet, P., Zhao, X., Wang, E., Zhou, K., Zou, R., Wang, X., Mathews, A., Wen, W., Chanan, G., Wu, P., and Chintala, S.: PyTorch 2: Faster Machine Learning Through Dynamic Python Bytecode Transformation and Graph Compilation, in: Proceedings of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2, pp. 929–947, ACM, La Jolla CA USA, ISBN 9798400703850, https://doi.org/10.1145/3620665.3640366, 2024.

Bevacqua, E., Shepherd, T. G., Watson, P. A. G., Sparrow, S., Wallom, D., and Mitchell, D.: Larger Spatial Footprint of Wintertime Total Precipitation Extremes in a Warmer Climate, Geophysical Research Letters, 48, e2020GL091 990, https://doi.org/10.1029/2020GL091990, 2021.

Beven, K. J.: Rainfall-Runoff Modelling: The Primer, Wiley-Blackwell, Chichester, West Sussex ; Hoboken, NJ, 2nd ed edn., ISBN 978-0-470-71459-1, 2012.

Bronstein, M. M., Bruna, J., Cohen, T., and Veličković, P.: Geometric Deep Learning: Grids, Groups, Graphs, Geodesics, and Gauges, arXiv:2104.13478 [cs, stat], 2021.

Chagas, V. B. P., Chaffe, P. L. B., Addor, N., Fan, F. M., Fleischmann, A. S., Paiva, R. C. D., and Siqueira, V. A.: CAMELS-BR: Hydrometeorological Time Series and Landscape Attributes for 897 Catchments in Brazil, Earth System Science Data, 12, 2075–2096, https://doi.org/10.5194/essd-12-2075-2020, 2020.

Chen, Z., Lin, H., and Shen, G.: TreeLSTM: A Spatiotemporal Machine Learning Model for Rainfall-Runoff Estimation, Journal of Hydrology: Regional Studies, 48, 101 474, https://doi.org/10.1016/j.ejrh.2023.101474, 2023.

Cheng, X., Vischer, M., Schellin, Z., Arras, L., Kuglitsch, M. M., Samek, W., and Ma, J.: Explainability in GeoAI, in: Handbook of Geospatial Artificial Intelligence, CRC Press, ISBN 978-1-00-330842-3, 2023.

Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y.: Learning Phrase Representations Using RNN Encoder-Decoder for Statistical Machine Translation, 2014.

Clark, S. R., Lerat, J., Perraud, J.-M., and Fitch, P.: Deep Learning for Monthly Rainfall–Runoff Modelling: A Large-Sample Comparison with Conceptual Models across Australia, Hydrology and Earth System Sciences, 28, 1191–1213, https://doi.org/10.5194/hess-28-1191-2024, 2024.

CopernicusClimateChangeService: ERA5-Land Hourly Data from 1950 to Present. Copernicus Climate Change Service (C3S) Climate Data Store (CDS), DOI: 10.24381/Cds.E2161bac (Accessed on 23-Oct-2021), 2022.

Coxon, G., Addor, N., Bloomfield, J. P., Freer, J., Fry, M., Hannaford, J., Howden, N. J. K., Lane, R., Lewis, M., Robinson, E. L., Wagener, T., and Woods, R.: CAMELS-GB: Hydrometeorological Time Series and Landscape Attributes for 671 Catchments in Great Britain, Earth System Science Data, 12, 2459–2483, https://doi.org/10.5194/essd-12-2459-2020, 2020.

https://doi.org/10.5194/egusphere-2024-3649
Preprint. Discussion started: 21 February 2025

Delaigue, O., Brigode, P., Andréassian, V., Perrin, C., Etchevers, P., Soubeyroux, J.-M., Janet, B., and Addor, N.: CAMELS-FR: A Large Sample Hydroclimatic Dataset for France to Explore Hydrological Diversity and Support Model Benchmarking, Other, display, https://doi.org/10.5194/iahs2022-521, 2022.

Fang, B., Bevacqua, E., Rakovec, O., and Zscheischler, J.: An Increase in the Spatial Extent of European Floods over the Last 70 Years, Hydrology and Earth System Sciences, 28, 3755–3775, https://doi.org/10.5194/hess-28-3755-2024, 2024.

Fowler, K. J. A., Acharya, S. C., Addor, N., Chou, C., and Peel, M. C.: CAMELS-AUS: Hydrometeorological Time Series and Landscape Attributes for 222 Catchments in Australia, Earth System Science Data, 13, 3847–3867, https://doi.org/10.5194/essd-13-3847-2021, 2021.

Gründemann, G. J., van de Giesen, N., Brunner, L., and van der Ent, R.: Rarest Rainfall Events Will See the Greatest Relative Increase in Magnitude under Future Climate Change, Communications Earth & Environment, 3, 1–9, https://doi.org/10.1038/s43247-022-00558-8, 2022.

Gupta, H. V., Kling, H., Yilmaz, K. K., and Martinez, G. F.: Decomposition of the Mean Squared Error and NSE Performance Criteria: Implications for Improving Hydrological Modelling, Journal of Hydrology, 377, 80–91, https://doi.org/10.1016/j.jhydrol.2009.08.003, 2009.

Hitokoto, M. and Sakuraba, M.: Hybrid Deep Neural Network and Distributed Rainfall-Runoff Model for Real-Time River-Stage Prediction, Journal of JSCE, 8, 46–58, https://doi.org/10.2208/journalofjsce.8.1_46, 2020.

Hochreiter, S. and Schmidhuber, J.: Long Short-term Memory, Neural Computation MIT-Press, 9, 1735–1780, 1997.

Höge, M., Kauzlaric, M., Siber, R., Schönenberger, U., Horton, P., Schwanbeck, J., Floriancic, M. G., Viviroli, D., Wilhelm, S., Sikorska-Senoner, A. E., Addor, N., Brunner, M., Pool, S., Zappa, M., and Fenicia, F.: CAMELS-CH: Hydro-Meteorological Time Series and Landscape Attributes for 331 Catchments in Hydrologic Switzerland, Earth System Science Data, 15, 5755–5784, https://doi.org/10.5194/essd-15-5755-2023, 2023.

Horton, P., Schaefli, B., and Kauzlaric, M.: Why Do We Have so Many Different Hydrological Models? A Review Based on the Case of Switzerland, WIREs Water, 9, e1574, https://doi.org/10.1002/wat2.1574, 2022.

Hu, F., Yang, Q., Yang, J., Luo, Z., Shao, J., and Wang, G.: Incorporating Multiple Grid-Based Data in CNN-LSTM Hybrid Model for Daily Runoff Prediction in the Source Region of the Yellow River Basin, Journal of Hydrology: Regional Studies, 51, 101 652, https://doi.org/10.1016/j.ejrh.2023.101652, 2024.

Hu, T., Wu, F., and Zhang, X.: Rainfall–Runoff Modeling Using Principal Component Analysis and Neural Network, Hydrology Research, 38, 235–248, https://doi.org/10.2166/nh.2007.010, 2007.

Jiang, S., Bevacqua, E., and Zscheischler, J.: River Flooding Mechanisms and Their Changes in Europe Revealed by Explainable Machine Learning, Hydrology and Earth System Sciences, 26, 6339–6359, https://doi.org/10.5194/hess-26-6339-2022, 2022.

Kam, P. M., Aznar-Siguan, G., Schewe, J., Milano, L., Ginnetti, J., Willner, S., McCaughey, J. W., and Bresch, D. N.: Global Warming and Population Change Both Heighten Future Risk of Human Displacement Due to River Floods, Environmental Research Letters, 16, 044 026, https://doi.org/10.1088/1748-9326/abd26c, 2021.

Kiranyaz, S., Avci, O., Abdeljaber, O., Ince, T., Gabbouj, M., and Inman, D. J.: 1D Convolutional Neural Networks and Applications: A Survey, Mechanical Systems and Signal Processing, 151, 107 398, https://doi.org/10.1016/j.ymssp.2020.107398, 2021.

Klingler, C., Schulz, K., and Herrnegger, M.: LamaH-CE: LArge-SaMple DAta for Hydrology and Environmental Sciences for Central Europe, Earth System Science Data, 13, 4529–4565, https://doi.org/10.5194/essd-13-4529-2021, 2021.

Klotz, D., Kratzert, F., Gauch, M., Keefe Sampson, A., Brandstetter, J., Klambauer, G., Hochreiter, S., and Nearing, G.: Uncertainty Estimation with Deep Learning for Rainfall–Runoff Modeling, Hydrology and Earth System Sciences, 26, 1673–1693, https://doi.org/10.5194/hess-26-1673-2022, 2022.

Klotz, D., Gauch, M., Kratzert, F., Nearing, G., and Zscheischler, J.: Technical Note: The Divide and Measure Nonconformity, Hydrology and Earth System Sciences Discussions, pp. 1–25, https://doi.org/10.5194/hess-2024-59, 2024.

Kratzert, F., Klotz, D., Brenner, C., Schulz, K., and Herrnegger, M.: Rainfall–Runoff Modelling Using Long Short-Term Memory (LSTM) Networks, Hydrology and Earth System Sciences, 22, 6005–6022, https://doi.org/10.5194/hess-22-6005-2018, 2018.

Kratzert, F., Herrnegger, M., Klotz, D., Hochreiter, S., and Klambauer, G.: NeuralHydrology – Interpreting LSTMs in Hydrology, in: Explainable AI: Interpreting, Explaining and Visualizing Deep Learning, edited by Samek, W., Montavon, G., Vedaldi, A., Hansen, L. K., and Müller, K.-R., Lecture Notes in Computer Science, pp. 347–362, Springer International Publishing, Cham, ISBN 978-3-030-28954-6, https://doi.org/10.1007/978-3-030-28954-6_19, 2019a.

Kratzert, F., Klotz, D., Herrnegger, M., Sampson, A. K., Hochreiter, S., and Nearing, G. S.: Toward Improved Predictions in Ungauged Basins: Exploiting the Power of Machine Learning, Water Resources Research, 55, 11 344–11 354, https://doi.org/10.1029/2019WR026065, 2019b.

Kratzert, F., Klotz, D., Shalev, G., Klambauer, G., Hochreiter, S., and Nearing, G.: Towards Learning Universal, Regional, and Local Hydrological Behaviors via Machine Learning Applied to Large-Sample Datasets, Hydrology and Earth System Sciences, 23, 5089–5110, https://doi.org/10.5194/hess-23-5089-2019, 2019c.

Kratzert, F., Gauch, M., Klotz, D., and Nearing, G.: Never Train a Long Short-Term Memory (LSTM) Network on a Single Basin, Hydrology and Earth System Sciences, 28, 4187–4201, https://doi.org/10.5194/hess-28-4187-2024, 2024.

Lees, T., Buechel, M., Anderson, B., Slater, L., Reece, S., Coxon, G., and Dadson, S. J.: Benchmarking Data-Driven Rainfall–Runoff Models in Great Britain: A Comparison of Long Short-Term Memory (LSTM)-Based Models with Four Lumped Conceptual Models, Hydrology and Earth System Sciences, 25, 5517–5534, https://doi.org/10.5194/hess-25-5517-2021, 2021.

Lees, T., Reece, S., Kratzert, F., Klotz, D., Gauch, M., De Bruijn, J., Kumar Sahu, R., Greve, P., Slater, L., and Dadson, S. J.: Hydrological Concept Formation inside Long Short-Term Memory (LSTM) Networks, Hydrology and Earth System Sciences, 26, 3079–3101, https://doi.org/10.5194/hess-26-3079-2022, 2022.

Liu, J., Koch, J., Stisen, S., Troldborg, L., Højberg, A. L., Thodsen, H., Hansen, M. F. T., and Schneider, R. J. M.: CAMELS-DK: Hydrometeorological Time Series and Landscape Attributes for 3330 Catchments in Denmark, Earth System Science Data Discussions, pp. 1–30, https://doi.org/10.5194/essd-2024-292, 2024.

Loritz, R., Dolich, A., Acuña Espinoza, E., Ebeling, P., Guse, B., Götte, J., Hassler, S. K., Hauffe, C., Heidbüchel, I., Kiesel, J., Mälicke, M., Müller-Thomy, H., Stölzle, M., and Tarasova, L.: CAMELS-DE: Hydro-Meteorological Time Series and Attributes for 1555 Catchments in Germany, Earth System Science Data Discussions, pp. 1–30, https://doi.org/10.5194/essd-2024-318, 2024.

Mai, J., Shen, H., Tolson, B. A., Gaborit, É., Arsenault, R., Craig, J. R., Fortin, V., Fry, L. M., Gauch, M., Klotz, D., Kratzert, F., O'Brien, N., Princz, D. G., Rasiya Koya, S., Roy, T., Seglenieks, F., Shrestha, N. K., Temgoua, A. G. T., Vionnet, V., and Waddell, J. W.: The Great Lakes Runoff Intercomparison Project Phase 4: The Great Lakes (GRIP-GL), Hydrology and Earth System Sciences, 26, 3537–3572, https://doi.org/10.5194/hess-26-3537-2022, 2022.

Moshe, Z., Metzger, A., Elidan, G., Kratzert, F., Nevo, S., and El-Yaniv, R.: HydroNets: Leveraging River Structure for Hydrologic Modeling, https://doi.org/10.48550/arXiv.2007.00595, 2020.

https://doi.org/10.5194/egusphere-2024-3649
Preprint. Discussion started: 21 February 2025

Muhebwa, A., Gleason, C. J., Feng, D., and Taneja, J.: Improving Discharge Predictions in Ungauged Basins: Harnessing the Power of Disaggregated Data Modeling and Machine Learning, Water Resources Research, 60, e2024WR037 122, https://doi.org/10.1029/2024WR037122, 2024.

Muñoz Sabater, J.: ERA5-Land Hourly Data from 1950 to Present. Copernicus Climate Change Service (C3S) Climate Data Store (CDS). DOI: 10.24381/Cds.E2161bac (Accessed on 17-Sep-2024), 2019.

Nash, J. E. and Sutcliffe, J. V.: River Flow Forecasting through Conceptual Models Part I — A Discussion of Principles, Journal of Hydrology, 10, 282–290, https://doi.org/10.1016/0022-1694(70)90255-6, 1970.

Nearing, G., Cohen, D., Dube, V., Gauch, M., Gilon, O., Harrigan, S., Hassidim, A., Klotz, D., Kratzert, F., Metzger, A., Nevo, S., Pappenberger, F., Prudhomme, C., Shalev, G., Shenzis, S., Tekalign, T. Y., Weitzner, D., and Matias, Y.: Global Prediction of Extreme Floods in Ungauged Watersheds, Nature, 627, 559–563, https://doi.org/10.1038/s41586-024-07145-1, 2024.

Nevo, S., Morin, E., Gerzi Rosenthal, A., Metzger, A., Barshai, C., Weitzner, D., Voloshin, D., Kratzert, F., Elidan, G., Dror, G., Begelman, G., Nearing, G., Shalev, G., Noga, H., Shavitt, I., Yuklea, L., Royz, M., Giladi, N., Peled Levi, N., Reich, O., Gilon, O., Maor, R., Timnat, S., Shechter, T., Anisimov, V., Gigi, Y., Levin, Y., Moshe, Z., Ben-Haim, Z., Hassidim, A., and Matias, Y.: Flood Forecasting with Machine Learning Models in an Operational Framework, Hydrology and Earth System Sciences, 26, 4013–4032, https://doi.org/10.5194/hess-26-4013-2022, 2022.

Newman, A. J., Clark, M. P., Sampson, K., Wood, A., Hay, L. E., Bock, A., Viger, R. J., Blodgett, D., Brekke, L., Arnold, J. R., Hopson, T., and Duan, Q.: Development of a Large-Sample Watershed-Scale Hydrometeorological Data Set for the Contiguous USA: Data Set Characteristics and Assessment of Regional Variability in Hydrologic Model Performance, Hydrology and Earth System Sciences, 19, 209–223, https://doi.org/10.5194/hess-19-209-2015, 2015.

Otero, N., Horton, P., Martius, O., Allen, S., Zappa, M., Wechsler, T., and Schaefli, B.: Impacts of Hot-Dry Conditions on Hydropower Production in Switzerland, Environmental Research Letters, 18, 064 038, https://doi.org/10.1088/1748-9326/acd8d7, 2023.

Shalev, G., El-Yaniv, R., Klotz, D., Kratzert, F., Metzger, A., and Nevo, S.: Accurate Hydrologic Modeling Using Less Information, 2019.

Shen, C., Laloy, E., Albert, A., Chang, F.-J., Elshorbagy, A., Ganguly, S., Hsu, K.-l., Kifer, D., Fang, Z., Fang, K., Li, D., Li, X., and Tsai, W.-P.: Deep Learning as a Promising Avenue toward Knowledge Discovery in Water Sciences, https://doi.org/10.5194/hess-2018-168, 2018.

Sit, M., Demiray, B., and Demir, I.: Short-Term Hourly Streamflow Prediction with Graph Convolutional GRU Networks, https://doi.org/10.48550/arXiv.2107.07039, 2021.

Sitterson, J., Knightes, C., Parmar, R., Wolfe, K., Avant, B., and Muche, M.: An Overview of Rainfall-Runoff Model Types, International Congress on Environmental Modelling and Software, 2018.

Smith, A., Tetzlaff, D., Marx, C., and Soulsby, C.: Enhancing Urban Runoff Modelling Using Water Stable Isotopes and Ages in Complex Catchments, Hydrological Processes, 37, e14 814, https://doi.org/10.1002/hyp.14814, 2023.

Smith, J. and Eli, R. N.: Neural-Network Models of Rainfall-Runoff Process, Journal of Water Resources Planning and Management, 121, 499–508, https://doi.org/10.1061/(ASCE)0733-9496(1995)121:6(499), 1995.

Sterle, G., Perdrial, J., Kincaid, D. W., Underwood, K. L., Rizzo, D. M., Haq, I. U., Li, L., Lee, B. S., Adler, T., Wen, H., Middleton, H., and Harpold, A. A.: CAMELS-Chem: Augmenting CAMELS (Catchment Attributes and Meteorology for Large-sample Studies) with Atmospheric and Stream Water Chemistry Data, Hydrology and Earth System Sciences, 28, 611–630, https://doi.org/10.5194/hess-28-611-2024, 2024.

https://doi.org/10.5194/egusphere-2024-3649
Preprint. Discussion started: 21 February 2025

Sun, A. Y., Jiang, P., Mudunuru, M. K., and Chen, X.: Explore Spatio-Temporal Learning of Large Sample Hydrology Using Graph Neural Networks, Water Resources Research, 57, e2021WR030 394, https://doi.org/10.1029/2021WR030394, 2021.

Sun, A. Y., Jiang, P., Yang, Z.-L., Xie, Y., and Chen, X.: A Graph Neural Network (GNN) Approach to Basin-Scale River Network Learning: The Role of Physics-Based Connectivity and Data Fusion, Hydrology and Earth System Sciences, 26, 5163–5184, https://doi.org/10.5194/hess-26-5163-2022, 2022.

Tursun, A., Xie, X., Wang, Y., Peng, D., Liu, Y., Zheng, B., Wu, X., and Nie, C.: Streamflow Prediction in Human-Regulated Catchments Using Multiscale Deep Learning Modeling With Anthropogenic Similarities, Water Resources Research, 60, e2023WR036 853, https://doi.org/10.1029/2023WR036853, 2024.

Van Der Knijff, J. M., Younis, J., and De Roo, A. P. J.: LISFLOOD: A GIS-based Distributed Model for River Basin Scale Water Balance and Flood Simulation, International Journal of Geographical Information Science, 24, 189–212, https://doi.org/10.1080/13658810802549154, 2010.

Vischer, M. A., Felipe Otero, N., and Ma, J.: Spatially Resolved Meteorological and Ancillary Data in Central Europe for Rainfall Streamflow Modeling, under review, 2025.

Xiang, Z. and Demir, I.: Fully Distributed Rainfall-Runoff Modeling Using Spatial-Temporal Graph Neural Network, https://doi.org/10.31223/X57P74, 2022.

Xie, J., Liu, X., Tian, W., Wang, K., Bai, P., and Liu, C.: Estimating Gridded Monthly Baseflow From 1981 to 2020 for the Contiguous US Using Long Short-Term Memory (LSTM) Networks, Water Resources Research, 58, e2021WR031 663, https://doi.org/10.1029/2021WR031663, 2022.

Yu, Q., Tolson, B. A., Shen, H., Han, M., Mai, J., and Lin, J.: Enhancing Long Short-Term Memory (LSTM)-Based Streamflow Prediction with a Spatially Distributed Approach, Hydrology and Earth System Sciences, 28, 2107–2122, https://doi.org/10.5194/hess-28-2107-2024, 2024.

Zhou, F., Chen, Y., and Liu, J.: Application of a New Hybrid Deep Learning Model That Considers Temporal and Feature Dependencies in Rainfall–Runoff Simulation, Remote Sensing, 15, 1395, https://doi.org/10.3390/rs15051395, 2023.

Zhu, S., Wei, J., Zhang, H., Xu, Y., and Qin, H.: Spatiotemporal Deep Learning Rainfall-Runoff Forecasting Combined with Remote Sensing Precipitation Products in Large Scale Basins, Journal of Hydrology, 616, 128 727, https://doi.org/10.1016/j.jhydrol.2022.128727, 2023.