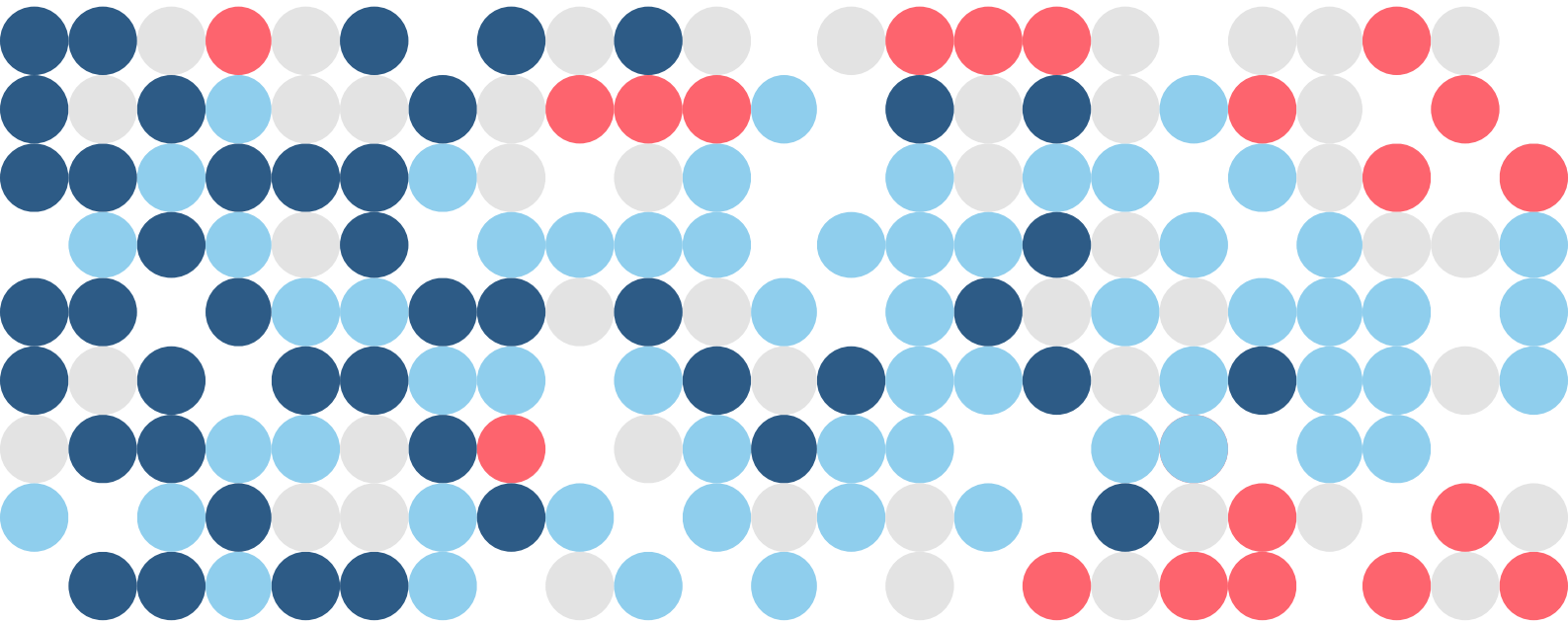
Operational Streamflow Drought Forecasting for the Rhine River at Lobith Using the LSTM Deep Learning Approach

Master Thesis Report

Jing Deng



TU Delft | Deltares

Operational Streamflow Drought Forecasting for the Rhine River at Lobith Using the LSTM Deep Learning Approach

By

Jing Deng

in partial fulfilment of the requirements for the degree of

Master of Science

in Civil Engineering

at the Delft University of Technology,

to be defended publicly on July 6, 2023 at time 1:00 PM.

Committee TU Delft:	Riccardo Taormina
	Markus Hrachowitz
Supervisors Deltares:	Anaïs Couasnon
	Ruben Dahm

An electronic version of this thesis is available at http://repository.tudelft.nl/.

TU Delft Deltares

ACKNOWLEDGEMENTS

I would like to express my gratitude for the incredible journey I have embarked on over the past two years since I made the decision to come to the Netherlands and pursue further studies. This endeavor has allowed me to delve deeply into the captivating fields of water management and artificial intelligence, opening up a world of knowledge and exploration.

My heartfelt gratitude goes to Riccardo, whose passion and extensive knowledge of machine learning in water management enriched my understanding of the subject. Our discussions on technical details were enjoyable and enlightening. Also, I would like to thank Markus for sharing his invaluable insights into hydrology and his critical thinking regarding data-driven models.

This thesis would not have been possible without Ruben's trust in my ability and the opportunity he provided for me to work on the thesis at Deltares. His vast knowledge and fascinating ideas served as a constant source of inspiration. I extend a huge thanks to Anaïs for her meticulous guidance as my daily supervisor. Her expertise in research and mentorship smoothed my path and provided invaluable support throughout the entire process.

Many thanks to Hans and Klaas-Jan for their insightful feedback on drought and forecasting. Thanks to Jonathan for sharing his expert knowledge in machine learning. I would also like to thank Albrecht for providing valuable data and results, Matthijs for introducing me to the FEWS system, and Nathalie for her valuable input during the scoping phase.

To my friends in the Netherlands, I feel incredibly fortunate to have had your support and companionship throughout this journey. You made it all the more meaningful.

Lastly, I want to give special thanks to my family in China for their love and support. And to my husband Zijian, my best cheerleader and soulmate, words cannot express my gratitude for your constant encouragement and belief in me.

Jing Deng
Delft, June 2023

ABSTRACT

Under future warmer climates, drought events are projected to occur more frequently with increasing impacts in many regions and river basins. This study focuses on exploring the potential of the LSTM deep learning (DL) approach for operational streamflow drought forecasting for the Rhine River at Lobith with a lead time (LT) of up to 46 days.

The research investigates optimal spatial resolution, input and target variables, and loss functions. Four LSTM-based model architectures are developed and tested, incorporating both historical observation and forecast data to generate 46-step forecasts simultaneously. The robustness and stability of the models are assessed through cross-validation, and their performances are compared. Subsequently, the performance of the LSTM-based model is compared to the physically-based models, namely Wflow-Rhine and FEWS-Rhine, in forecasting streamflow drought.

The results suggest that utilizing a subbasin spatial resolution, including historical discharge as input, and training the model on time-differenced data enhance the forecast skill. Among the evaluated models, the model architecture with two LSTMs in cascade exhibits stable and robust performance across the forecast horizon and is considered for operational use in this study. Comparisons between the DL model and physically-based models indicate that: 1) When using observed meteorology forcing from ERA5, the DL model demonstrates a notable performance compared to Wflow-Rhine simulation using the same forcing data. 2) When utilizing SEAS5 for forecasting, the DL model demonstrates skill over Wflow-Rhine in predicting discharge levels during the dry season up to 10 days ahead, as well as for discharges between 950 and 2200 m^3/s across the entire forecast horizon. However, for discharges between 700 and 950 m^3/s with longer LTs beyond 20 days, Wflow-Rhine shows skill over the DL model. 3) While FEWS-Rhine successfully forecasts drought events in 2018 throughout the forecast horizon, it tends to produce more Type I errors (false positives). The DL model, forecasting with SEAS5, accurately predicts drought events in 2018 for LTs up to 30 days and generally has higher precision values. Despite using different forcing datasets, the DL model can predict the timing and trend of past drought events, indicating its potential in capturing streamflow patterns.

This study contributes to operational water management in the Netherlands by employing the LSTM deep learning approach in an operational framework for drought forecasting. By leveraging historical observation data and forecasted meteorology forcing data, these models achieve skillful performances for streamflow drought forecasts. Future research could focus on further enhancing model performance, exploring the applicability of the LSTM-based models in other river basins, and validating the results in real operational settings.

Contents

ACKNOWLEDGEMENTS i
ABSTRACT ii
List of figures vi
List of tables x
List of abbreviations xi
1 Introduction 1
1.1 Research motivation 1
1.2 Problem statement 4
1.3 Research objective 5
1.4 Research questions 6
1.5 Reading guide 6
2 Backgrounds 7
2.1 Model framework for operational forecast 7
2.2 FEWS-Rhine 8
2.3 Wflow-Rhine 9
2.4 Deep learning models for operational hydrological forecasting 10
2.4.1 Leveraging both historical observation and forecast data 10
2.4.2 Multi-step time series forecasting using DL models 12
3 Materials and methods 13
3.1 Study area 13
3.2 Datasets 14
3.2.1 Streamflow at Lobith 14
3.2.2 Meteorological parameters 15
3.2.3 System storage parameters 16
3.3 Model architectures 16
3.3.1 Model 1 17
3.3.2 Model 2 17
3.3.3 Model 3 18
3.3.4 Model 4 18
3.4 Data processing 19
3.4.1 Preprocessing 19
3.4.2 Postprocessing 21
4 Experimental designs 22

4.1 Quantifying the impact of spatial resolution, input and target variables, and loss functions (SQ1) 22
4.1.1 Spatial resolution 22
4.1.2 Input and target variables 22
4.1.3 Loss functions 23
4.1.4 Experiment setup 23
4.2 Comparing different model architectures (SQ2) 25
4.2.1 Cross-validation 25
4.2.2 Comparative analysis 26
4.2.3 Experiment setup 26
4.3 Comparing the DL model with physically-based models (SQ3) 27
4.3.1 Experiment 3A: DL model vs Wflow-Rhine with ERA5 27
4.3.2 Experiment 3B: DL model vs Wflow-Rhine with SEAS5 28
4.3.3 Experiment 3C: DL model vs FEWS-Rhine 29
5 Results and discussions 30
5.1 Quantifying the impact of spatial resolution, input and target variables, and loss functions (SQ1) 30
5.1.1 Experiment 1A: spatial resolution 30
5.1.2 Experiment 1B: input and target variables 32
5.1.3 Experiment 1C: loss weight 33
5.1.4 Discussion of results for SQ1 35
5.2 Comparing different model architectures (SQ2) 37
5.2.1 Experiment 2A: cross-validation 37
5.2.2 Experiment 2B: comparative analysis 41
5.2.3 Discussion of the results for SQ2 43
5.3 Comparing the DL model with physically-based models (SQ3) 44
5.3.1 Experiment 3A: DL model vs Wflow-Rhine with ERA5 44
5.3.2 Experiment 3B: DL model vs Wflow-Rhine with SEAS5 46
5.3.3 Experiment 3C: DL model vs FEWS-Rhine 50
5.3.4 Discussion of the results for SQ3 52
6 Limitations and recommendations 54
7 Conclusion 56
Bibliography 58
A Time terms 62
B DL model hyperparameters and settings 63
C Cross-validation method 65

D Quantifying the impact of spatial resolution, input and target variables, and loss functions 66
D.1 Experiment 1A: spatial resolution 66
D.2 Experiment 1B: input and target variables 68
E Comparing different model architectures 69
E.1 Experiment 2A: cross-validation 69
E.2 Experiment 2B: comparative analysis 77
F Comparing the DL model with physically-based models 79

List of figures

Figure 1-1 Threshold level method for streamflow drought identification, including an illustration of drought duration, deficit volume, and pooled events. The solid line is the observed or forecasted streamflow. The dashed line is the streamflow drought detection threshold. When the solid line drops below the dashed line, a drought takes place. (Modified from Van Loon (2015)). 1

Figure 1-2 Hydrological map of the Netherlands (Rijkswaterstaat, 2019). 2

Figure 1-3 An example of 5-day forecast and 14-day forecast with uncertainty bands reported on Rijkswaterstaat's water reporting website. This forecast starts on May 24th, 2023. The thick red line represents the 5-day forecast. The red "plume" represents the 14-day forecast with uncertainty bands where the light red is 10-90% band, and the dark red is 33-66% band. The 46-day forecast is not reported and shown on the website. 3

Figure 2-1 Illustration of historical and forecast modes of operation for real-time operational streamflow forecast model. (Modified from Weerts (2009)) 7

Figure 2-2 HBV sub-basins for the Rhine. 8

Figure 2-3 Overview of the different processes and fluxes in the wflow_sbm model (van Verseveld et al., 2023). 9

Figure 2-4 Visualization of the standard LSTM cell, where c[t] denotes the cell state at time step t, h[t] the hidden state, x[t] the input. f stands for the forget gate, i for the input gate, g for the cell update, and o for the output gate (Kratzert et al., 2019). 10

Figure 2-5 Illustration of two LSTMs in parallel. "Dense" means dense layer in DL models. . 11

Figure 2-6 Illustration of two LSTMs in cascade. "Dense" means dense layer, and "FC" represents fully connected layer in DL models. 11

Figure 3-1 Nine subbasins of the Rhine River basin (Deltares, 2019) and the location of Lobith. 13

Figure 3-2 River discharge climatology at Lobith based on different 10-year periods. (Data source: Rijkswaterstaat) 14

Figure 3-3 Illustration of Model 1 architecture. Note that X1 might include historical discharge Y from past L days. For simplicity, notation X1 is used to stand for all inputs of LSTM-1..... 17

Figure 3-4 Illustration of Model 2 architecture. Note that X1 might include historical discharge Y from past L days. For simplicity, notation X1 is used to stand for all inputs of LSTM-1..... 18

Figure 3-5 Illustration of Model 3 architecture. Note that X1 might include historical discharge Y from past L days. For simplicity, notation X1 is used to stand for all inputs of LSTM-1..... 18

Figure 3-6 Illustration of Model 4 architecture. Note that X1 might include historical discharge Y from past L days. For simplicity, notation X1 is used to stand for all inputs of LSTM-1..... 19

Figure 3-7 Example plot of the original streamflow, log-transformed streamflow, and time-differenced log-transformed streamflow at Lobith. 20

Figure 3-8 Schematic of the sequence preparation. For each initialization time, the data sequences of past 270 days (light blue) are for LSTM-1. The data sequences of future 46 days (blue) are for LSTM-2 (or the second group of LSTMs). 21

Figure 4-1 Four types of loss weights designed for the experiments. 23

Figure 4-2 Schematic of the experiment workflow for SQ1. The multiplication factor here refers to the exploration of all possible combinations between the two sides. 24

Figure 5-1 MAE and MAPE of Experiment 1A results from basin mean approach with various input variables (Model 3). Basin mean approach refers to the spatial resolution experiment where the spatially distributed variables are processed as mean values across the entire Rhine basin. 30

Figure 5-2 MAE and MAPE of Experiment 1A results from subbasin mean approach with various input variables (Model 3). Subbasin mean approach refers to the spatial resolution experiment where the spatially distributed variables are processed as mean values over the eight subbasins upstream of Lobith. 31

Figure 5-3 MAE and MAPE of Experiment 1B results from training the model on the time-differenced data (*delta_Q*) with various input variables (Model 3). 32

Figure 5-4 MAE and MAPE of Experiment 1B results from training the model on discharge (Q) directly with various input variables (Model 3). Note that Figure 5-4 is the same as Figure 5-2, included again in this section for comparison purposes. 32

Figure 5-5 MAE and MAPE of Experiment 1B results on different targets variables (*delta_Q* and *Q*) when including historical discharge at Lobith (*Q_his*) as an input variable. 33

Figure 5-6 MAE and MAPE of Experiment 1C results on different types of loss weights for a) Model 1, b) Model 2, c) Model 3, and d) Model 4. 34

Figure 5-7 Autocorrelation of discharge at Lobith with different lags based on different 10-year periods. (Data source: Rijkswaterstaat) 36

Figure 5-8 MAE and MAPE of Experiment 1B results on different targets variables (delta_Q and Q) when only general meteorology forcing parameters (*tp*, *t2m*, *pev*) are used as input variables. (Model 3) 37

Figure 5-9 MAE and MAPE of cross-validation results for Model 1. Iteration 1 to 5 represent the five models trained on different cross-validation time series data splits. Benchmark represent the model trained using all the available training data without any cross-validation splitting. 38

Figure 5-10 Absolute error (AE) of cross-validation results for Model 1. Iteration 1 to 5 represent the five models trained on different cross-validation time series data splits. Benchmark represent the model trained using all the available training data without any cross-validation splitting. 39

Figure 5-11 MAE and MAPE of cross-validation results for Model 3. Iteration 1 to 5 represent the five models trained on different cross-validation time series data splits. Benchmark represent the model trained using all the available training data without any cross-validation splitting. 40

Figure 5-12 Absolute error (AE) of cross-validation results for Model 3. Iteration 1 to 5 represent the five models trained on different cross-validation time series data splits.

Benchmark represent the model trained using all the available training data without any cross-validation splitting. 40

Figure 5-13 AE results of different model architectures. 41

Figure 5-14 AE results of different model architectures for several LTs. 42

Figure 5-15 CDF of AE results of different model architectures. Several LTs are highlighted in colors. The grey lines are the results for other LTs. 42

Figure 5-16 Forecast results of the DL model and simulation results of Wflow-Rhine for Experiment 3A. The blue line represents the true observation. The orange line represents the simulation results of Wflow-Rhine with ERA5. The red-yellow-green gradient line represents the forecast results of the DL model (Model 3) with ERA5, where the forecast is initialized daily. Note that the gradient colors represent different LTs, with red indicating shorter LTs and green indicating longer LTs..... 45

Figure 5-17 Comparison of MAE and MAPE with standard deviation (std) for forecast results of the DL model and simulation results of Wflow-Rhine in Experiment 3A. Blue lines represent the MAE/MAPE with std for DL model forecast. The orange line represents the MAE/MAPE and the orange fill represents the std of wflow simulation, assuming the same for all LTs... 46

Figure 5-18 Forecast results of DL model and Wflow-Rhine with SEAS5 initialized on 2018-08-01 and 2018-09-01. 46

Figure 5-19 Forecast results of DL model with ERA5 initialized on 2018-09-01. 47

Figure 5-20 CRPS of the DL model forecast results and the wflow forecast results. 48

Figure 5-21 CRPSS of the DL model forecasts compared to wflow model forecasts. The dashed line indicates the CRPSS value of zero..... 48

Figure 5-22 Median CRPS of the DL model forecast results and the wflow forecast results for different discharge levels. Colors are proportional to the CRPS value. Blank means there are less than 2 samples for that cell. 49

Figure 5-23 Median CRPSS of the DL model forecasts compared to wflow model forecasts for different discharge levels. Colors represent different value range. Blue represents positive value, i.e., the DL model forecast outperforms the wflow forecast, and red is the opposite. Blank means there are less than 2 samples for that cell. 49

Figure 5-24 Number of samples for each combination of discharge observation (Qobs) bin and lead time of Experiment 3B: DL model vs Wflow-Rhine with SEAS5. Blank means there are less than 2 samples for that cell. 49

Figure A-1 Illustration of relevant time terms and notations in the context of this study forecasting up to 46 days ahead.....62

Figure C-1 Illustration of cross validation method “sliding window with gaps” used in this study.....65

Figure D-1 MAE and MAPE of Experiment 1A results from basin mean approach and subbasin mean approach with various input variables for a) Model 1, b) Model 2, c) Model 4. Basin mean approach refers to the spatial resolution experiment where the spatially distributed variables are processed as mean values across the entire Rhine basin, while subbasin mean approach refers to the spatial resolution experiment where the spatially distributed variables are processed as mean values over the eight subbasins upstream of Lobith.....66

Figure D-2 MAE and MAPE of Experiment 1B results from training the model on different targets, i.e., the time-differenced data (delta_Q) or discharge (Q), with various input variables, for a) Model 1 and b) Model 2……..68

Figure E-1 Cross-validation results for a) Model 1, b) Model2, c) Model 3, and d) Model 4. Iteration 1 to 5 represent the five models trained on different cross-validation time series data splits. Benchmark represent the model trained using all the available training data without any cross-validation splitting……69

Figure E-2 APE results of different model architectures ……77

Figure E-3 APE results of different model architectures for several LTs……77

Figure E-4 CDF of APE results of different model architectures. Several LTs are highlighted in colors. The grey lines are the results for other LTs……78

Figure F-1 CRPS of the DL model forecast results, and CRPSS of the results compared to wflow model forecasts. EOBS is used for both X1 (input for LSTM-1) and X2 (input for LSTM-2) in training mode, and used for X1 together with SEAS5 for X2 in the forecast mode……79

Figure F-2 DL model forecast results for the drought event in 2018 with SEAS5……80

Figure F-3 FEWS-Rhine forecast results for the drought event in 2018 with ENS extended……81

List of tables

Table 1-1 River discharge criteria at Lobith for scaling up from level 0 (normal management, in Dutch “normaal beheer”) to level 1 (impending water shortages, in Dutch “dreigende watertekorten”). This includes the expectation that the situation will last longer than three days. The discharge criteria is important for being able to meet the water demand of, among others, agriculture, nature, industry, drinking water, process and cooling water and for a number of surface water functions such as shipping. Modified from de Vries et al. (2021).... 3

Table 2-1 Overview of different forecast lines. 9

Table 3-1 Summary of the parameters and datasets used in this study. 16

Table 4-1 Overview of the experiment setup for SQ1 25

Table 4-2 Train-validation-testing period for cross validation. 26

Table 4-3 Overview of the experimental setup for cross-validation and comparative analysis. 27

Table 4-4 Experimental setup details of Experiment 3A. 28

Table 4-5 Experimental setup details of Experiment 3B. 28

Table 4-6 Experimental setup details of Experiment 3C. 29

Table 5-1 The combinations of spatial resolution, input and target variables, and loss weights for different model architectures employed in the experiments in this section. 37

Table 5-2 Training time and number of trainable parameters for each model architecture. The training time is from Experiment 2B where the models are trained on all available training data. All the trainings are done on Google Colab with GPU A100. 44

Table 5-3 Confusion matrix of DL model forecast results for different initialization times and lead time (LT) bins. 51

Table 5-4 Recall and precision values derived from confusion matrix of DL model forecast results for different initialization times and lead time (LT) bins. 51

Table 5-5 Confusion matrix of FEWS-Rhine forecast results for different initialization times and lead time (LT) bins. Note that the forecast results are missing after 2018-12-14, so for LT 41-46 of initialization 2018-11-01, results are only available for 3 days. 51

Table 5-6 Recall and precision values derived from confusion matrix of FEWS-Rhine forecast results for different initialization times and lead time (LT) bins. 52

Table B-1 Model 1 hyperparameters and settings 63

Table B-2 Model 2 hyperparameters and settings 63

Table B-3 Model 3 hyperparameters and settings 63

Table B-4 Model 4 hyperparameters and settings 64

List of abbreviations

AE	Absolute Error
ARMA	Autoregressive Moving Average
APE	Absolute Percentage Error
CRPS	Continuous Ranked Probability Score
CRPSS	Continuous Ranked Probability Skill Score
DL	Deep Learning
ECMWF	European Centre for Medium-Range Weather Forecasts
FC	Fully Connected Layer
LSTM	Long Short-Term Memory
LT	Lead Time
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
MIMO	Multi-Input Multi-Output Strategy
SHMI	Swedish Hydrological and Meteorological Institute
SQ	Sub-research Question
WMCN	Netherlands Water Management Center

1 Introduction

1.1 Research motivation

Since the beginning of the 21st century, Europe has experienced a series of severe droughts (2003, 2015, 2018, and 2022), affecting a wide range of socio-economic sectors including agriculture, energy production, waterborne transportation, public water supply and freshwater ecosystem (EEA, 2010; Ionita et al., 2017; WMO, 2020). Under future warmer climate, drought events are projected to occur more frequently with increasing impacts in many regions and river basins (Cammalleri et al., 2020; Prudhomme et al., 2014; van der Wiel et al., 2019; Wanders & Van Lanen, 2015). The areas affected by droughts are typically larger than those for other hazards. But the slow onset of droughts allows more time for monitoring and forecasting.

Droughts are generally classified into four categories (Tallaksen & Van Lanen, 2004; Van Loon, 2015; Wilhite & Glantz, 1985): meteorological drought, soil moisture drought, hydrological drought, and socioeconomic drought. Van Loon et al. (2016) propose to broaden the definition of drought to include water shortage caused and modified by human processes. Hydrological drought is related to negative anomalies in surface and subsurface water. It is a result of climate variability, catchment characteristics and anthropogenic influences (Van Lanen et al., 2013; Van Loon & Van Lanen, 2012). Climate variability includes precipitation deficits, heat wave induced high evaporation, freezing conditions in winter in snow-dominated catchments, or low temperatures in summer in glacier-dominated catchments (Van Loon, 2015). Catchment characteristics, such as land cover, soil types, geology, and groundwater storage capacity, show a significant relation with hydrological drought (Van Loon, 2015). Anthropogenic influences drive hydrological drought through water extraction, reservoir construction, as well as deforestation and urbanization (Van Loon et al., 2016). This research focuses on streamflow drought, which is a part of hydrological drought and is defined as below-normal river discharge (Van Loon, 2015). Drought duration, severity and frequency can be identified using the threshold level method (Figure 1-1).

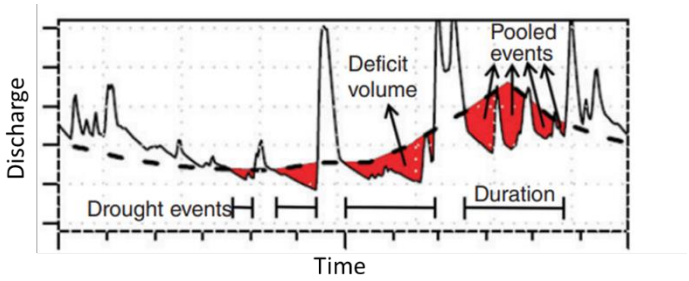


Figure 1-1 Threshold level method for streamflow drought identification, including an illustration of drought duration, deficit volume, and pooled events. The solid line is the observed or forecasted streamflow. The dashed line is the streamflow drought detection threshold. When the solid line drops below the dashed line, a drought takes place. (Modified from Van Loon (2015)).

The Netherlands has experienced drought during the summer in recent years. The 2018, especially, was an extremely dry year with an averaged national water deficit of 309 mm (Kramer et al., 2019). Given its unique geographical composition, consisting of low-lying delta regions with peat and clay soils, as well as upland regions with sandy soils, the Netherlands

heavily relies on two large transboundary rivers, i.e., the Rhine and the Meuse, for fresh water supply.

The Rhine enters the Netherlands at Lobith. The river first splits as the Waal River and Pannerdensch Kanaal at the Pannerdense Kop. The latter flows into the Lower Rhine. At the IJsselkop, the IJssel River leaves the Lower Rhine, supplying water to Lake IJssel. The Lower Rhine supplies water to feed the Amsterdam-Rhine Canal for inland shipping, drinking-water supply, and cooling power stations in Utrecht and Amsterdam. The rest flows into the Waal River towards the sea, acting as a force against water intrusion from the sea. The distribution of water through the branches of the Rhine is shown in Figure 1-2.

Figure 1-2 Hydrological map of the Netherlands (Rijkswaterstaat, 2019).

It is important to know how much water is currently flowing into Lobith and how much to expect in the coming period, as it determines the navigable depth for shipping and the availability of water for agriculture, nature and drinking water in a large part of the country. During dry periods, when the discharge at Lobith is lower than a certain threshold (Table 1-1), operational measures need to be taken to distribute river water according to the “priority sequencing hierarchy” (Rijkswaterstaat, 2019). Therefore, a reliable and robust forecasting of streamflow drought at Lobith is essential for Dutch water managers and stakeholders to develop robust strategies for drought mitigation and adaptation.

Table 1-1 River discharge criteria at Lobith for scaling up from level 0 (normal management, in Dutch "normaal beheer") to level 1 (impending water shortages, in Dutch "dreigende watertekorten"). This includes the expectation that the situation will last longer than three days. The discharge criteria is important for being able to meet the water demand of, among others, agriculture, nature, industry, drinking water, process and cooling water and for a number of surface water functions such as shipping. Modified from de Vries et al. (2021)

Maand	**Rijnafvoer bij Lobith** (etmaalgemiddeld in m^3/s)
Januari – april	1000
Mei	1400
Juni	1300
Juli	1200
Augustus	1100
September – december	1000

Efforts have been made on developing operational forecasting systems that can predict hydrological variables such as water levels and discharge. One such system is the FEWS-Rhine, a state-of-the-art operational flood early warning system developed for the Dutch government. This system utilizes physically-based hydrological models to forecast the discharge of the Rhine River at Lobith, offering a lead time up to 46 days based on available weather forecast products. The system also helps to provide a 5-day forecast established by Netherlands Water Management Center (WMCN) and a 14-day forecast with uncertainty bands that are reported on Rijkswaterstaat's water reporting website[1] (See Figure 1-3 for example).

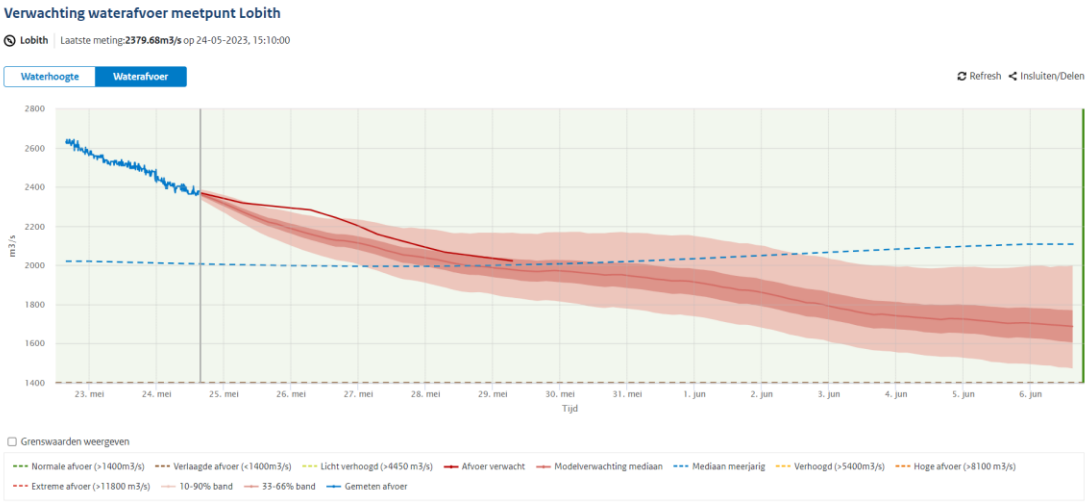


Figure 1-3 An example of 5-day forecast and 14-day forecast with uncertainty bands reported on Rijkswaterstaat's water reporting website. This forecast starts on May 24th, 2023. The thick red line represents the 5-day forecast. The red "plume" represents the 14-day forecast with uncertainty bands where the light red is 10-90% band, and the dark red is 33-66% band. The 46-day forecast is not reported and shown on the website.

However, the FEWS-Rhine system has primarily been calibrated for short-term and flood forecasting, with less emphasis on low flows dynamics or droughts. As such, there is a need

[1] https://waterberichtgeving.rws.nl/owb/droogtemonitor /rijnenmaas

to understand the performance of the system for low flows and develop an operational streamflow drought forecasting system that can accurately predict low flow events at Lobith. Such a system would be valuable in providing timely and accurate information to water managers, policymakers, and other stakeholders, enabling them to make informed decisions regarding water allocation and management during periods of drought.

1.2 Problem statement

Commonly applied methods for forecasting hydrological drought involve the use of physically-based hydrological models combined with meteorological forecasts (Fundel et al., 2013; Sutanto et al., 2020; van Hateren et al., 2019). However, physically-based hydrological models do have limitations. Since they attempt to comply with the laws of conservation of mass, energy, and momentum, they heavily rely on the knowledge of boundary conditions, system stages, and system parameters (Hrachowitz & Clark, 2017). Obtaining accurate knowledge of these factors can be challenging, especially for large basins such as the Rhine basin. Additionally, anthropogenic influences on regulated rivers, such as reservoir operation and water extraction by power stations, can also be difficult to incorporate into the model due to a lack of information, which can negatively affect the skill of hydrological drought predictions and make calibration particularly challenging.

Over the past few years, data driven approaches, such as deep learning (DL) models, have been explored and tested out increasingly in hydrology (Shen, 2018; Shen et al., 2021). In particular, studies have shown that Long Short-Term Memory (LSTM) models have been proven to be effective tools for the dynamic modeling of streamflow (Kratzert et al., 2019) and soil moisture (Fang et al., 2017), which has led to an increase in the use of DL techniques across all domains of hydrology.

Recent literatures on drought prediction using data driven methods (Aghelpour et al., 2021; Amanambu et al., 2022; Borji et al., 2016; Dikshit et al., 2021; Shamshirband et al., 2020) and the literature review on machine learning applications for hydrological streamflow forecasting (Ibrahim et al., 2022) show a trend toward using DL models. Drought is a complex phenomenon involving several variables that are often correlated at various lag times, and deep neural networks can effectively capture the decay-weighted lag–lead sequence relationship, leading to better forecasting results at both short and long lead times (Dikshit et al., 2022). LSTM architecture is particularly effective in this regard, and recent research (Hunt et al., 2022) suggests that LSTMs outperform both random forest models and XGBoost on streamflow prediction problems, especially with increasing sample size.

Most of the studies on drought forecasting using DL techniques focus on predicting drought indices such as meteorological drought indices Standardized Precipitation Index (SPI) and Standardized Precipitation and Evapotranspiration Index (SPEI) (Dikshit et al., 2022), as well as hydrological drought indices Streamflow Drought Index (SDI) (Aghelpour et al., 2021; Borji et al., 2016; Shamshirband et al., 2020). These indices are typically calculated on a monthly scale. There are relatively few studies using DL techniques to forecast streamflow drought or low flow time series on a daily scale. Sahoo et al. (2019) develop LSTM and RNN models to predict one-step-ahead monthly low flow time series using the past two months' low flow values. Amanambu et al. (2022) use a transformer and a LSTM model with past daily stage

level as input to predict stage levels for multi-step ahead (i.e., 30, 60, 90, 120, and 180 days), which are then post-processed into hydrological drought series. However, these studies use the same variable for both input and output, without incorporating meteorology forcing. Moreover, none of these studies apply an operational framework where forecasted meteorological forcing data can provide additional and vital information for long-term, multi-step ahead time series forecasting.

Recent studies have shown that DL models can be used effectively for streamflow forecasting in operational frameworks. Google's operational flood forecasting system, for instance, models stage forecasting with two LSTMs that take into account historical and forecast precipitation, as well as stages of the target gauge and upstream gauges (Nevo et al., 2022). Hunt et al. (2022) used the LSTM model to predict streamflow at various river gauge stations across the western United States, demonstrating the effectiveness of the model in simulating streamflow during a testing phase when the models were fed with ERA5 data, and in forecasting streamflow at lead times of up to 10 days during an operational phase when the models were fed forecast variables from the European Centre for Medium-Range Weather Forecasts (ECMWF).

There is a lack of research on using DL models for streamflow drought forecasting in operational frameworks, especially at lead time longer than 10 days. Hence, this research aims to fill this gap by investigating the potential of the LSTM deep learning approach for operational streamflow drought forecasting for the Rhine River at Lobith, with lead times up to 46 days ahead which is in line with the current forecasting system FEWS-Rhine. The study aims to assess the performance of a DL drought forecasting approach that could potentially provide improved forecast skills for operational water management of droughts in the Netherlands. By applying DL models in an operational framework, both near-real-time observation data and forecasted meteorology forcing data can be leveraged to improve the accuracy and lead time of streamflow drought forecasts.

1.3 Research objective

The overall goal of this research is to investigate the potential of the LSTM deep learning approach for operational streamflow drought forecasting for the Rhine River at Lobith, with a lead time of up to 46 days, on a daily scale.

The first objective of this research is to explore different spatial resolution, input and target variables, and loss functions to identify the optimal combination for forecasting streamflow drought using LSTM-based models. This will involve additional information on snow, lake level and historical discharge at Lobith based on hydrological drought and operation knowledge and at which spatial resolution. Additionally, different target variables and loss functions will be tested to evaluate their impact on the model performance.

The second objective is to develop and explore different model architectures that can handle the various data sources available in an operational framework. This will include exploring direct and recursive methods for forecasting multi-step ahead time series. The optimal combination of spatial resolution, input and target variables, and loss functions identified in the first objective will be used to compare the performance of the different model architectures.

The third objective is to compare the performance of the LSTM-based models with the physically-based model in forecasting streamflow drought. The research will examine how the forecast meteorology data could be integrated into the LSTM-based model and how it performs when compared to the distributed model Wflow-Rhine and current operation system FEWS-Rhine. This will provide insights into the relative strengths and weaknesses of the two approaches and inform the potential of LSTM-based models to supplement physically-based models in an operational framework.

1.4 Research questions

Based on the aforementioned objective, the main research question for this study is:

To what extent can the LSTM deep learning approach be used for operational streamflow drought forecasting for the River Rhine at Lobith?

To address this question, three sub-research questions (SQ) have been formulated to align with the specific research objectives.

SQ1: What combinations of spatial resolution, input and target variables, and loss functions can be used to optimize the performance of LSTM-based models for drought forecasting?

SQ2: What LSTM-based model architectures are suitable for handling the various data sources available in an operational framework and how do they compare in performance?

SQ3: How does the performance of the LSTM-based model compared to physically-based models for drought forecasting?

1.5 Reading guide

The report is structured as follows: Chapter 2 provides the theoretical background for the study, including the model framework for operational forecast, an overview of FEWS-Rhine and Wflow-Rhine, and an exploration of critical issues related to DL models for operational hydrological forecasting. Chapter 3 presents the study area, datasets and data processing steps, and gives an overview of the model architectures. Chapter 4 describes the experimental designs for each sub-research question. Chapter 5 presents and discusses the results of the experiments for each sub-research question. Chapter 6 states the limitations of the study and recommendations for future research in the field. Chapter 7 presents the conclusions and key findings of the study.

2 Backgrounds

In this chapter, background information is given on the model framework for operational forecast in section 2.1. An overview of the FEWS-Rhine operation system that is currently being used in operation is provided in section 2.2. And the Wflow-Rhine, which is envisioned as the future model for operational use, is described in section 2.3. Two critical considerations related to the implementation of DL techniques for operational hydrological forecasting in this study are discussed in section 2.4.

2.1 Model framework for operational forecast

Real-time operational streamflow forecasts, which currently heavily rely on physically based models, often utilize a cascade of hydrological and sometimes also hydrodynamic models. These models are interconnected and are typically embedded in a data-management environment such as FEWS-Rhine. Model cascades operates in two main modes: historical mode and forecast mode (Weerts, 2009).

The historical mode involves forcing the models with hydrological and meteorological observations over a limited time period preceding the forecast. This mode is used to initialize the model storages and establish the initial conditions. The forecast mode is employed to run the models over the required forecast lead time. In this mode, models are forced by outputs from other models, with the internal model states at the end of the historical run used as initial conditions for the forecast run. The outputs from other models may include meteorological forecasts such as precipitation, air temperature, and evaporation, as well as forecasts from upstream river locations. Figure 2-1 illustrates these different modes of operation and shows how they differ from the model calibration during which much longer period of records are used.

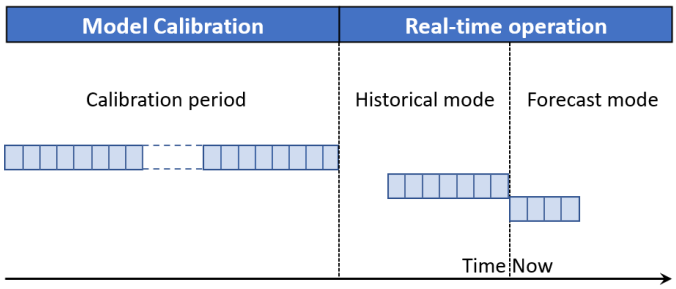


Figure 2-1 Illustration of historical and forecast modes of operation for real-time operational streamflow forecast model. (Modified from Weerts (2009))

In operational forecasting, various time terms are used to specify different aspects of the forecast, for instance, forecast initialization time, horizon and lead time. Details of the relevant terms and notations used in the context of this study can be found in Appendix A.

2.2 FEWS-Rhine

FEWS-Rhine is the operational system used by Rijkswaterstaat to generate forecasts for water levels and discharge in the Rhine. Initially, it was set up as a flood forecasting system, but its application has increasingly shifted in recent years to the entire discharge range. With FEWS-Rhine, measured and forecasted meteorological, hydrological and hydrodynamic data from multiple sources are automatically imported, validated, transformed and prepared for various forecast models.

To simulate the discharge from the sub-basins of the Rhine, FEWS-Rhine employs the HBV hydrological model, a lumped model from the Swedish Hydrological and Meteorological Institute (SHMI). In this model, each sub-basin is represented by interconnected "basins", each representing a hydrologically relevant zone. Figure 2-2 provides a visualization of the HBV sub-basins for the Rhine.

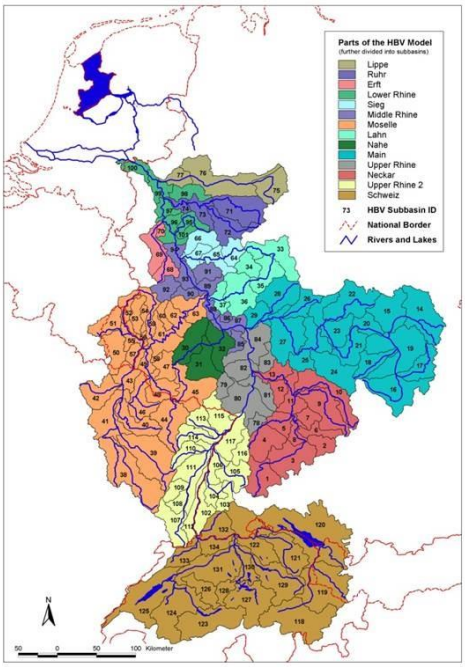


Figure 2-2 HBV sub-basins for the Rhine.

FEWS-Rhine utilizes predicted precipitation and temperature data from European Centre for Medium-Range Weather Forecasts (ECMWF) meteorological models to force its forecast models. Three specific ECMWF forecast products [2] are employed for this purpose: the Atmospheric Model high resolution 10-day forecast (HRES), the Ensemble 15-day forecast (ENS), and the Ensemble extended forecast (ENS extended). These forecast products serve as inputs to the FEWS-Rhine system, enabling it to generate various outputs. Some of the outputs are further corrected with an autoregressive moving average model (ARMA). An overview of different forecast lines, including forecast products, horizon, frequency, and output correction, is provided in Table 2-1.

[2] Catalogue of ECMWF real-time products: https://www.ecmwf.int/en/forecasts/datasets/catalogue-ecmwf-real-time-products

Table 2-1 Overview of different forecast lines.

Forecast line notation	Forecast product	Forecast horizon	Initialization frequency	Output correction
fews_hbv_hres_bias	HRES	10 days	Every day	ARMA
fews_hbv_ens_bias	ENS	15 days	Every day	ARMA
fews_hbv_ens_ext	ENS extended	46 days	Twice every week	No correction

2.3 Wflow-Rhine

The distributed hydrological model wflow is being considered for operational and policy purposes to replace the HBV model as this would provide discharge predictions along the whole rivers and not at specific points only. As part of this transition, the wflow_sbm model has been developed for the Rhine and is currently undergoing experimental operational testing. The wflow_sbm model (Figure 2-3) is a spatially distributed hydrological model based on the topog_sbm model (Vertessy & Elsenbeer, 1999) with a kinematic wave approach for lateral subsurface and overland and river flow processes (Imhoff et al., 2020).

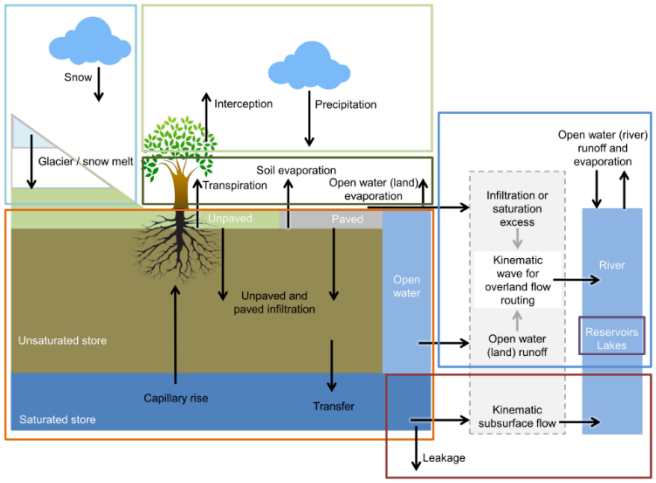


Figure 2-3 Overview of the different processes and fluxes in the wflow_sbm model (van Verseveld et al., 2023).

Although the Wflow-Rhine model has not yet been implemented in operational settings, two sets of experimental results are available for assessing its simulation and forecasting capabilities. One is the simulation using the ERA5 dataset[3], the other is the forecasting with the SEAS5 dataset[4], a seasonal forecast product from ECMWF. These results will be used for comparison purposes in this study.

[3] https://www.ecmwf.int/en/forecasts/dataset/ecmwf-reanalysis-v5
[4] https://www.ecmwf.int/en/forecasts/documentation-and-support/long-range

2.4 Deep learning models for operational hydrological forecasting

DL techniques have gained significant attention for their ability to accurately capture the complexity of highly non-linear systems. In the field of streamflow modeling, more recent studies have found DL techniques, such as Long Short-Term Memory (LSTM), to be a promising approach, providing improvements in prediction accuracy, scalability, and regional generalization compared to conventional conceptual models (e.g., Mosavi et al., 2019). LSTM models, specifically designed for processing sequential data like time series, have been successfully applied by Kratzert et al. (2019) in over 500 basins across the United States, demonstrating enhanced discharge predictions compared to conceptual models. A detailed description of LSTM architecture can be found in Kratzert et al. (2018). And a visualization of the standard LSTM cell is shown in Figure 2-4.

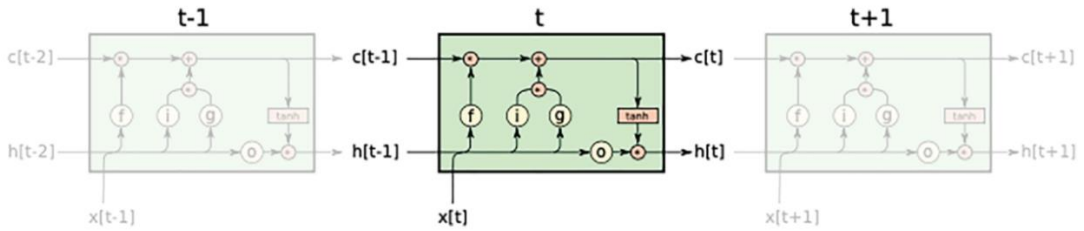


Figure 2-4 Visualization of the standard LSTM cell, where c[t] denotes the cell state at time step t, h[t] the hidden state, x[t] the input. f stands for the forget gate, i for the input gate, g for the cell update, and o for the output gate (Kratzert et al., 2019).

This study aims to design and apply DL models based on LSTM architectures. There are two critical considerations when utilizing LSTM-based models for this research. First, methods for effectively leveraging both historical observation and forecast data within a single DL model need to be explored to integrate them into an operational framework. Second, the challenge of multi-step time series forecasting using DL models needs to be addressed. Several strategies to tackle these issues are presented below.

2.4.1 Leveraging both historical observation and forecast data

In addressing the challenge of incorporating both historical observation and forecast data, two potential methods have been identified based on the literature review and expert knowledge.

The first method involves using a single LSTM model, which is the most commonly used approach by far. During the training phase, historical observation data is utilized, while during the inference phase, forecast data is applied. This approach has been employed by Hunt et al. (2022) in their study on streamflow prediction in the western United States, where the LSTM models were trained using ERA5 reanalysis data as historical "observation", and during the operational phase, forecast variables from the ECMWF were used.

The second method entails using two LSTM models, with one LSTM processing historical observation data and another LSTM processing forecast data. The two LSTMs can be connected either in parallel or in a cascade. Figure 2-5 and Figure 2-6 illustrate the model architecture for this method.

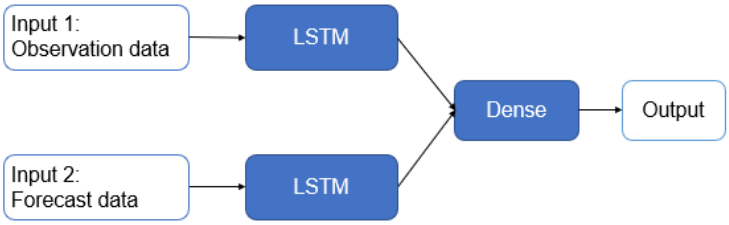


Figure 2-5 Illustration of two LSTMs in parallel. “Dense” means dense layer in DL models.

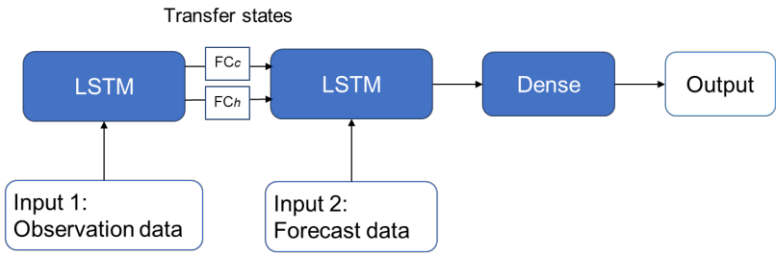


Figure 2-6 Illustration of two LSTMs in cascade. “Dense” means dense layer, and “FC” represents fully connected layer in DL models.

The parallel configuration is a straightforward approach where one LSTM processes the observation data while the other LSTM handles the forecast data. The two LSTMs independently learn representations from their respective inputs. The representations are concatenated and then fed into a Dense layer, which aids in generating the final outputs.

The cascade configuration is inspired by the historical and forecast modes of operation in physically-based operational streamflow forecast model discussed in section 2.1. It is similar to the encoder-decoder architecture which is commonly used for sequence-to-sequence (seq2seq) problems like language translation. The first LSTM sequentially processes observation data from the past days, taking the variables such as historically observed precipitation, temperature, and potential evaporation. It runs until the current time, which is the forecast initialization time. The final cell state and hidden state of the first LSTM (represented as c[t] and h[t] in Figure 2-4) are passed through a fully connected layer, and the resulting “transferred states” are used as the initial cell state and hidden state for the second LSTM. The second LSTM then processes forecast data at each lead time, taking variables like forecasted precipitation, temperature, and potential evaporation.

In the cascade LSTM approach, the first LSTM acts as the historical mode used to initialize the model’s storages and establish the initial conditions. The cell states in LSTM can be interpreted as storages in physically-based hydrological models (Kratzert et al., 2018). The second LSTM operates as the forecast mode, utilizing the internal model states (i.e., "transferred states") from the end of the historical run (first LSTM) as initial conditions for the forecast run. The use of fully connected layers to transfer states between the two LSTMs is inspired by Gauch et al. (2021) where the transferred states allowed modeling at multiple timescales. Notably, Google’s operational flood forecasting system also adopts a similar model structure for stage forecasting (Nevo et al., 2022).

2.4.2 Multi-step time series forecasting using DL models

Multi-step ahead forecasting remains an ongoing challenge in time series forecasting. Ben Taieb et al. (2012) provide a comprehensive review and comparison of strategies for addressing this issue in neural network modeling. Two strategies, MIMO and recursive strategy, have been found to be useful for this study.

The Multi-Input Multi-Output (MIMO) strategy is motivated by the need to capture the stochastic dependencies between future values, which can significantly impact forecast accuracy (Ben Taieb et al., 2012). In this approach, the forecasts for all steps are generated simultaneously. This strategy aligns well with LSTM models, as LSTM can output a value for each step it processes. Therefore, implementing the MIMO strategy in LSTM models is straightforward.

The recursive strategy is the oldest and most intuitive forecasting strategy. It involves training a single model to perform one-step ahead forecasts. When conducting multi-step ahead forecasting, the model is first used to forecast the first step, and the value just forecasted is incorporated as part of the input variables for predicting the subsequent step (using the same one-step ahead model). This process continues until the entire forecasting horizon is covered. However, the recursive strategy may encounter challenges in multi-step ahead forecasting tasks due to error accumulation. Errors in intermediate forecasts can propagate forward, influencing subsequent forecasts.

A study conducted by Lam et al. (2022) provides insights into modifying the recursive strategy. In their research, they developed a multi-step forecast autoregressive model called "GraphCast" for weather simulation, where they embedded the “autoregressive” or “recursive” idea within the model architecture. The model is trained on the full forecasting horizon, allowing it to learn the dependencies and dynamics of the time series data across multiple steps, thus helping to mitigate error propagation.

The insights from the aforementioned studies provide guidance for designing DL model architectures to be utilized in this study. In section 3, a comprehensive description of the specific DL model architectures employed in this research will be provided.

3 Materials and methods

In this chapter, information on the study area is introduced in section 3.1. Information on the datasets and sources is presented in section 3.2. A description of the DL model architectures designed and employed in this study is provided in section 3.3. General data processing steps are described in section 3.4.

3.1 Study area

The Rhine originates in Switzerland, flowing along 1230 km course before discharge into the North Sea. The Rhine basin has an area of 185,000 km^2, covering major parts of Switzerland and Luxembourg, and parts of Germany, France, and the Netherlands. The topography of the basin varies from 4000 m in the Alps to 6 m below sea level in the Netherlands.

The Rhine basin can be divided into nine subbasins (Figure 3-1), displaying different discharge behaviors. The southern alpine area is a “snow regime”, which is characterized by the interplay of winter snow cover, summer snowmelt, and relatively high summer precipitation. As a result, low water events occur mainly in winter, while flood events mainly in summer. On the other hand, subbasins such as Neckar, Main, and Mosel, which drain the low mountain regions, exhibit a “rain regime”. This regime is characterized by a dominance of winter flood and summer low water. In areas downstream of the Rhine, such as Cologne and Lobith, where the snow regime and rain regime overlap, a "combined regime" is observed. The discharge is more evenly distributed throughout the year (International Commission for the Protection of the Rhine, 2018).

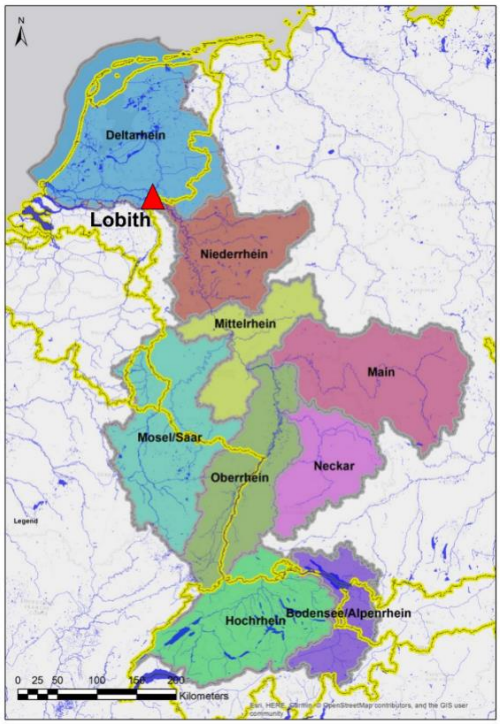


Figure 3-1 Nine subbasins of the Rhine River basin (Deltares, 2019) and the location of Lobith.

The river discharge climatology at Lobith based on different 10-year periods is shown in Figure 3-2. During the summer months, more than 70% of the discharge at Lobith originates from the Alps (Middelkoop & van Haselen, 1999). Less is from other parts of the basin, as most of summer precipitation at other subbasins evaporates before it reaches the river. Average discharge at Lobith is highest in winter, most of which is from tributaries in subbasin Neckar, Main and Mosel with intense rainfall and low evaporation. Only 30% of the discharge at Lobith during winter months is from the Alps, as winter precipitation falls as snow (Middelkoop & van Haselen, 1999).

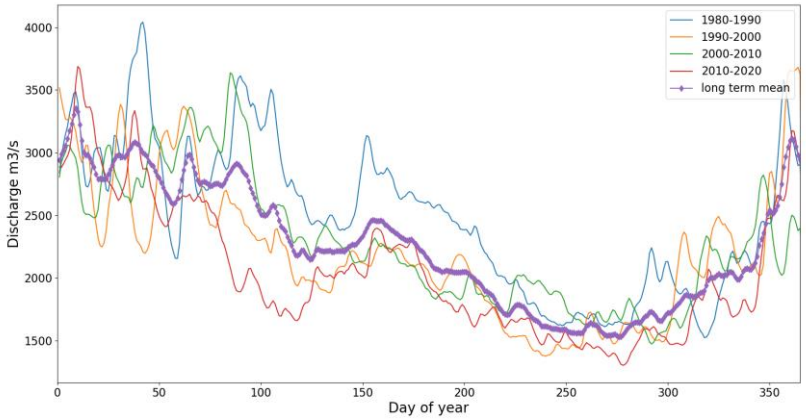


Figure 3-2 River discharge climatology at Lobith based on different 10-year periods. (Data source: Rijkswaterstaat)

3.2 Datasets

3.2.1 Streamflow at Lobith

Streamflow is the final output of the models employed in this study. Although the primary focus of the study is on streamflow drought, the decision has been made to utilize the model for forecasting the entire streamflow time series. Subsequently, the model performance is evaluated specifically during the dry period, which spans from April to September when streamflow droughts are most likely to occur at Lobith. For this study, streamflow drought at Lobith is defined as instances when the streamflow falls below the river discharge criteria outlined in Table 1-1.

There are two primary reasons for the decision not to rely on DL techniques to directly output streamflow droughts. Firstly, droughts are relatively infrequent events, and as a result, the dataset available for streamflow drought time series modeling can be limited and biased. This poses challenges for DL models, which typically require extensive datasets to train on. Secondly, Long Short-Term Memory (LSTM) excels at processing lengthy time series data as it can retain information about important aspects of earlier time periods for modeling target variables in subsequent time periods. By solely focusing on modeling streamflow droughts during the dry season, the LSTM's ability to leverage its long-term memory for events occurring in the wet season, such as snow accumulation which significantly influences summer flow at Lobith, would not be utilized effectively. Therefore, this study employs the DL model to forecast the complete streamflow time series and then concentrates on evaluating its

performance during the dry season to assess its effectiveness in forecasting streamflow drought.

For this study, streamflow observations are essential for training the DL models, as well as to assess model performance. Observations are available from Rijkswaterstaat's Waterinfo website[5], on a daily scale from 1979 to the present.

3.2.2 Meteorological parameters

For physically-based hydrological models, specific input parameters are required. The wflow model, for instance, requires precipitation, air temperature, and potential evaporation time series as forcing inputs. On the other hand, DL models also rely on the input parameters to describe the meteorological conditions over time, although the specific choice of forcing parameters may vary. In this study, three parameters are used, i.e., total precipitation, 2 meter temperature and potential evaporation. These parameters will be derived from the EOBS and ERA5 datasets, which provide observation data. Additionally, as part of the study, the performance of both models will be evaluated under operational conditions. To achieve this, forcing data from forecast products, including SEAS5 and ENS extended from European Centre for Medium-Range Weather Forecasts (ECMWF), will be incorporated. It is important to note that the forecast products used do not directly provide potential evaporation information. Therefore, the Makkink method (de Bruin, 1987) is applied to compute potential evaporation on a grid scale using 2 meter temperature and incoming shortwave radiation, both of which are directly available from the forecast products.

EOBS: The EOBS dataset is based on observations from meteorological stations across Europe which are provided by the National Meteorological and Hydrological Services (NMHSs) and other data holding institutes. Daily observational meteorological parameters are available from 1950 to 2022, with spatial resolutions of 0.1° and 0.25°.

ERA5: The ERA5 dataset is the climate reanalysis (fifth generation) of ECMWF providing atmospheric parameters with global coverage. All parameters are available from 1979 to present, a back-extension to January 1950 is already available. ERA5 has a spatial resolution of 0.25 degree (~ 31km) and provides forcing time series on hourly time steps. The data origins from the reanalysis of observations and the model output from the ECMWF Integrated Forecast System.

SEAS5: The SEAS5 is the fifth generation of the ECMWF seasonal forecasting system. It comprises ensembles of individual forecasts coupled to an ocean model and post-processed products of average conditions (e.g. monthly averages) with the associated uncertainty. Products are initialized at the first day of each month, and provide forecasts up to 7 months ahead, with daily temporal resolution. The SEAS5 data used in this study has been bias corrected. For simplicity, this report omits the "bias-corrected" notation for SEAS5.

ENS extended: The ENS extended product is generated by the ENS model from ECMWF. The ENS model generates ensemble of forecasts which provide an estimate of the reliability of a single forecast. The ENS extended product is the extension of ENS up to 46 days. It is initialized

[5] https://waterinfo.rws.nl/#!/nav/index/

twice a week, comprising ensembles of individual forecasts and post-processed products of average conditions (e.g. weekly averages) and the associated uncertainty.

For further details and in-depth information about the forecast products provided by ECMWF, the forecast user guide is available for reference[6].

3.2.3 System storage parameters

In addition to meteorological parameters, the contribution of discharge at Lobith is influenced by water storage in snowpack and large lakes, particularly in Switzerland, during dry periods (Demirel et al., 2013). Therefore, snow and lake level data are selected as potential additional input parameters for the DL models in this study.

To incorporate snow data, the ERA5 dataset will be used, which provides essential snow-related parameters such as snowfall and snow depth.

Lake Constance in Alpine region has the most significant influence for the Rhine. Other reservoirs outside the Alpine region have less uniform storage management objectives and less significant influence for the Rhine (International Commission for the Protection of the Rhine, 2018). Therefore, in this study, the water level of Lake Constance obtained from Bundesamt für Umwelt BAFU[7] will be used to represent the lake storage parameter.

A summary of the datasets used in this study is presented in Table 3-1.

Table 3-1 Summary of the parameters and datasets used in this study.

Category	Parameter	Symbol	Source	Type
Streamflow at Lobith	Discharge	*Q*	Rijkswaterstaat (RWS)	Observation
Meteorology	Total precipitation 2 meter temperature Potential evaporation	*tp* *t2m* *pev*	EOBS, ERA5 SEAS5, ENS_extended	Observation Forecast
Snow	Snow depth Snowfall	*sd* *sf*	ERA5	Observation
Lake Constance	Water level	*wl*	BAFU	Observation

3.3 Model architectures

In this study, LSTM-based model architectures are chosen for operational hydrological forecasting. Four model architectures are designed and tested specifically for the purpose of this study. Each of these architectures utilizes two groups of LSTMs, which effectively incorporate both historical observation and forecast data, and connected either in parallel or in cascade. Furthermore, all four architectures are designed to output the 46 steps of

[6] https://confluence.ecmwf.int/display/FUG/Forecast+User+Guide
[7] https://www.hydrodaten.admin.ch/de/2032.html

forecasting simultaneously. In this section, a detailed description of the four model architectures is provided.

3.3.1 Model 1

Model 1 (Figure 3-3) is designed with two parallel LSTM layers, which are subsequently connected to a Dense layer. LSTM-1 is responsible for sequentially processing the historical data from the past L days (also called the "look back window"), where meteorological and hydrological variables are provided as inputs at each time step. LSTM-1 operates until the current time (defined as the initialization time, see section 2.1) and generates a final output. LSTM-2 processes the future 46 days of data, taking only the forecast meteorological variables as inputs at each time step, and returns the full sequence of outputs. It should be noted that for inputs of LSTM-2, only forecast meteorological variables such as *tp*, *t2m*, *pev*, *sd* and *sf* are used. No hydrological variables such as lake water levels are used, as their forecast data is normally not available. The outputs from LSTM-1 and LSTM-2 are concatenated and fed into the Dense layer. This Dense layer produces 46 predictions simultaneously as the final output of Model 1.

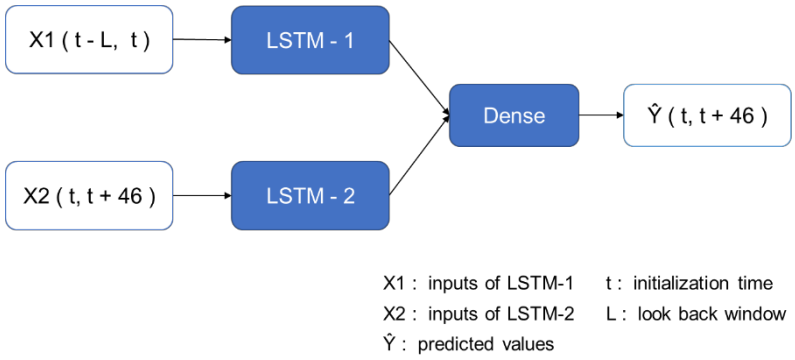


Figure 3-3 Illustration of Model 1 architecture. Note that X1 might include historical discharge Y from past L days. For simplicity, notation X1 is used to stand for all inputs of LSTM-1.

A less extensive hyperparameter tuning is conducted for this model architecture, as the hyperparameters of the LSTM model for streamflow prediction have been studied and optimized by several studies (Gauch et al., 2021; Kratzert et al., 2018, 2019; Nevo et al., 2022). A detailed overview on hyperparameters and settings of Model 1 is provided in Appendix B.

3.3.2 Model 2

Model 2 (Figure 3-4) is constructed with two LSTMs arranged in a cascade configuration, followed by a Dense layer. LSTM-1 processes data from the past L days sequentially. The final cell state and hidden state of LSTM-1 are passed through a fully connected layer (FC), and the resulting "transferred states" are used as the initial cell state and hidden state for LSTM-2. LSTM-2 processes meteorological data of the future 46 days, and returns the full sequence of outputs. The outputs of LSTM-2 are then fed into the Dense layer, which produces the final output of 46 predictions simultaneously. Details on hyperparameters and settings of Model 2 are presented in Appendix B.

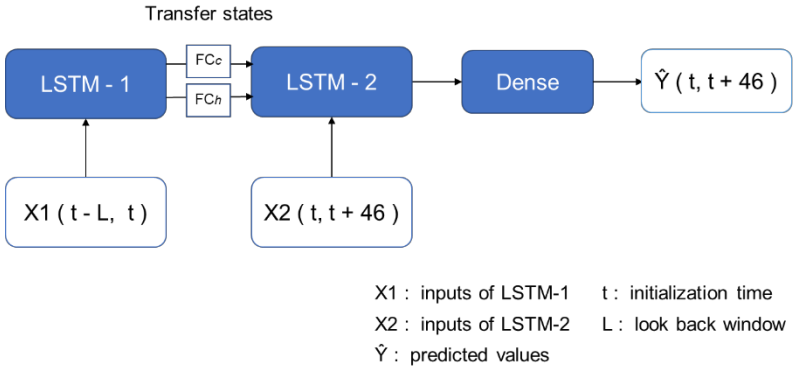


Figure 3-4 Illustration of Model 2 architecture. Note that X1 might include historical discharge Y from past L days. For simplicity, notation X1 is used to stand for all inputs of LSTM-1.

3.3.3 Model 3

Model 3, shown in Figure 3-5, is an extension of Model 2. The key difference from Model 2 lies in the handling of the final hidden state of LSTM-1 after passing through the FC layer. In Model 3, the final hidden state is not only used as the initial state for LSTM-2 but is also distilled using a Dense layer to lower dimension as “history representation”. This “history representation” is then concatenated with input X2 at each time step in LSTM-2. And LSTM-2 returns the full sequence of outputs to Dense-1. By doing this, LSTM-2 is explicitly prompted to retain and utilize the initial state obtained from LSTM-1 throughout its processing.

The intention behind this approach in Model 3 is to explicitly encourage LSTM-2 to preserve and utilize the initial state obtained from LSTM-1 throughout its processing. The concept of the "history representation" is inspired by the work of Wang et al. (2019), where a sequence-to-sequence model is used with a vectorized history representation of dialog history to enhance response generation for generative conversational agents. Details on hyperparameters and settings of Model 3 are presented in Appendix B.

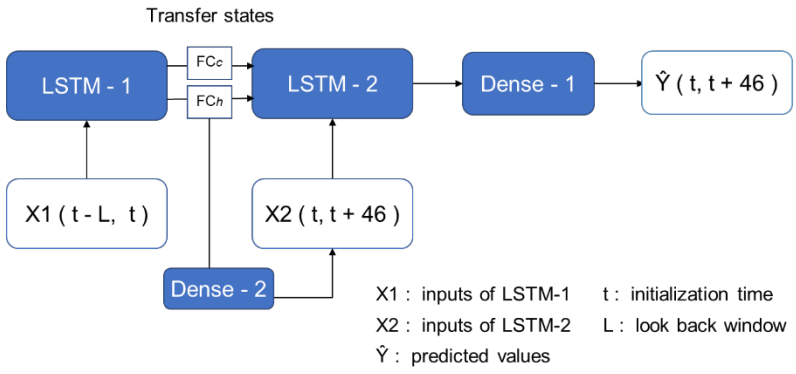


Figure 3-5 Illustration of Model 3 architecture. Note that X1 might include historical discharge Y from past L days. For simplicity, notation X1 is used to stand for all inputs of LSTM-1.

3.3.4 Model 4

Model 4 (Figure 3-6) follows a distinct structure compared to the other models. It consists of LSTM-1, similar to the previous models, as well as 46 individual LSTMs dedicated to each forecast step. Each LSTM in Model 4 generates one prediction. Similar to Model 2 and 3, the

final cell state and hidden state of the previous LSTM are utilized to initialize the subsequent LSTM. Additionally, akin to Model 3, the final hidden state of LSTM-1 is distilled to a lower dimension and concatenated with input X2 at each forecast step. The main difference of this model compared to the others, is the inclusion of observed true values (Y) or predicted values (Ŷ) from the previous R days as inputs for each forecast step. By including this additional information, Model 4 aims to leverage temporal dependencies and historical patterns of streamflow for improved forecasting accuracy. Details on hyperparameters and settings of Model 4 are presented in Appendix B.

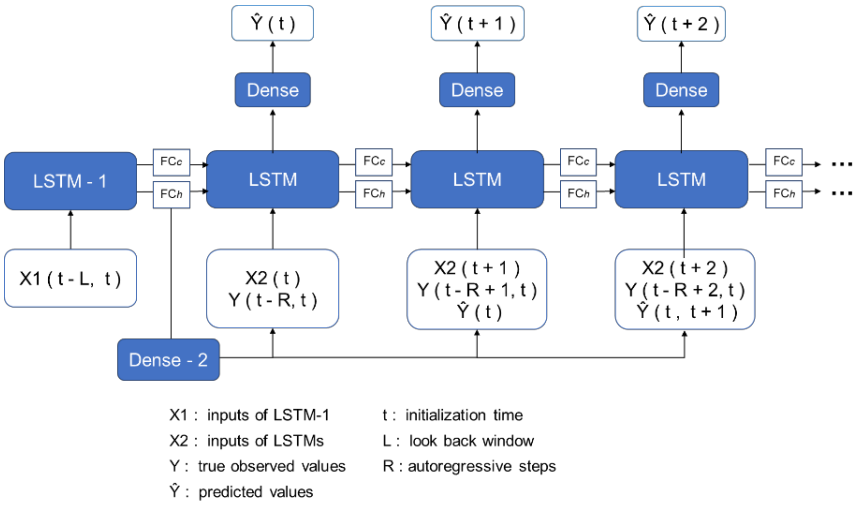


Figure 3-6 Illustration of Model 4 architecture. Note that X1 might include historical discharge Y from past L days. For simplicity, notation X1 is used to stand for all inputs of LSTM-1.

3.4 Data processing

The general preprocessing and postprocessing steps and methods used in this study are described in this section.

3.4.1 Preprocessing

In the preprocessing stage, several general steps are followed to prepare the data for the DL models. These steps include the log-transformation of discharge, the differencing of log-transformed discharge, the dataset splitting into training, validation, and testing sets, the scaling of parameters, and the preparation of sequences, as explained next in detail.

Log-transformation of discharge

To address the objective of forecasting streamflow drought in this study, the focus lies on low flows. As the discharge at Lobith exhibits a wide range, varying from below 1,000 m^3/s in the dry season to over 10,000 m^3/s in the wet season, it is necessary to apply a log-transformation to the discharge data. The log-transformation puts more emphasis on the low flow parts and therefore helps balance the representation of low flows and peak flows within the DL models.

The log-transformed streamflow should be denoted as *logQ*. However, for simplicity, this report omits the explicit "log" notation.

Differencing of log-transformed discharge

Several experiments carried out in this study will use time-differenced data, that is, the (log-transformed) discharge differences between consecutive time steps. The time-differenced log-transformed discharge should be denoted as *delta_logQ*. For simplicity, this report omits the "log" notation.

An example of the original streamflow, log-transformed streamflow, and time-differenced log-transformed streamflow at Lobith is illustrated in Figure 3-7.

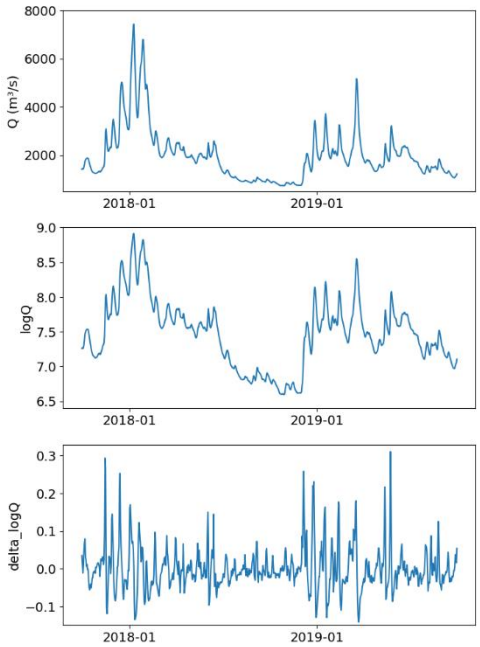


Figure 3-7 Example plot of the original streamflow, log-transformed streamflow, and time-differenced log-transformed streamflow at Lobith.

Split of dataset

The dataset is then split into training, validation, and testing subsets. During the development or experimental stage of DL model construction, training and validation subsets are used to facilitate the learning of data relations and obtain optimal model parameters. The trained model is then tested on the unseen testing dataset to provide a fair evaluation of its performance. The specific proportion of splitting between training+validation and testing datasets can vary based on the available amount of data. In this study, the exact splits will be described in each experiment's set-up. It should be noted that for operational use, the DL model should be retrained on the full dataset to generate the best real-time forecasts.

Scaling of variables

In this study, the DL models utilize multiple input variables that have varying value ranges and units. The model's sensitivity to different variables would be affected, leading to a potential decrease in model performance. Therefore, it is necessary to perform scaling on the input variables.

The scaling is done by standardization. The process begins by using the time series of the input variables solely from the training period. It is important to note that the scaler is created solely

based on the training dataset and does not include the validation or testing datasets. This is done to prevent any potential "data leaking" that could impact the fairness and accuracy of the model evaluation. From this training dataset, the mean $\bar{x}$ and standard deviation σ_x for each variable are calculated. Then, for every value in the time series, the standardized value x_{scaled} is computed using Equation (3-1). The scaler, which comprises the mean and standard deviation of each variable determined using the training dataset, is stored locally. The same scaling method is applied to the target variable.

$$x_{scaled} = \frac{x - \bar{x}}{\sigma_x} \tag{3-1}$$

Preparation of sequences

In time series modeling, the preparation of sequences is a crucial step to effectively utilize the available data and align with the specific model architecture. Based on the model architectures designed for this study, two different sequence lengths are required: 270 days (look back window) for LSTM-1 and 46 days (forecast horizon) for LSTM-2 (or the second group of LSTMs). A schematic of the sequence preparation is shown in Figure 3-8.

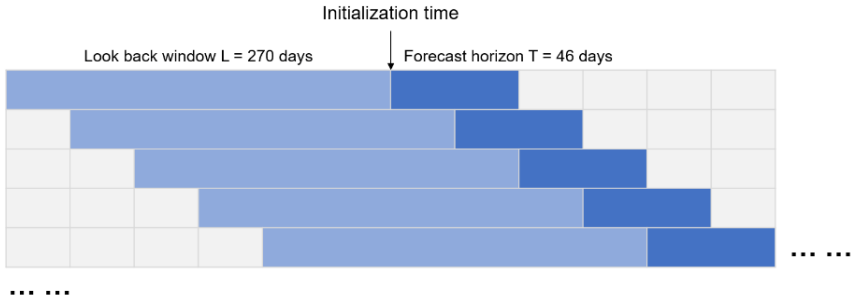


Figure 3-8 Schematic of the sequence preparation. For each initialization time, the data sequences of past 270 days (light blue) are for LSTM-1. The data sequences of future 46 days (blue) are for LSTM-2 (or the second group of LSTMs).

3.4.2 Postprocessing

In the postprocessing stage, there are general steps involved in handling the predicted values. These steps include back-scaling of the predicted value and back-transformation of the discharge.

Back-scaling of predicted value: In the preprocessing step, the target variable (predicted values) also undergoes scaling, resulting in standardized values y_{scaled}. The mean $\bar{y}$ and standard deviation σ_y of the observed target variable are determined using the training dataset. The scaler will then be applied to back-scale the modeled target using Equation (3-2).

$$y = y_{scaled} \times \sigma_y + \bar{y} \tag{3-2}$$

Back-transformation of discharge: In the preprocessing step, the discharge is log-transformed. However, after obtaining the modeled values, it is necessary to back-transform the modeled values to revert them to the right discharge values. Note that if the data is time-differenced during preprocessing, the results should first be added to the *logQ* from the previous day, and then back transformed to obtain the discharge values.

4 Experimental designs

In this chapter, the experimental designs for each sub-research question are presented.

4.1 Quantifying the impact of spatial resolution, input and target variables, and loss functions (SQ1)

The first sub-research question aims to determine the optimal combination of spatial resolution, input and target variables, and loss functions for forecasting streamflow drought using LSTM-based models. The following information provides important details on each of these subtopics, and the experiment setup for the first sub-research question (SQ1) is provided in the end.

4.1.1 Spatial resolution

Meteorological parameters, such as *tp*, *t2m*, *pev*, *sf* and *sd*, contain both spatial and temporal information. The spatial resolution of these parameters plays a crucial role in studying and utilizing spatial differences. In this study, the spatially distributed variables will be processed and applied in two different resolutions: either as mean values across the entire Rhine basin (referred to as basin mean approach), or as mean values over the eight subbasins upstream of Lobith as depicted in Figure 3-1 (referred to as subbasin mean approach).

4.1.2 Input and target variables

The model considers various input variables, including the commonly used meteorological variables (*tp*, *t2m* and *pev*). Additionally, this study explores incorporating additional information on snow, lake levels, and historical discharge at Lobith. The addition of the latter is uncommon in physically-based hydrological models where the input does not include streamflow but mainly forcing data and catchment properties.

Regarding the target variables, DL models offer greater flexibility compared to physically-based hydrological models in selecting target variables. This study will investigate two options:

- Training the model on discharge (Q) directly.
- Training the model on the time-differenced data (*delta_Q*).

The idea of predicting the time-differenced data arises from expert knowledge, indicating that DL models are more adept at capturing value changes rather than precisely modeling the exact values themselves. In this context, manual calculations are performed to obtain the first derivatives, while the subsequent derivatives are left to be handled by DL models. Several studies, such as Lam et al. (2022), have implemented this approach and demonstrated improved forecasting performance.

4.1.3 Loss functions

This study explores the impact of different loss weights on forecast performance for each lead time. The LSTM-based models used in the study can directly output 46 forecast steps, indicating that they are trained to predict 46 outputs simultaneously. The mean squared error (MSE) is employed as the default loss function with equal loss weight for each output.

However, considering the specific interest in forecasting streamflow droughts over longer lead times, this study aims to enhance the robustness of the forecast skill for extended periods. To achieve this, four types of loss weights are designed to experiment with, as illustrated in Figure 4-1.

- Type 0 represents the default scenario where the weights for different lead times are the same in the loss function.
- Type 1 employs increasing weights, where the weight for each output increases as the lead time extends.
- Type 2 adopts decreasing weights, assigning higher weights to shorter lead times and lower weights to longer lead times.
- Type 3 uses a U-shaped weight distribution, assigning more weights to both shorter and longer lead times.

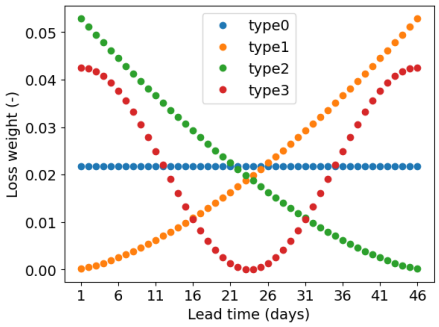


Figure 4-1 Four types of loss weights designed for the experiments.

4.1.4 Experiment setup

Experiment workflow:

The experiment workflow, as depicted in Figure 4-2, incorporates approaches to address SQ1. Firstly, **Experiment 1A** is conducted to investigate spatial resolutions. In this experiment, different spatial resolutions are combined with various input variables. Notably, the target variable *Q* and loss weight of type 0 are utilized and kept the same throughout Experiment 1A. The spatial resolution identified from Experiment 1A based on the model performance evaluation, is then used for Experiment 1B. In **Experiment 1B**, different target variables with various input variables are tested while maintaining the same loss weights as type 0. The target and input variables identified from Experiment 1B based on model performance evaluation, is then used for Experiment 1C. In **Experiment 1C**, different types of loss weights will be experimented.

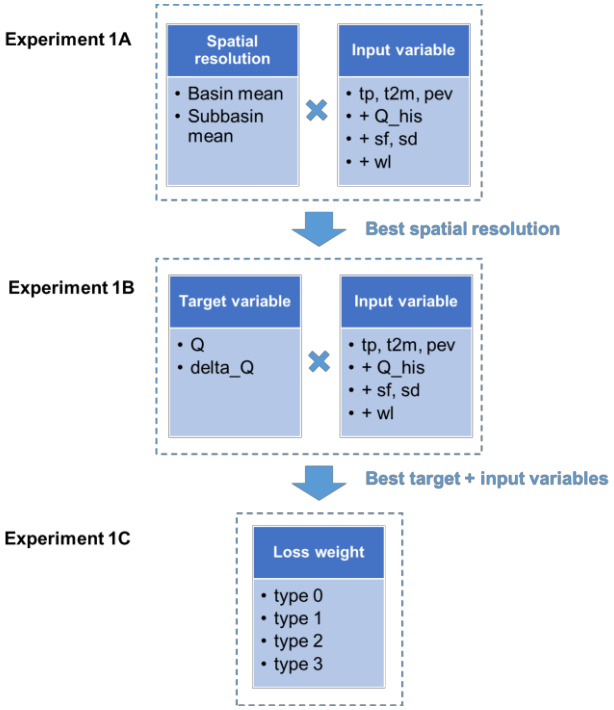


Figure 4-2 Schematic of the experiment workflow for SQ1. The multiplication factor here refers to the exploration of all possible combinations between the two sides.

Model architecture:

This experiment workflow is tested on all four model architectures, except that Experiment 1B is not tested for Model 4, as the architecture of Model 4 cannot be easily modified to model *delta_Q* due to the inclusion of observed values (Y) or predicted values ($\hat{Y}$) from the previous R days as inputs for each forecast step and the long training time and computer memory required compared to other three model architectures. Therefore, in all the studies conducted for Model 4, the target variable of *Q* is used.

Evaluation method:

The model performance evaluation is conducted during the testing period, against observed discharges. The evaluation specifically focuses on the results obtained during the dry season, which is defined as the period from April 1 to September 30 in this study. Two evaluation metrics **MAE** and **MAPE** are used. MAE stands for the mean absolute error in unit the same as target (m^3/s). MAPE stands for the mean absolute percentage error in unit 100%. Both MAE and MAPE are calculated for each time step (lead time) and averaged over the entire forecast time series within the dry season.

An overview of the experiment setup for SQ1 is provided in Table 4-1.

Table 4-1 Overview of the experiment setup for SQ1.

Experiment	Spatial resolution	Input variables	Target variable	Loss weight
Experiment 1A	Basin mean Subbasin mean	*tp, t2m, pev* *+ Q_his* *+ sf, sd* *+ wl*	*Q*	type 0
Experiment 1B	Best spatial resolution	*tp, t2m, pev* *+ Q_his* *+ sf, sd* *+ wl*	*Q* *delta_Q*	type 0
Experiment 1C	Best spatial resolution	Best combination	Best target	type 0 type 1 type 2 type 3
General information				
Model architecture tested	Model 1, Model 2, Model 3, Model 4 (except for Experiment 1B)			
Meteorology Dataset	*tp, t2m, pev*: EOBS *sf, sd*: ERA5			
Training-validation-testing period	Training: [1979-10-01, 2013-09-30] Validation: [2013-10-01, 2016-09-30] Testing: [2016-10-01, 2019-09-30]			
Evaluation metrics	MAE, MAPE			

4.2 Comparing different model architectures (SQ2)

The second sub-research question focuses on evaluating the effectiveness of various model architectures for this study, and comparing their performances by employing the optimal combination of spatial resolution, input and target variables, and loss functions identified in SQ1. To accomplish this objective, two steps are taken:

1) Cross-validation: Cross-validation is performed for each model architecture to obtain a robust estimate of the model's performance.
2) Comparative analysis: The performances of different architectures are compared to determine the DL model used for operational use in SQ3.

4.2.1 Cross-validation

In SQ1, the models were trained on the same data splits to identify the optimal combination of spatial resolution, input and target variables, and loss functions. However, in order to obtain a more reliable estimate of the model's performance, it is necessary to test the model on different data splits. This approach allows us to assess the robustness and stability of the model when trained and tested on different time series datasets. It helps determine if the model overfits the training data when trained on the entire dataset.

There are several cross-validation techniques suitable for time series forecasting models (Cerqueira et al., 2020). For this study, the “sliding window with gaps” method is chosen. This method has several advantages:

- This method maintains the temporal order of the data, which could provide better error estimations for time series data with potential non-stationarities.
- The size of the training sets remains constant, which ensures that each iteration in the cross-validation is comparable.
- The inclusion of gaps between training and testing sets enhances the independence between the two. This is particularly important when conducting cross-validation on LSTM models, as it helps evaluate the model's generalization capability.

An illustration and details of the cross-validation method can be found in Appendix C.

Cross-validation is performed on all four model architectures using the same datasets as in SQ1. The train-validation-testing splits for each iteration and the benchmark model are presented in in Table 4-2.

Table 4-2 Train-validation-testing period for cross validation.

	Train	Validation	Testing
Iteration 1	[1979-10-01, 2000-09-30]	[2001-10-01, 2004-09-30]	[2016-10-01, 2019-09-30]
Iteration 2	[1982-10-01, 2003-09-30]	[2004-10-01, 2007-09-30]	[2016-10-01, 2019-09-30]
Iteration 3	[1985-10-01, 2006-09-30]	[2007-10-01, 2010-09-30]	[2016-10-01, 2019-09-30]
Iteration 4	[1988-10-01, 2009-09-30]	[2010-10-01, 2013-09-30]	[2016-10-01, 2019-09-30]
Iteration 5	[1991-10-01, 2012-09-30]	[2013-10-01, 2016-09-30]	[2016-10-01, 2019-09-30]
Benchmark	[1979-10-01, 2013-09-30]	[2013-10-01, 2016-09-30]	[2016-10-01, 2019-09-30]

The model performance evaluation is carried out during the testing period, specifically focusing on the dry season, and comparing the model outputs against observed discharges. Similar to SQ1, the evaluation employs MAE and MAPE. In addition, the evaluation also analyzes the absolute error (AE) and absolute percentage error (APE) for each time step (lead time) over the entire forecast time series within the dry season. This analysis allows for an assessment of the variation and uncertainty of the model results at different lead times.

4.2.2 Comparative analysis

Following the cross-validation process, the model architectures that demonstrate robustness and stability are selected for further comparison. These chosen models are retrained on all available training data, and are subsequently evaluated on the testing data, following the same splits as specified in Table 3.8 for the benchmark model. To compare the models, the AE and APE for each time step across the entire forecast time series within dry season are analyzed. By conducting this comparative analysis, the model architectures that exhibit the skillful performance can be identified, thus could be selected for operational use in SQ3.

4.2.3 Experiment setup

An overview of the experimental setup for cross-validation and comparative analysis is provided in Table 4-3.

Table 4-3 Overview of the experimental setup for cross-validation and comparative analysis.

	Cross-validation (Experiment 2A)	Comparative analysis (Experiment 2B)
Model architecture tested	Model 1, Model 2, Model 3, Model 4	
Input variables	Optimal combination from SQ1	
Meteorology Dataset	*tp, t2m, pev*: EOBS *sf, sd*: ERA5	
Training-validation-testing period	See Table 4-2	Same as benchmark in Table 4-2
Evaluation metrics	MAE, MAPE, AE, APE	AE, APE

4.3 Comparing the DL model with physically-based models (SQ3)

The third sub-research question aims to compare the performance of the LSTM-based model identified in SQ2 with the physically-based models in forecasting streamflow drought. To conduct this comparison, three experiments were designed and executed based on the availability of forecast forcings and existing results from physically-based models.

- Experiment 3A: the DL model forecasting with observed meteorology forcing from ERA5 versus Wflow-Rhine simulation with the same forcing
- Experiment 3B: DL model forecasting with SEAS5 versus Wflow-Rhine forecasting with SEAS5
- Experiment 3C: DL model forecasting with SEAS5 versus FEWS-Rhine forecasting with ENS_extended

By conducting Experiment 3A and 3B, a general estimation of the forecast performance of the DL model relative to the wflow model can be obtained. Experiment 3C enables us to investigate the DL model's potential in generating past drought events when compared to the FEWS system.

4.3.1 Experiment 3A: DL model vs Wflow-Rhine with ERA5

Experiment 3A focuses on comparing the best of DL model with the best of wflow model. The best of DL model is the one that forecasts with historical observations of forcing (ERA5). The best of wflow model is hypothesized to be the one simulating with ERA5, which is to model streamflow one step ahead rather than carry out true forecasting. Details of this comparison experiment setup, including datasets used, meteorological parameters, training and comparison period, and evaluation methods, are outlined in Table 4-4.

It should be noted that only three meteorological parameters (i.e., *tp*, *t2m*, *pev*, excluding *sf* and *sd*) are used for consistency with the inputs of the wflow model. Additionally, since the DL model uses the past 270 days of data as input and does not involve any forecasting results within this period, the actual comparison period only start 270 days after 2016-10-01. As for evaluation, both MAE and MAPE with standard deviation are calculated for each LT and averaged over the entire forecast time series within the dry season.

Table 4-4 Experimental setup details of Experiment 3A.

	DL model	Wflow-Rhine
Input variables	Optimal combination from SQ1	*tp, t2m, pev*
Meteorology Dataset	X1: ERA5 X2: ERA5	ERA5
Training period	[1979-10-01, 2016-09-30]	n/a
Testing period	[2016-10-01, 2019-09-30]	
Evaluation metrics	MAE and MAPE with standard deviation (std)	

4.3.2 Experiment 3B: DL model vs Wflow-Rhine with SEAS5

Experiment 3B aims to evaluate the performance of the DL model in forecasting streamflow drought in comparison to the wflow model, using the same forecast forcing product. The experimental setup for this comparison is provided in Table 4-5.

Table 4-5 Experimental setup details of Experiment 3B.

	DL model	Wflow-Rhine
Input variables	Optimal combination from SQ1	*tp, t2m, pev*
Meteorology Dataset	Training mode: X1/X2: ERA5 Forecast mode: X1: ERA5, X2: SEAS5	Historical mode: ERA5 Forecast mode: SEAS5
Training period	[1979-10-01, 2016-09-30]	n/a
Initialization time	From 2017-10-01 to 2020-05-01, first day of each month	
Evaluation metrics	CRPS, CRPSS	

Regarding the experimental setup, there are a few clarifications to be made:

- Only three meteorological parameters (i.e., *tp*, *t2m*, *pev*, excluding *sf* and *sd*) are used in the experiments due to the unavailability of *sf* and *sd* in the SEAS5 dataset.
- The SEAS5 dataset provides forecasts up to 7 months ahead with a daily temporal resolution. However, in the experiments, only the first 46 days of forecasts are utilized.
- In the DL model for operational use, it is preferable to train the model on as much data as possible before initialization. However, for simplicity, the DL model is trained solely on the data from 1979 to 2016. This trained model is then employed for all forecasts.
- The initialization times for the experiments span from 2017-10-01 to 2020-05-01, at first day of each month. This results in a total of 33 samples, with 14 samples in dry season.
- The wflow model is forced with the full length (7 months) of the SEAS5 dataset, but for comparison purposes, only the first 46 days of forecast results are used. The initial states of the wflow model for each forecast are derived from the simulation results with ERA5, rather than the true observations.

The evaluation will focus on the results initialized during the dry season. In this experiment, both the DL model and the wflow model generate ensemble forecasts, which are probabilistic forecasts. To assess and compare their performance, the evaluation employs two metrics: Continuous Ranked Probability Score (CRPS) and Continuous Ranked Probability Skill Score (CRPSS).

CRPS quantifies the dissimilarity between the cumulative density function of the ensemble forecast and the Heaviside function of the true observation. The optimal value is 0, and the score increases with the increasing inaccuracy of the probabilistic forecast.

The CRPS can serve as a Skill Score (**CRPSS**) by comparing it to a reference forecast, which in this case is the Wflow-Rhine. The CRPSS is calculated using Equation (4-1). If the DL model forecast is perfect, its CRPS will be 0, and the CRPSS with regard to Wflow-Rhine's forecast will become 1. If both forecasts demonstrate equal accuracy, the CRPSS will be 0. If the DL model forecast outperforms the wflow forecast, the CRPSS will yield a positive value. Conversely, the CRPSS will be negative.

$$CRPSS = \frac{CRPS_{DL\ model}}{CRPS_{wflow-Rhine}} \tag{4-1}$$

4.3.3 Experiment 3C: DL model vs FEWS-Rhine

Experiment 3C aims to assess the capability of the DL model in generating past drought events in comparison to the existing operational system, FEWS-Rhine. The experimental setup for this comparison is outlined in Table 4-6.

Table 4-6 Experimental setup details of Experiment 3C.

	DL model	FEWS-Rhine
Input variables	Optimal combination from SQ1	*tp, t2m, pev*
Meteorology Dataset	Training mode: X1/X2: ERA5 Forecast mode: X1: ERA5, X2: SEAS5	Historical mode: n/a Forecast mode: ENS extended
Training period	[1979-10-01, 2017-09-30]	n/a
Forecasted events	From July to November in 2018	
Evaluation	Confusion matrix, recall, precision	

This experiment aims to compare the performance of both models in forecasting drought events specifically in the year 2018. The results of DL model from Experiment 3B will be used. The results of FEWS-Rhine forecasted with ENS extended (i.e., forecast line fews_hbv_ens_ext) are used for comparison. Note that the initialization times of the two models are not exactly the same, and only the initialization times of FEWS-Rhine which are close to that of the DL model are chosen for comparison.

To assess the performance, the model results are postprocessed into the drought or non-drought class using the river discharge criteria outlined in Table 1-1. For probabilistic forecasts, a threshold of 0.5 is applied to the drought occurrence probabilities. If the drought occurrence probability is equal or greater than 0.5, the result is classified as drought. If the drought occurrence probability is less than 0.5, the result is classified as non-drought. Subsequently, a confusion matrix is used to assess the performance of the modeled binary classification (drought and non-drought) in comparison to the actual classification based on observed discharges. This matrix categorizes the results into four categories: true positives (TP), false negatives (FN), false positives (FP), and true negatives (TN). From the confusion matrix, recall and precision values are derived. Recall, also referred to as hit rate or true positive rate, is calculated as TP/(TP+FN). Precision, also known as positive predictive value, is calculated as TP/(TP+FP).

5 Results and discussions

In this chapter, the results are presented and discussed in three sections: section 5.1 quantifies the impact of spatial resolution, input and target variables, and loss functions, section 5.2 compares the performance of different DL model architectures, section 5.3 compares the DL model with physically-based models. Each section addresses a sub-research question, first presenting the results and findings, followed by a discussion.

5.1 Quantifying the impact of spatial resolution, input and target variables, and loss functions (SQ1)

5.1.1 Experiment 1A: spatial resolution

This section presents the results for the experiment to investigate spatial resolutions, which is referred to as Experiment 1A.

The results of Experiment 1A, evaluating the model performance using Mean Absolute Error (MAE) and Mean Absolute Percentage Error (MAPE), are shown in Figure 5-1 and Figure 5-2. Note that the results presented here are only those of Model 3, while the results of other model architectures can be found in Appendix D. The spatial resolution experiment where the spatially distributed variables are processed as mean values across the entire Rhine basin, is referred to as basin mean approach. The spatial resolution experiment where the spatially distributed variables are processed as mean values over the eight subbasins upstream of Lobith as depicted in Figure 3-1, is referred to as subbasin mean approach.

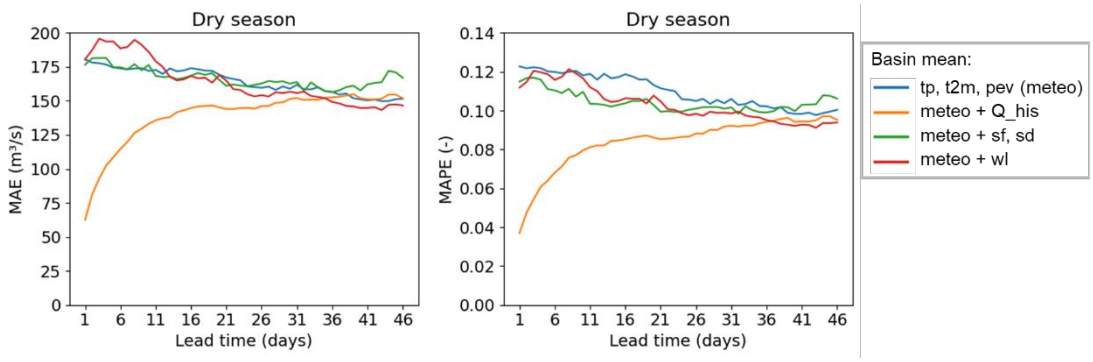


Figure 5-1 MAE and MAPE of Experiment 1A results from basin mean approach with various input variables (Model 3). Basin mean approach refers to the spatial resolution experiment where the spatially distributed variables are processed as mean values across the entire Rhine basin.

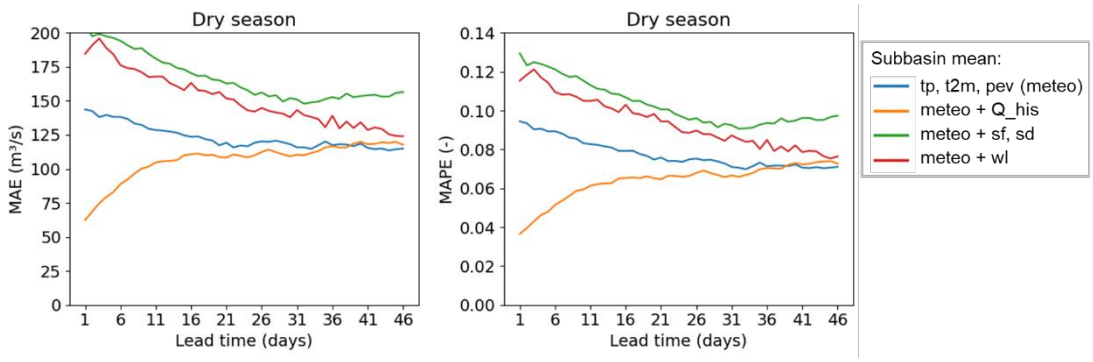


Figure 5-2 MAE and MAPE of Experiment 1A results from subbasin mean approach with various input variables (Model 3). Subbasin mean approach refers to the spatial resolution experiment where the spatially distributed variables are processed as mean values over the eight subbasins upstream of Lobith.

Considering the basin mean approach (Figure 5-1), when only general meteorology forcing parameters (*tp*: total precipitation, *t2m*: 2 meter temperature, *pev*: potential evaporation) are used, or when combined with snow information (*sf*: snowfall, *sd*: snow depth), or with lake level information (*wl*: water level), the average MAE for forecasting 46 steps ranges from around 150 to 180 m^3/s. The MAE exhibits a gradual decrease as the lead time (LT) increases. However, when historical discharges at Lobith (*Q_his*) are included, the average MAE for 46 steps drops to approximately 60-150 m^3/s. Interestingly, the MAE increases as the lead time increases from 1 to 16 days, ranging from around 60 m^3/s to 150 m^3/s. Beyond 16 days, the MAE stabilizes, showing minimal variation. Similar trends are observed in the MAPE results. Therefore, the addition of historical discharge significantly enhances model performance compared to using only meteorology forcing variables, whereas the inclusion of snow or lake level information does not yield significant improvements.

In the case of subbasin mean approach (Figure 5-2), when solely general meteorology forcing variables are utilized, the MAE decreases from approximately 150 m^3/s to 110 m^3/s as the lead time increases. However, when snow information is added, the MAE increases to a range of 150-200 m^3/s, gradually decreasing along the forecast horizon. Similarly, the inclusion of lake water level information results in an MAE range of 125-200 m^3/s, also decreasing along the forecast horizon. Conversely, when historical discharge (*Q_his*) is added as an input variable, the MAE decreases to a range of 60-110 m^3/s, and is increasing as the lead time increases. The MAPE results exhibit a similar pattern. Therefore, similar to the basin mean results, incorporating historical discharge improves model performance.

When comparing averaging over the whole basin versus for each subbasin, the performance generally improves across all experiments, except when adding snow information. For the models using only meteorology forcing variables, the MAE decreases by approximately 25 m^3/s for all lead times. In models incorporating *Q_his*, the MAE decreases by approximately 40 m^3/s for lead times beyond 16 days.

Consequently, for Model 3, employing subbasin mean approach generally enhances the results, albeit with varying degrees of improvement depending on the input variables. The results for other model architectures are presented in Appendix D, which also demonstrate improved performance when utilizing subbasin mean approach along with meteorology forcing and *Q_his*.

5.1.2 Experiment 1B: input and target variables

This section presents the results for Experiment 1B which aims to investigate the effects of different combinations of target and input variables on model performance.

Experiment 1B is conducted following Experiment 1A. The spatial resolution of subbasin mean approach is selected for this experiment. The MAE and MAPE results of Experiment 1B are illustrated in Figure 5-3 and Figure 5-4.

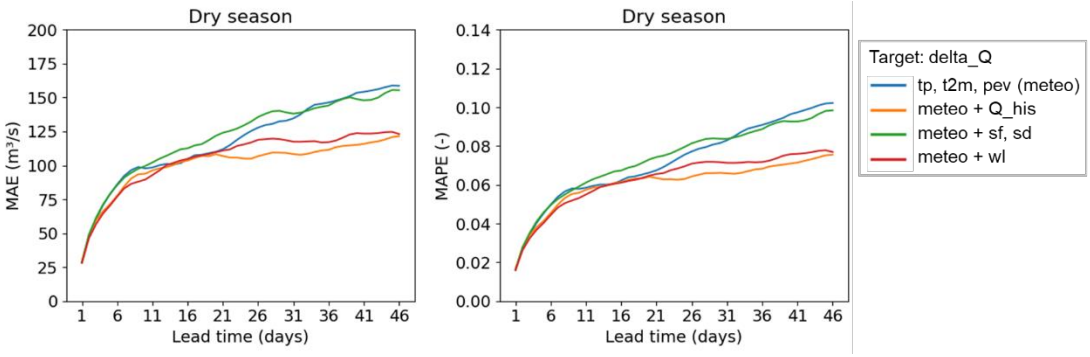


Figure 5-3 MAE and MAPE of Experiment 1B results from training the model on the time-differenced data (*delta_Q*) with various input variables (Model 3).

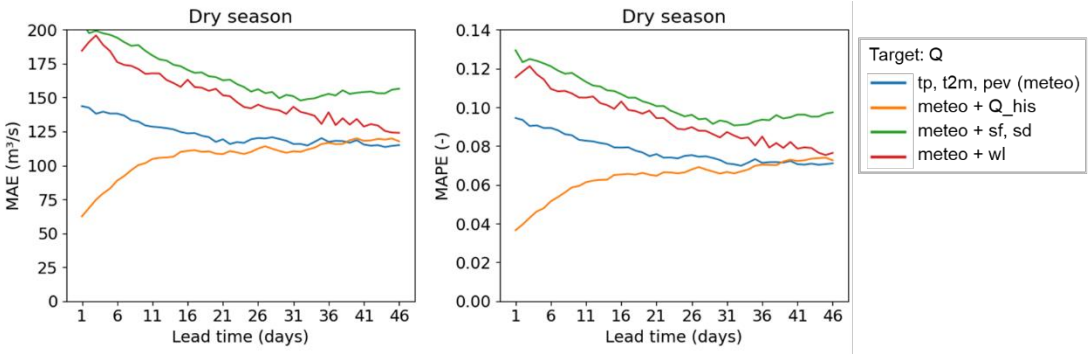


Figure 5-4 MAE and MAPE of Experiment 1B results from training the model on discharge (Q) directly with various input variables (Model 3). Note that Figure 5-4 is the same as Figure 5-2, included again in this section for comparison purposes.

Comparing the model performance when using time-differenced data (*delta_Q*) as the target variable versus discharge (Q) as the target variable, employing *delta_Q* as the target significantly enhances model performance, particularly for short lead times (1-6 days), regardless of the input variables considered. The MAEs all start around 25 m^3/s for LT=1 and increase to 110 m^3/s for LT=21. Beyond LT=21, the MAE continues to increase, albeit at different rates for different input variables. For the experiment that only use meteo as input variables, the MAE increases to a level that is higher than the model trained on *Q*. However, for *delta_Q* as the target variable, incorporating historical discharge and lake level information have the performances comparable to that of trained on *Q*.

Examining the performance of different input variables, including *Q_his* as an input variable yields better results, regardless of the target variable used (*Q_his* or *delta_Q*). Figure 5-5 consolidates the results of experiments involving *Q_his*. Training on *delta_Q* significantly improves the performance for the first 16 days, especially the initial 6 days, which is crucial for operational forecasting as it ensures the forecast starts from an accurate state. For lead

times beyond 6 days, the performance of the model trained on *delta_Q* is comparable to the model trained on *Q*. It should be noted that the model trained on *delta_Q*, after obtaining the predicted values of *delta_Q*, during the data post-processing phase, the *delta_Q* needs to be added to the discharge from the previous day, which introduces error propagation along the forecast horizon. Hence, the MAE increases more rapidly as the lead time increases compared to the model trained on *Q*.

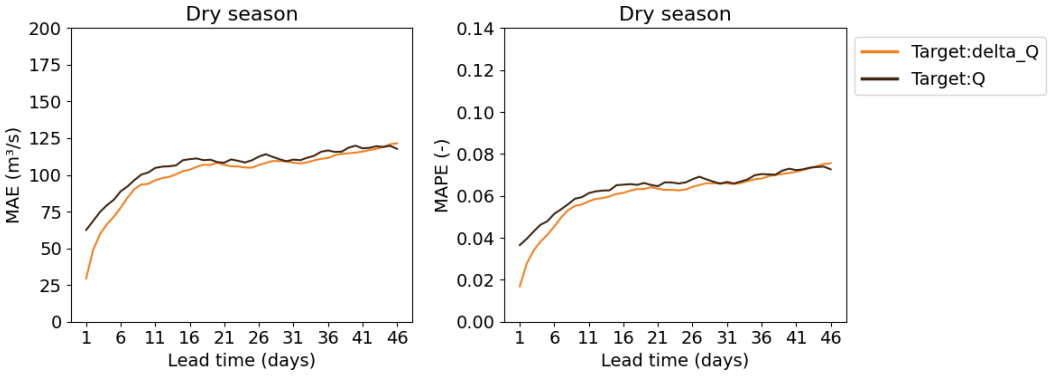


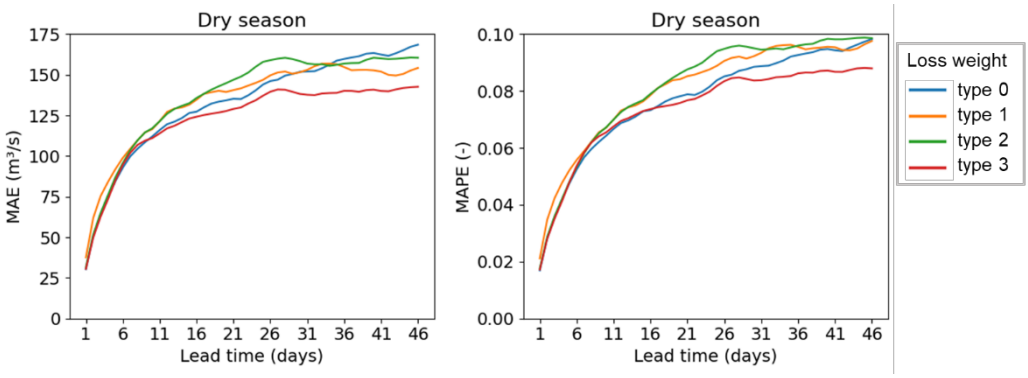
Figure 5-5 MAE and MAPE of Experiment 1B results on different targets variables (*delta_Q* and *Q*) when including historical discharge at Lobith (*Q_his*) as an input variable.

Consequently, in Model 3, the best performance is obtained by using *delta_Q* as the target variable and including *tp*, *t2m*, *pev* and *Q_his* as input variables. Similar results are observed in Model 1 and Model 2, as shown in Appendix D.

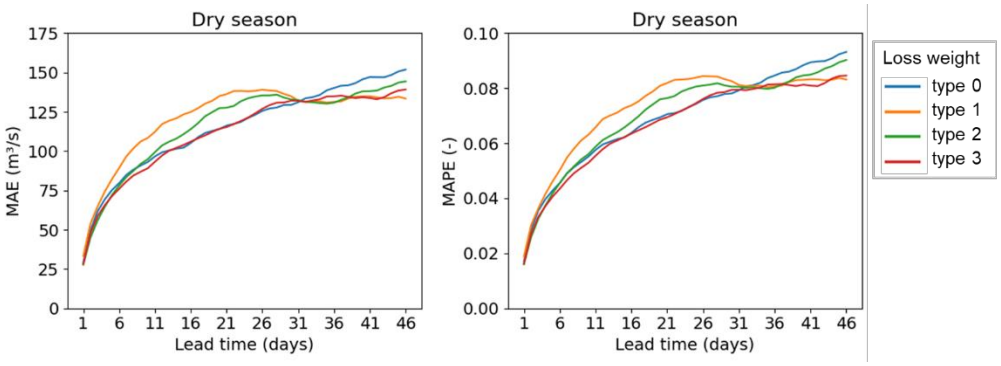
5.1.3 Experiment 1C: loss weight

This section presents the results for Experiment 1C which explores the influence of loss weights on model performance.

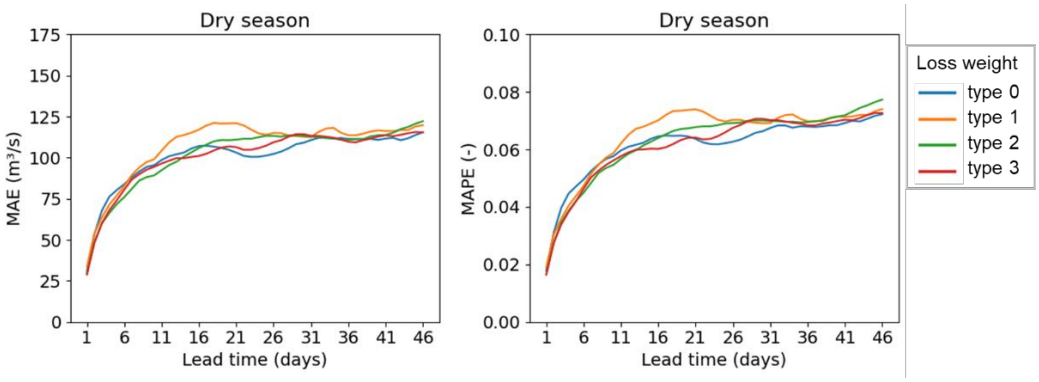
Based on experiments A and B, the spatial resolution of subbasin mean approach, along with the target variable *delta_Q* and the input variables *tp*, *t2m*, *pev*, and *Q_his*, are chosen for Experiment 1C. Notably, for Model 4, the target variable remains as *Q*. The MAE and MAPE results of Experiment 1C for different model architectures are shown in Figure 5-6. The outcomes vary across the model architectures.



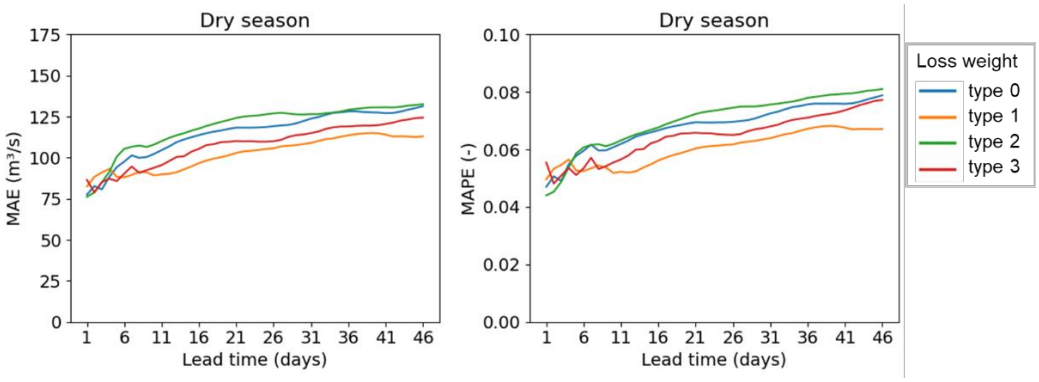
a) Model 1



b) Model 2



c) Model 3



d) Model 4

Figure 5-6 MAE and MAPE of Experiment 1C results on different types of loss weights for a) Model 1, b) Model 2, c) Model 3, and d) Model 4.

In the case of Model 1, there is low sensitivity to changes in loss weights for short lead times (1 to 11 days) but noticeable sensitivity for longer lead times (11 to 46 days). Allocating less weight to shorter lead times and more weight to longer lead times (type 1) does not yield improved performance for longer lead times. This may be attributed to error propagation along the forecast horizon when training on *delta_Q*. If the forecasts for shorter lead times perform poorly, the errors propagate to longer lead times. On the other hand, giving more weight to shorter lead times but less weight to longer lead times (type 2) worsens the performance for longer lead times compared to type 0. However, assigning more weight to both short and long lead times while reducing weight for middle lead times (type 3) improves the performance for longer lead times compared to type 0. Overall, type 3 generates better forecast results.

For Model 2, the forecast performance across different LTs varies a bit for different loss weight types. Loss weight of type 3 exhibits overall better forecast results.

For Model 3, the model performance across different forecast lead times shows relatively low sensitivity and no clear pattern to changes in loss weights. Only an increasing loss weight (type 1) demonstrates a decrease in performance for lead times between 6 and 26 days compared to other types.

For Model 4, the forecast performance varies for different loss weight types, particularly for type 1 and type 2. Increasing weights (type 1) outperform decreasing weights (type 2). This discrepancy can be attributed to Model 4 being trained on *Q*, where the performance for different lead times shows similarity with a gradual increase in MAE from 80 to 125 m3/s along the forecast horizon. By allocating more weight to longer lead times, where relatively higher errors are expected, the model is penalized more severely if it performs poorly at longer lead times. Consequently, the model learns to improve its performance in those segments. Therefore, the strategy of penalizing errors in longer lead times leads to an overall improvement in model performance as Model 4 is trained on all 46 steps together. Overall, type 1 generates better forecast results.

5.1.4 Discussion of results for SQ1

Regarding the spatial resolution, Experiment 1A clearly indicates that using subbasin mean approach for meteorology forcing significantly improves the model performance. This aligns with expectations since the discharge at Lobith originates from different subbasins in different seasons, and a subbasin spatial resolution allows the model to capture the information from various subbasins at different times. In contrast, using a basin mean approach would obscure this crucial information. This finding is consistent with previous studies (Khakbaz et al., 2012; Shah et al., 1996; Troutman, 1983) which have shown that considering the spatial variabilities of meteorological forcing, particularly precipitation, has a significant impact on the hydrologic response of basins.

Concerning the input variables, the results from Experiment 1A and 1B demonstrate that incorporating historical discharge at Lobith enhances the overall forecast skill. This improvement can be attributed to the model's ability to learn from past states and utilize that information to enhance the forecast performance, particularly for short LTs. This finding aligns with operational forecast approaches that rely on autoregressive or linear models, which employ historical discharge data as input for short-term forecasts due to the relatively high autocorrelation present. For longer LTs beyond 20 days, the autoregression approach becomes less effective as the autocorrelation drops below 0.5 (see Figure 5-7).

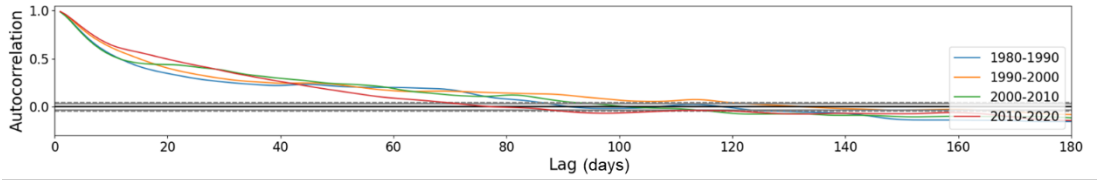


Figure 5-7 Autocorrelation of discharge at Lobith with different lags based on different 10-year periods. (Data source: Rijkswaterstaat)

The addition of snow information does not provide significant added value to the model performance. This may be because the inputs already include highly correlated variables such as total precipitation and air temperature, which capture the dynamics related to snow. Introducing snow data can lead to a degradation in deep learning model performance due to the high correlation among the inputs. Also, previous studies (e.g., Kratzert et al., 2018) have shown that DL models can already capture snow dynamics and its influence on discharge using precipitation and temperature inputs without explicit snow data.

The water level at Lake Constance (*wl*) exhibits varying performance when different target variables and model architectures are used. However, due to time constraints, this research does not delve into the details of *wl*. It is recommended to explore the impact and contribution of it in future studies.

For the target variables, Experiment 1B shows that training the model on *delta_Q* as the target variable enhances the forecast performance, particularly for short lead times (1-6 days), regardless of the input variables used. This finding aligns with expert knowledge suggesting that DL models excel at capturing value changes rather than precise value estimation. However, the improvement provided by training on *delta_Q* diminishes for longer lead times due to error propagation along the forecast horizon.

In Experiment 1B, it has been found that optimal performance is obtained when using *delta_Q* as the target variable, along with *tp, t2m, pev*, and *Q_his* as input variables. This finding specifically applies to the Rhine River at Lobith, where long-term continuously measured discharges are available. However, for Predictions in Ungauged Basins (PUB) where observed discharges are lacking, only general meteorology forcing (e.g., *tp*, *t2m*, *pev*) may be accessible. Figure 5-8 summarizes the results of experiments on different targets variables using only the three meteorology parameters as inputs from Experiment 1B. The figure shows that for short LT forecasts up to 21 days, utilizing *delta_Q* as the target with just the meteorology parameters produces quite nice forecasts, with MAE ranging from 25 m^3/s to 110 m^3/s. For longer LT forecasts beyond 21 days, the performance improves when using *Q* as the target variable instead of *delta_Q*. Therefore, based on these findings, it is feasible to carry out forecasts for ungauged basins through transfer learning, using trained models from gauged basins with only meteorology parameters as inputs.

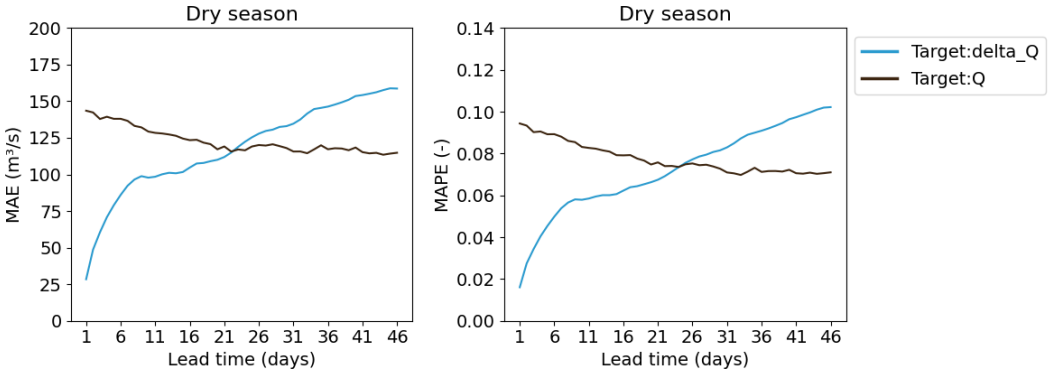


Figure 5-8 MAE and MAPE of Experiment 1B results on different targets variables (delta_Q and Q) when only general meteorology forcing parameters (*tp*, *t2m*, *pev*) are used as input variables. (Model 3)

Regarding the loss function, the impact of loss weights on model performance varies across different model architectures. Model 1, Model 2 and Model 4 exhibit different performance patterns across LTs with varying loss weights, while Model 3 shows relatively low sensitivity and no clear pattern to changes in loss weights. This discrepancy might be attributed to the already strong and stable performance of Model 3. And as a result, altering the loss weights for different lead times has minimal influence on the results.

5.2 Comparing different model architectures (SQ2)

The experiments in this section utilize the combinations of spatial resolution, input and target variables, and loss weights identified in section 5.1. Note that Model 3 shows relatively low sensitivity and no clear pattern to changes in loss weights. For SQ2 experiments, loss weight of type 2 is selected for Model 3. To provide a clear overview of these combinations, Table 5-1 presents a summary of the various configurations employed for each model architecture.

Table 5-1 The combinations of spatial resolution, input and target variables, and loss weights for different model architectures employed in the experiments in this section.

Model architecture	Spatial resolution	Input variables	Target variable	Loss weight
Model 1	Subbasin mean	*tp, t2m, pev, Q_his*	*delta_Q*	type 3
Model 2	Subbasin mean	*tp, t2m, pev, Q_his*	*delta_Q*	type 3
Model 3	Subbasin mean	*tp, t2m, pev, Q_his*	*delta_Q*	type 2
Model 4	Subbasin mean	*tp, t2m, pev, Q_his*	*Q*	type 1

5.2.1 Experiment 2A: cross-validation

This section presents cross-validation results for obtaining a robust model performance estimate.

The cross-validation process in this study consists of five iterations where each iteration involves using 70% of the total available time series data for training the model. After the five iterations, five trained models are obtained. Additionally, a sixth model is developed as a benchmark which is trained using all the available training data without any cross-validation splitting. The performance of the models is assessed using several metrics including MAE, MAPE, AE (Absolute Error), and APE (Absolute Percentage Error). In this section, an overview

of the performance for Model 1 and Model 3 is provided, while the complete results for Model 2 and Model 4 can be found in Appendix E.

For Model 1, Figure 5-9 indicates that the model performances of different iterations are relatively consistent for LT ranging from 1 to 21 days, despite being trained on different data splits. However, beyond 21 days, the performances of the iterations start to diverge. Notably, the iteration trained on the latest data splits (purple line) demonstrates better performance compared to other iterations and the benchmark. The box plots of AE (Figure 5-10) show that the median values for all iterations are below 150 m^3/s, while the upper quantiles are within 250 m^3/s. As the lead time increases, the uncertainties also increase, and after 21 days, the uncertainties of different iterations vary across different ranges. Thus, the cross-validation results indicate that Model 1 exhibits stable and robust performance for LTs between 1 and 21 days, but its stability decreases for LTs beyond 21 days.

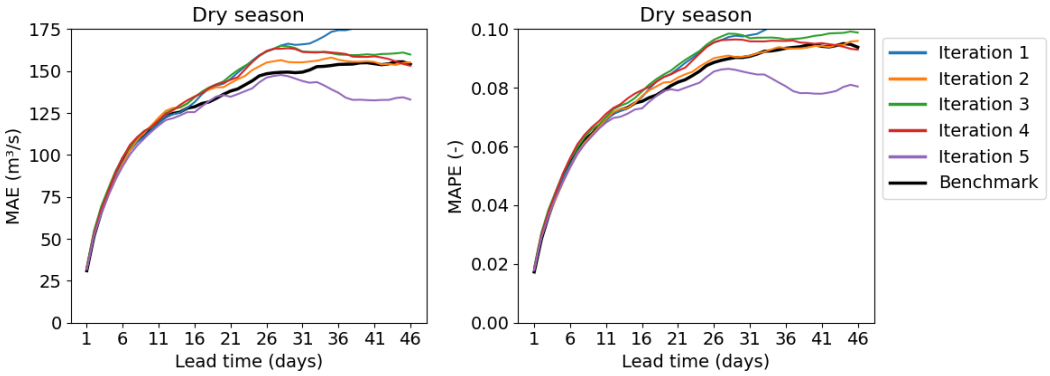


Figure 5-9 MAE and MAPE of cross-validation results for Model 1. Iteration 1 to 5 represent the five models trained on different cross-validation time series data splits. Benchmark represent the model trained using all the available training data without any cross-validation splitting.

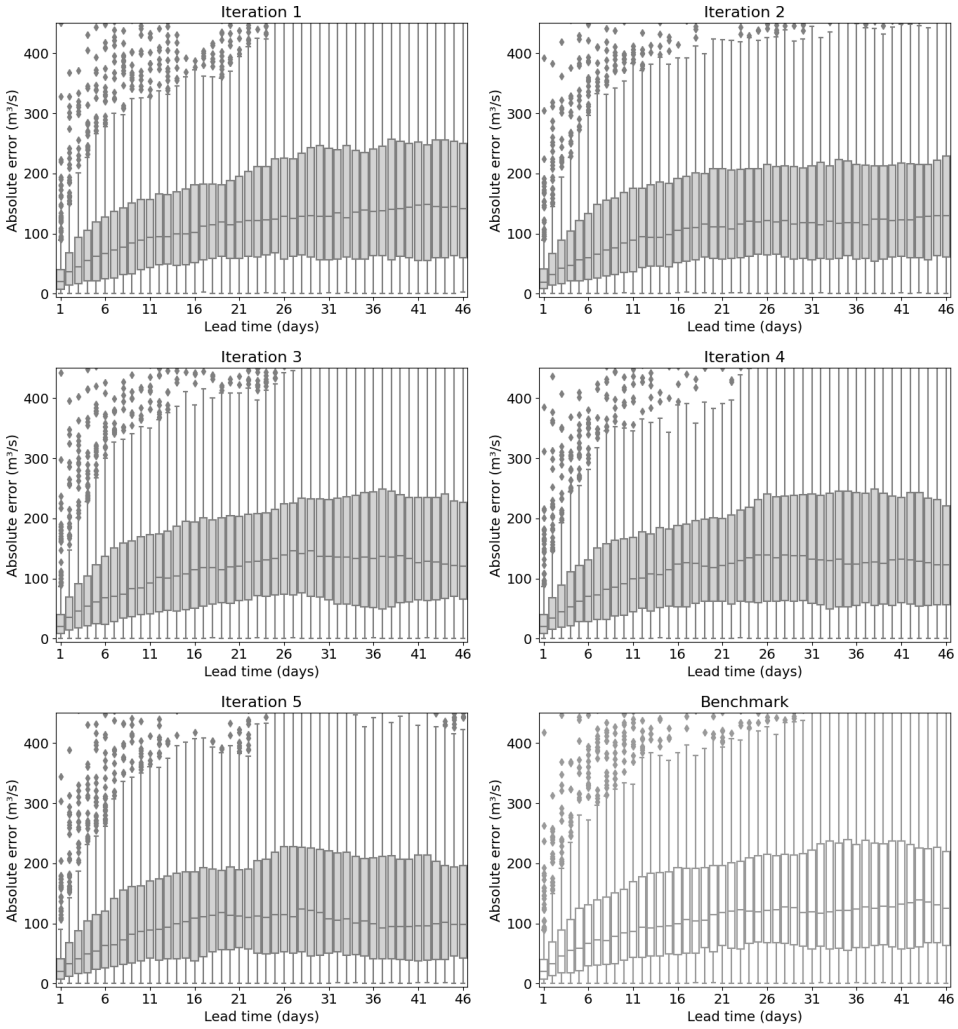


Figure 5-10 Absolute error (AE) of cross-validation results for Model 1. Iteration 1 to 5 represent the five models trained on different cross-validation time series data splits. Benchmark represent the model trained using all the available training data without any cross-validation splitting.

For Model 3, Figure 5-11 reveals that the performances of different iterations are similar for the initial 6 days. However, after 6 days, the performances start to differ, although the maximum difference in MAE between iterations remains around 25 m^3/s for all LTs. In contrast to Model 1, the iteration trained on the latest data splits (purple line) does not consistently achieve the best performance. The box plots of AE (Figure 5-12) show that the median values for all iterations are below 100 m^3/s, while the upper quantiles are within 200 m^3/s. Additionally, the maximum values for all LTs are lower than 450 m^3/s. The uncertainties among different iterations for all LTs are relatively consistent, and after 6 days, the uncertainty ranges for different LTs remain similar without significant increases along the forecast horizon. Consequently, the cross-validation results suggest that Model 3 demonstrates relatively stable and robust performance across the entire forecast horizon. Similarly, Model 2 and Model 4 (Appendix E) also exhibit consistent performance throughout the forecast horizon.

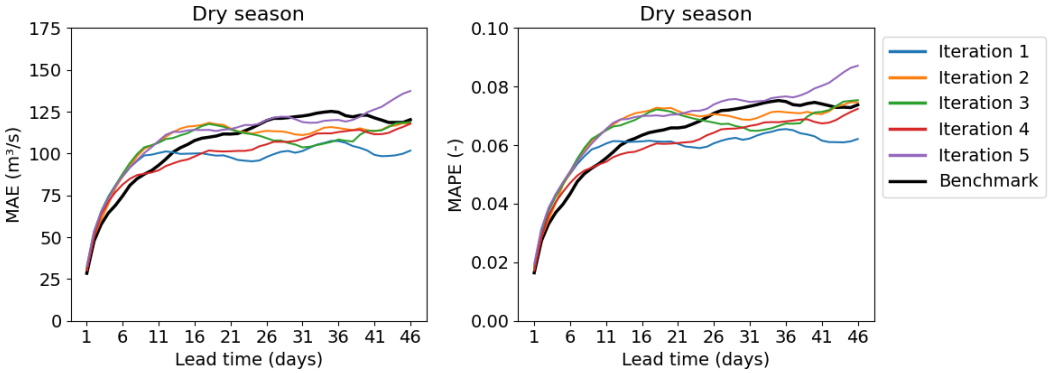


Figure 5-11 MAE and MAPE of cross-validation results for Model 3. Iteration 1 to 5 represent the five models trained on different cross-validation time series data splits. Benchmark represent the model trained using all the available training data without any cross-validation splitting.

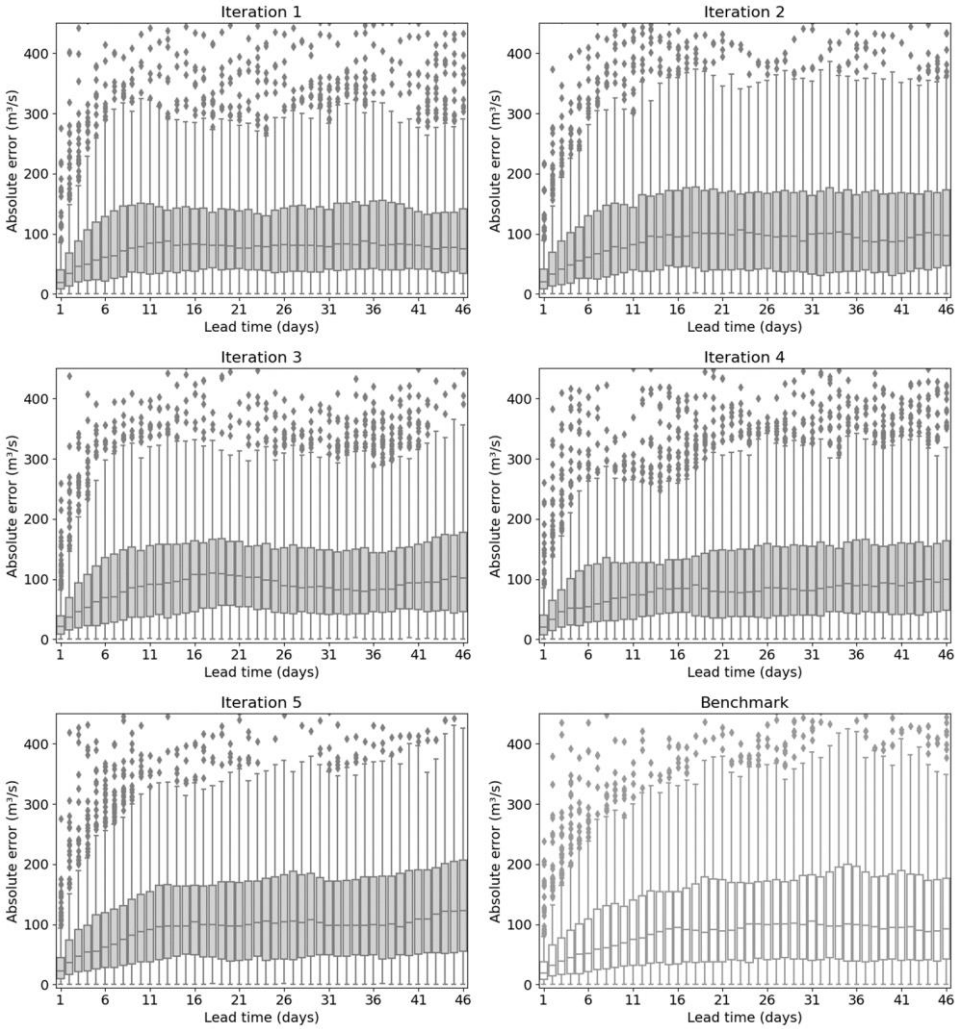


Figure 5-12 Absolute error (AE) of cross-validation results for Model 3. Iteration 1 to 5 represent the five models trained on different cross-validation time series data splits. Benchmark represent the model trained using all the available training data without any cross-validation splitting.

5.2.2 Experiment 2B: comparative analysis

Based on the cross-validation experiments, all four model architectures exhibit varying degrees of stability and robustness. Therefore, all models are selected for retraining using the entire available training data, followed by evaluation on the testing data. The evaluation is based on AE and APE metrics for different LTs, with the AE results presented in this section and the APE results provided in Appendix E.

Figure 5-13 illustrates that the median AE values for all LTs in each model architecture are approximately within 100 m^3/s. However, the uncertainties differ when examining the various LTs. For Model 1, the upper quantile of AE increases from 40 to 250 m^3/s as the forecast LT increases. The maximum AE value also experiences a rapid increase with longer LTs, exceeding 450 m^3/s after 26 days. In the case of Model 2, the upper quantile increases at a slower rate compared to Model 1, ranging from 40 to 200 m^3/s along the forecast horizon. The maximum AE values also increase but remain mostly below 450 m^3/s. For Model 3, the upper quantile increases from approximately 40 to 140 m^3/s at LT=6, after which it stabilizes. Similarly, the maximum AE values become relatively stable after 6 days, not exceeding 400 m^3/s. In the case of Model 4, the upper quantile gradually increases from 100 to 150 m^3/s along the forecast horizon, while the maximum AE values exhibit slow growth, not surpassing 350 m^3/s.

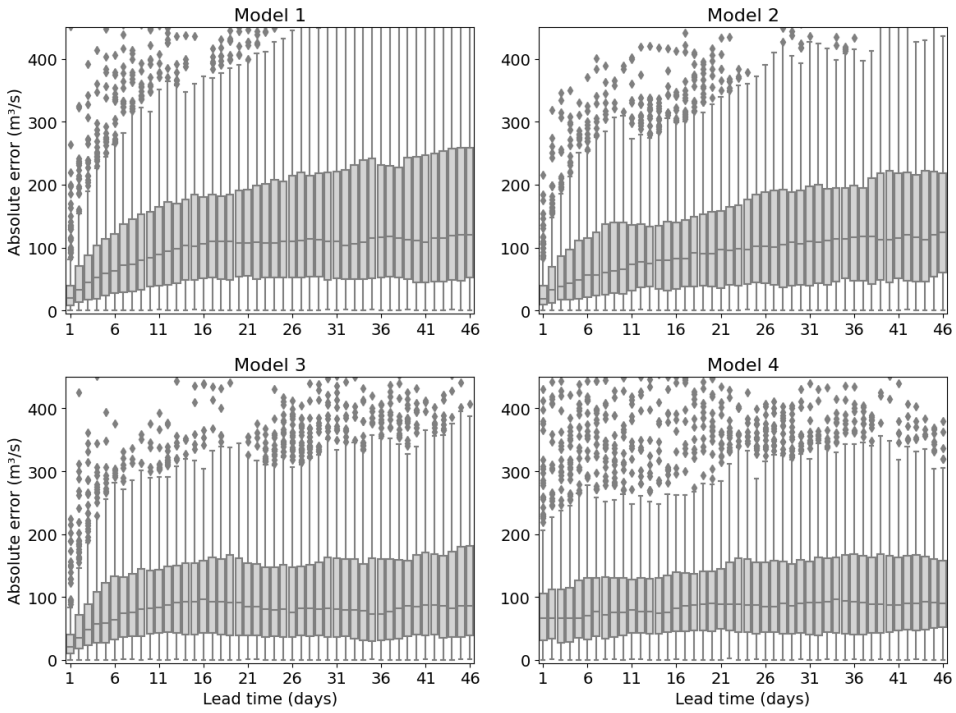


Figure 5-13 AE results of different model architectures.

To compare the performance of the model architectures, box plots are selected out from Figure 5-13 for several LTs and presented in Figure 5-14. For LT=1, Model 1, Model 2, and Model 3 demonstrate comparable performance, outperforming Model 4. At LT=5 and LT=10, the performance of all four models is similar. However, for LT=15 and LT=20, Model 2 and Model 4 perform slightly better than Model 1 and Model 3. Furthermore, for LTs beyond 20, Model 3 and Model 4 exhibit similar and superior performance compared to Model 1 and Model 2.

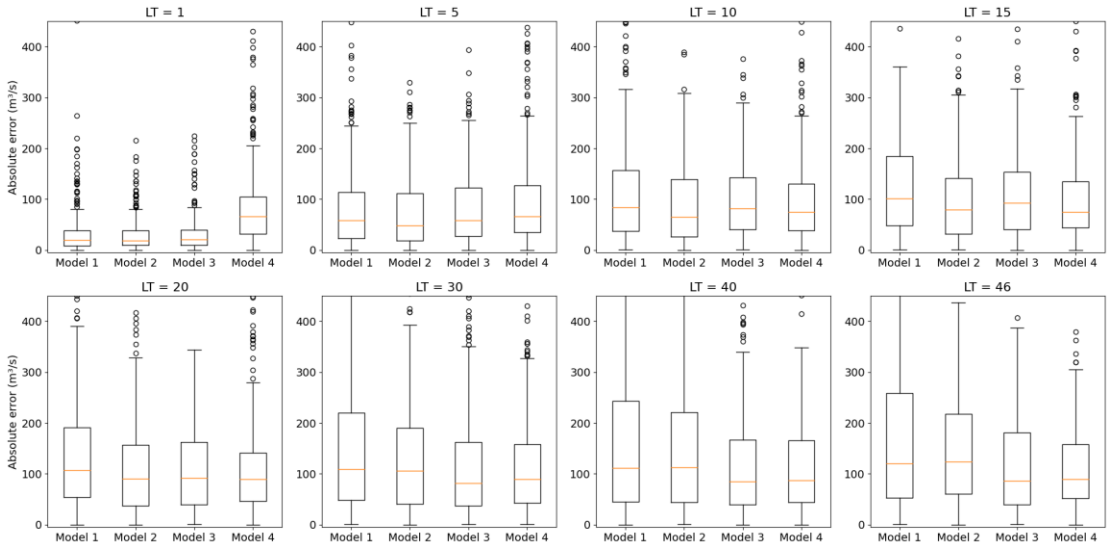


Figure 5-14 AE results of different model architectures for several LTs.

These comparative results are further depicted in the CDF plots shown in Figure 5-15. Model 1 and Model 2 perform better at shorter LTs (i.e., LT 1 to 20) compared to Model 4, but perform worse than Model 4 at longer LTs (i.e., LT 20 to 46). Model 3 demonstrates good performance across both short and long LTs.

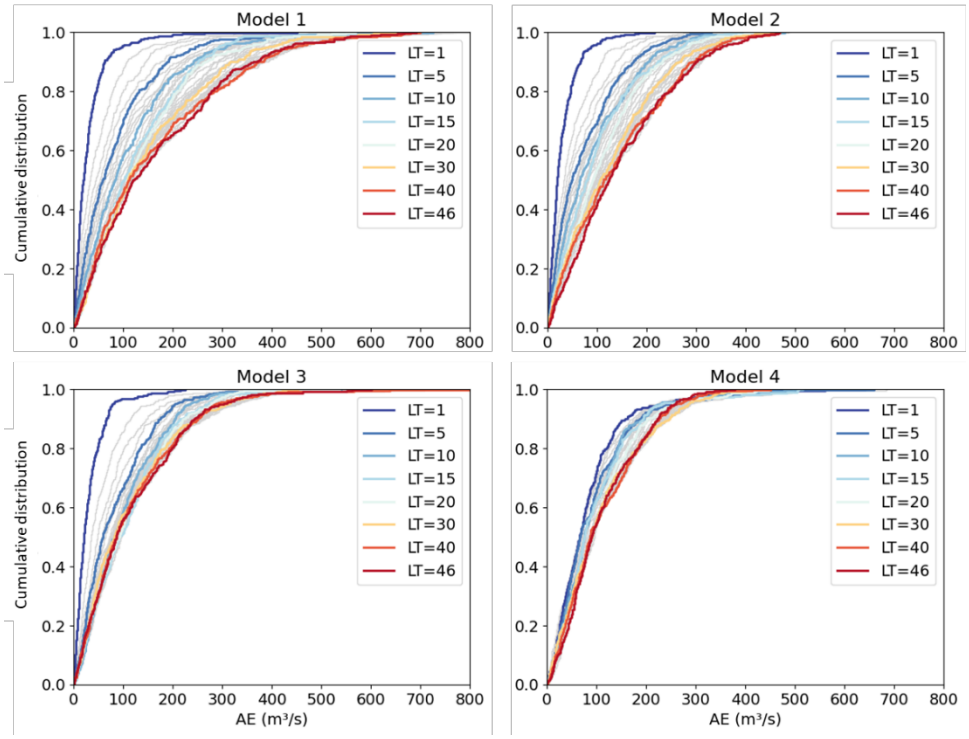


Figure 5-15 CDF of AE results of different model architectures. Several LTs are highlighted in colors. The grey lines are the results for other LTs.

5.2.3 Discussion of the results for SQ2

To address the second sub-research question (SQ2), which focuses on evaluating the effectiveness of different model architectures and comparing their performances, cross-validation and comparative analysis were conducted.

The cross-validation results reveal that Model 1 exhibits stable and robust performance for LTs ranging from 1 to 21 days, but its performance becomes less stable for LTs beyond 21 days. In contrast, Models 2, 3, and 4 demonstrate consistent and robust performance across the entire forecast horizon.

The discrepancy in Model 1's stability can be attributed to the fact that, in Model 1, the first LSTM does not share the learned past states with the second LSTM. Only the last output is concatenated with forecast forcing to generate the 46-step forecast. In hydrological terms, the forecast model (second LSTM) lacks knowledge of the hydrological system storages. While short-term forecasts can benefit from autoregression using past discharge due to relatively high autocorrelation, this approach becomes ineffective for longer LTs when the autocorrelation drops below 0.5 beyond 20 days (see Figure 5-7). Consequently, despite incorporating forecasted meteorology forcing, the model lacks information on system storages to leverage the new forcing information.

In Model 2, 3, and 4, on the other hand, the states learned from historical observations are transferred to future forecasting steps. This allows the models to start with correct states and utilize historical discharge information through autoregression for short LTs, as well as process forecasted meteorology forcing based on system storage information for longer LTs. Hence, the "transfer states" is a crucial step in the model architecture.

Comparing Model 2 and Model 3, their performances are similar for short LTs (1 to 20). However, for longer LTs (20 to 46), Model 3 outperforms Model 2. This result can be attributed to the "history representation" in Model 3, where the final hidden state is not only used as the initial state for LSTM-2 but is also concatenated with input X2 at each time step in LSTM-2. This architectural difference enables the forecast model (LSTM-2) to retain and utilize the initial state obtained from LSTM-1 for more steps, thereby enhancing performance for longer LTs.

In the case of Model 4, its performances are consistently stable across all LTs compared to the other three models. This can be attributed to the inclusion of observed true values (Y) or predicted values (Ŷ) from the previous R days as inputs for each forecast step in Model 4. By leveraging temporal dependencies and historical patterns of streamflow, Model 4 enhances its forecasting capability. However, it should be noted that Model 4 requires a longer training time, more than six times that of the other model architectures.

The cross-validation and comparative analysis are based on the model architectures together with the hyperparameters used in this study. However, it should be noted that the number of trainable parameters varies among the models (see Table 5-2). Model 1 has the fewest trainable parameters, while Model 3 has the highest number. It is possible that Model 3 outperforms the other models due to its larger parameter count. Therefore, to ensure a fair comparison of these model architectures, it is necessary to conduct extensive hyperparameter tuning for each model and compare the performances considering the number of trainable parameters.

Considering both model performance and training time, Model 3 is considered for operational use in this study. It exhibits stable and robust performance across the forecast horizon, with a slight advantage over Model 2 for longer LTs.

Table 5-2 Training time and number of trainable parameters for each model architecture. The training time is from Experiment 2B where the models are trained on all available training data. All the trainings are done on Google Colab with GPU A100.

Model architecture	Training time for 30 epochs [s]	Number of trainable parameters
Model 1	36	184238
Model 2	52	444590
Model 3	53	526638
Model 4	342	259969

5.3 Comparing the DL model with physically-based models (SQ3)

5.3.1 Experiment 3A: DL model vs Wflow-Rhine with ERA5

The forecast results of the DL model and the simulation results of Wflow-Rhine are plotted in Figure 5-16 for different years. The DL model forecast is initialized daily from 2017-06-28 to 2019-08-16, and all the forecast results are plotted together. It should be noted that the graph includes the continuous plotting of the forecast results of the last few days of each year, which extend into the next year.

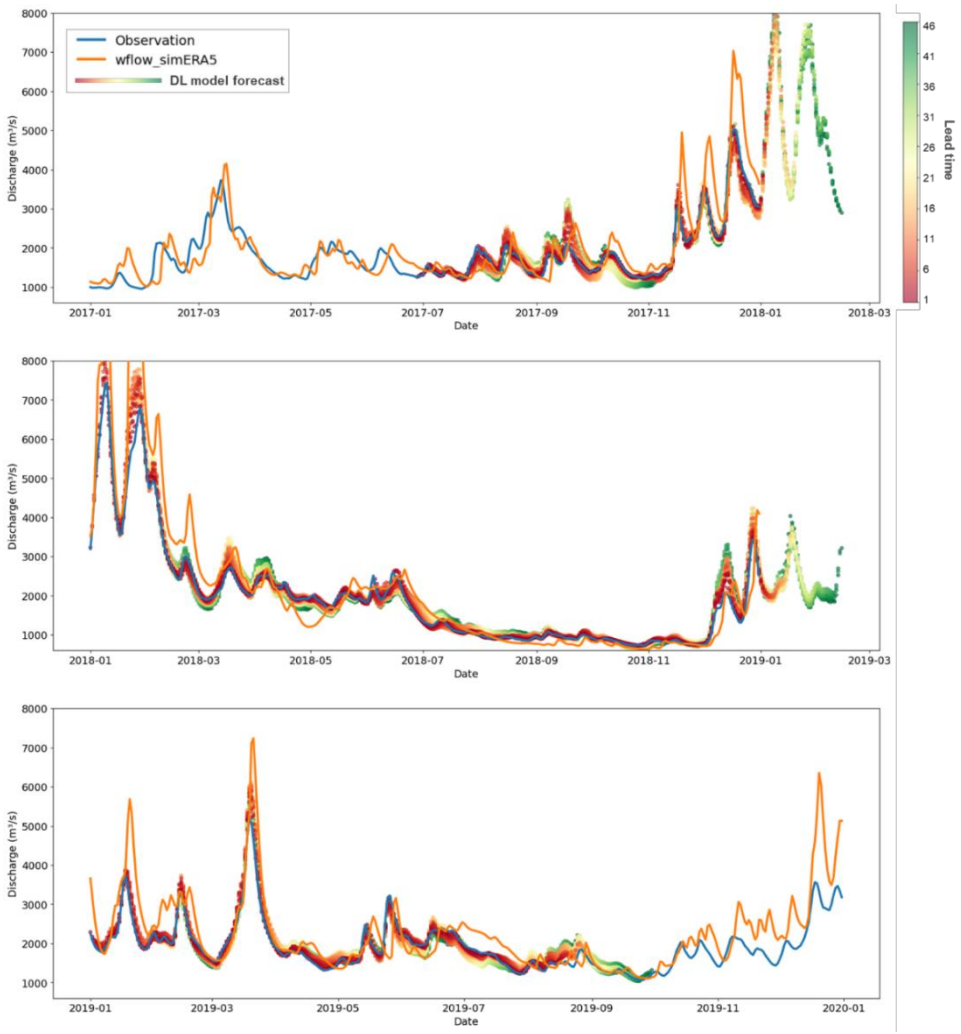


Figure 5-16 Forecast results of the DL model and simulation results of Wflow-Rhine for Experiment 3A. The blue line represents the true observation. The orange line represents the simulation results of Wflow-Rhine with ERA5. The red-yellow-green gradient line represents the forecast results of the DL model (Model 3) with ERA5, where the forecast is initialized daily. Note that the gradient colors represent different LTs, with red indicating shorter LTs and green indicating longer LTs.

From the time series plot, it can be seen that the DL model can effectively forecast both low flow in the dry season and peaks in the wet season, demonstrating a notable performance compared to the wflow simulation. Moreover, the forecasts of shorter LTs align more closely with the observed values, while deviations become more apparent as the LT increases.

To evaluate the results, both MAE and MAPE with standard deviation are calculated for each LT and averaged over the entire time series within the dry season, as shown in Figure 5-17. The MAE of the DL model forecast ranges from approximately 40 to 150 m^3/s for all LTs, which is smaller than the MAE of the wflow simulation. Furthermore, the uncertainty associated with the wflow simulation is greater than that of the DL model forecast. The standard deviation of the wflow simulation is approximately 200 m^3/s, whereas the DL model forecast exhibits a standard deviation ranging between 50 to 130 m^3/s across different LTs. Notably, the majority of the MAE+std values for the DL model forecast remain below the MAE values of the wflow simulation.

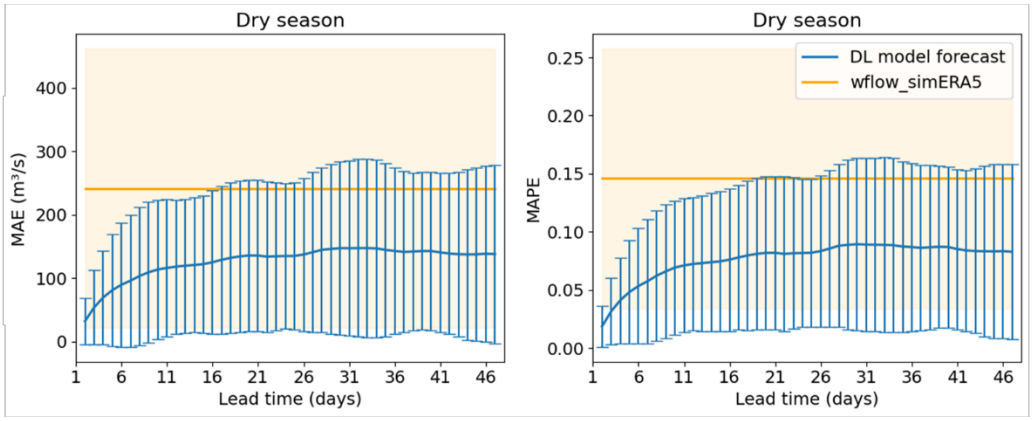


Figure 5-17 Comparison of MAE and MAPE with standard deviation (std) for forecast results of the DL model and simulation results of Wflow-Rhine in Experiment 3A. Blue lines represent the MAE/MAPE with std for DL model forecast. The orange line represents the MAE/MAPE and the orange fill represents the std of wflow simulation, assuming the same for all LTs.

5.3.2 Experiment 3B: DL model vs Wflow-Rhine with SEAS5

The forecasts in this experiment are initialized monthly from 2017-10-01 to 2020-05-01. In total, there are 32 forecast results, with 14 of them initialized in dry season from April to September. The example presented in Figure 5-18 demonstrates the forecast results of the DL model and Wflow-Rhine with SEAS5 initialized on 2018-08-01 and 2018-09-01.

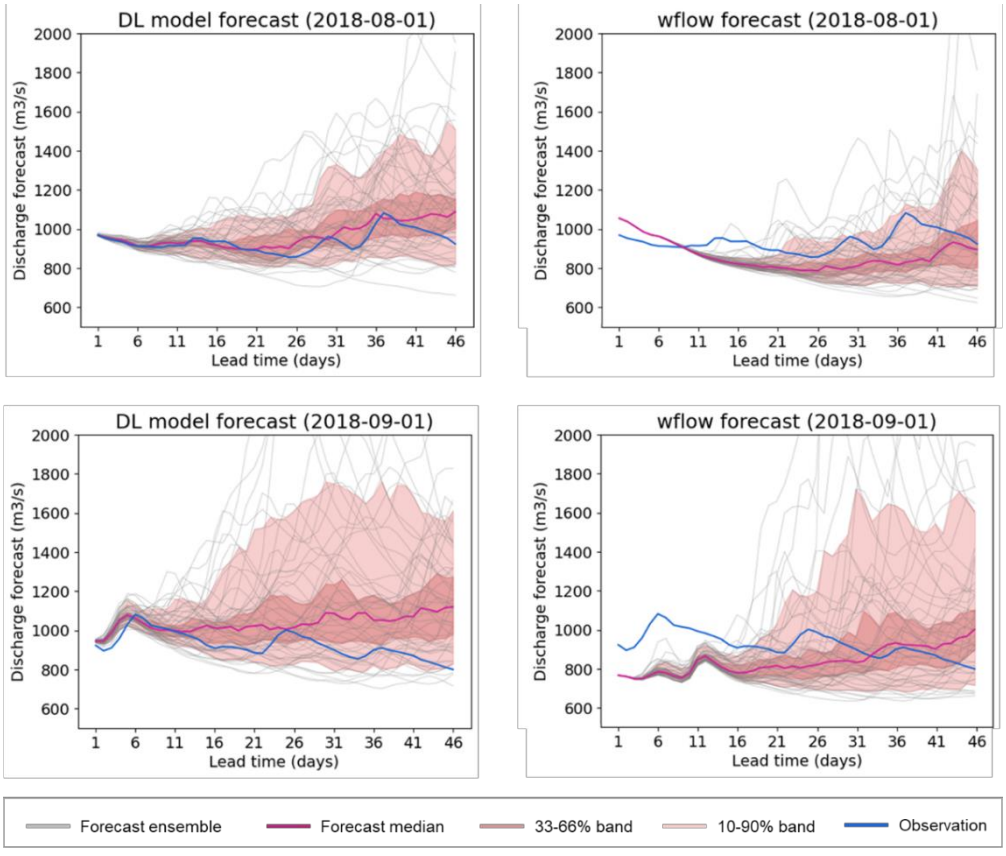


Figure 5-18 Forecast results of DL model and Wflow-Rhine with SEAS5 initialized on 2018-08-01 and 2018-09-01.

Regarding the forecasts initialized on 2018-08-01, the DL model exhibits a strong performance in predicting this streamflow drought event. The forecast begins with an accurate initial state, resulting in well-predicted discharge values for the first 6 days compared to the observed data. Furthermore, the median values of the DL model forecasts closely align with the observed values throughout the entire forecast horizon. Additionally, 29 out of 46 observations falls within the 33-66% band of the DL model forecasts, and all 46 observations falls within the 10-90% band. In contrast, the Wflow-Rhine model does not perform as well as the DL model. The initial state is not corrected using near real time observations, leading to a deviation from the true value in the forecast of the first few days. As the LT increases, the Wflow-Rhine model tends to underestimate the discharge, thereby overestimating the severity of the drought conditions. Specifically, for LT 1 to 21, the observed values fall outside the 10-90% band of the Wflow-Rhine forecasts.

For the forecasts initialized on 2018-09-01, the DL model again starts with an accurate state, with slight deviations occurring in LT 2-6. However, it successfully captures a small bump in LT=6. After 11 days, the DL model tends to overestimate the discharge, resulting in an underestimation of the drought conditions. Nevertheless, the low flow observations still fall within the 10-90% band of the DL model forecasts. In contrast, the Wflow-Rhine model deviates significantly from the initial state, and in LT=12, it generates a bump that does not align with the observed data. Neither the DL model nor the Wflow-Rhine model perform well for LT beyond 16 days, as the forecasted discharge exhibits a different trend compared to the observed data. One possible explanation for this discrepancy is the quality of the forecast meteorology data from SEAS5. To explore this explanation, observed meteorology data from ERA5 is used to carry out the forecast initialized on 2018-09-01. The result is shown in Figure 5-19. It can be seen that the DL model demonstrates a similar trend in the forecasted discharge as the observed discharge.

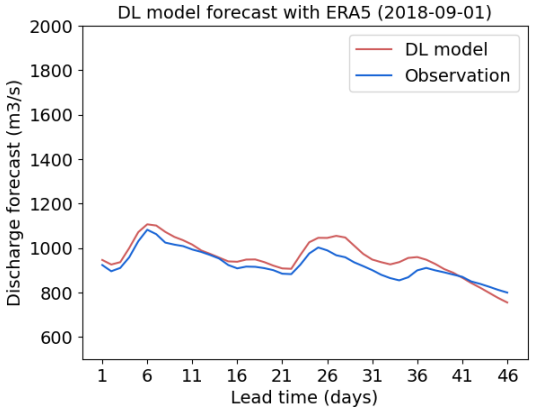


Figure 5-19 Forecast results of DL model with ERA5 initialized on 2018-09-01.

To evaluate the forecast results during the dry season, Continuous Ranked Probability Score (CRPS) and Continuous Ranked Probability Skill Score (CRPSS) are utilized. Figure 5-20 presents the CRPS of the DL model forecast results and the Wflow-Rhine forecast results in box plots. From the box plots, it can be seen that, in general, the DL model performs better than the Wflow-Rhine model in this comparative experiment, particularly for LT ranging from 1 to 11 days and the longer LTs. The DL model consistently exhibits lower CRPS values compared to the wflow model, indicating higher forecast accuracy.

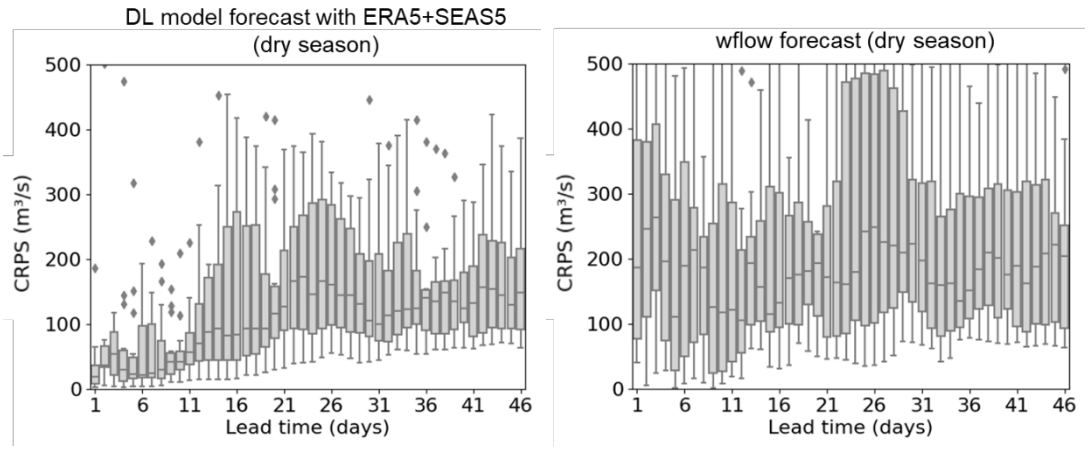


Figure 5-20 CRPS of the DL model forecast results and the wflow forecast results.

To facilitate a more straightforward comparison, the CRPSS of the DL model is computed relative to the Wflow-Rhine forecast results, as shown in Figure 5-21. The median values of all LTs are above zero, indicating that the DL model exhibits skill in forecasting during the dry season, with improved performance over the Wflow-Rhine model.

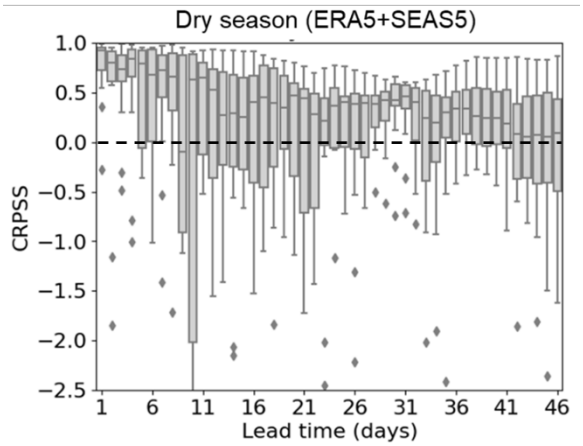


Figure 5-21 CRPSS of the DL model forecasts compared to wflow model forecasts. The dashed line indicates the CRPSS value of zero.

Given the focus of this study on forecasting streamflow drought, it is of great interest to assess how the models perform for low discharge events. Therefore, median CRPS and CRPSS are calculated for different discharge levels (bins), as depicted in Figure 5-22 and Figure 5-23. It should be noted that the bins are generated based on observed discharge, rather than forecasted discharge, and only discharges below 2700 m^3/s are plotted. Also, for each combination of bin and LT, only the cells with two or more samples are colored, while the rests are left blank. The forecasts in this experiment are initialized monthly from 2017-10-01 to 2020-05-01. The number of samples for each combination of bin and LT is shown in Figure 5-24.

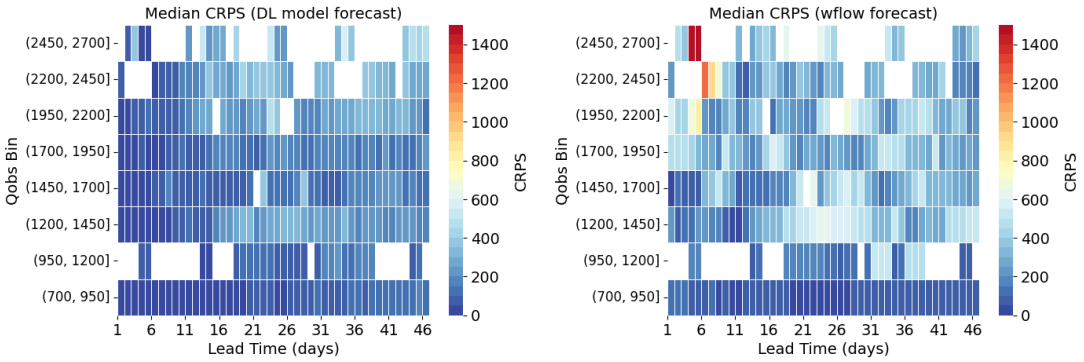


Figure 5-22 Median CRPS of the DL model forecast results and the wflow forecast results for different discharge levels. Colors are proportional to the CRPS value. Blank means there are less than 2 samples for that cell.

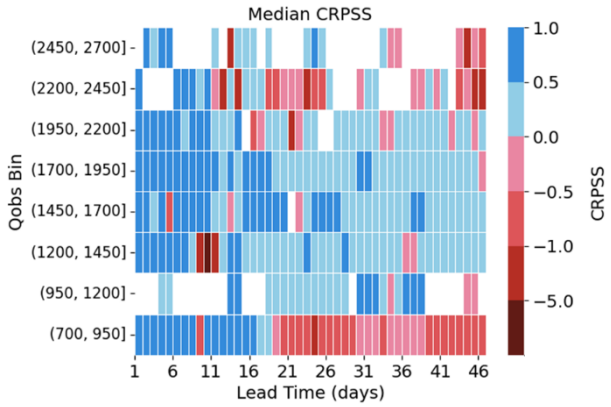


Figure 5-23 Median CRPSS of the DL model forecasts compared to wflow model forecasts for different discharge levels. Colors represent different value range. Blue represents positive value, i.e., the DL model forecast outperforms the wflow forecast, and red is the opposite. Blank means there are less than 2 samples for that cell.

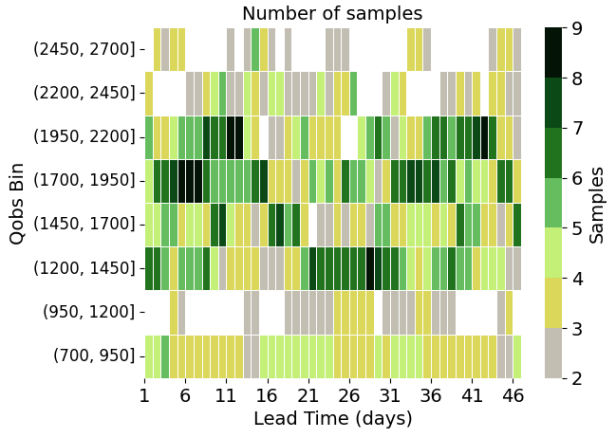


Figure 5-24 Number of samples for each combination of discharge observation (Qobs) bin and lead time of Experiment 3B: DL model vs Wflow-Rhine with SEAS5. Blank means there are less than 2 samples for that cell.

Figure 5-23 reveals that for discharge between 950 and 2200 m^3/s, most cells with LT from 1 to 11 days have median CRPSS values above 0.5, and most cells with LT beyond 11 days have values between 0 and 0.5. For discharge between 700 and 950 m^3/s, the median CRPSS values are above 0.5 for LTs from 1 to 16 days, but drop below 0 for LTs beyond 20 days. For discharge

between 2200 and 2700 m^3/s, the values are quite stable as above 0 for LT from 1 to 11 days, but varies for LT beyond 11 days.

Therefore, in this experiment, the DL model demonstrates forecast skill over Wflow-Rhine for short LTs (1-11 days) across all discharge levels. It also exhibits skill for discharge between 950 and 2200 m^3/s for all LTs. However, for discharge between 700 and 950 m^3/s with LTs beyond 20 days, Wflow-Rhine shows skill over the DL model. It should be noted that the sample size for this experiment is relatively small. For instance, for most cells with discharges lower than 1200 m^3/s, there are fewer than five samples available for analysis (Figure 5-24). In order to obtain a more robust and comprehensive analysis, longer forecast terms are needed.

5.3.3 Experiment 3C: DL model vs FEWS-Rhine

Experiment 3C aims to evaluate the performance of the DL model in forecasting drought events compared to FEWS-Rhine, specifically focusing on the year 2018. The DL model results from Experiment 3B are compared with the forecast results of FEWS-Rhine using the ENS extended forecast line (fews_hbv_ens_ext). The forecasted streamflow drought events at Lobith are defined based on the river discharge criteria outlined in Table 1-1. The forecast results of both models are presented in Appendix F.

Regarding the DL model forecasts, it successfully predicts the onset of the drought event in July, approximately around July 15th. This drought event continues until September. In the first two days of September, there is a slight increase in discharge, reaching above 1000 m^3/s. However, the forecasted discharge remains around 1000 m^3/s until the end of September, while the actual discharge falls back below 1000 m^3/s. Starting in October, the forecasted discharge aligns closely with the observed values, indicating a streamflow drought throughout the entire month. This drought event extends until early December, and the DL model accurately forecasts its end. Overall, the DL model correctly predicts the timing of both the start and end of the drought event. One discrepancy is observed in September when the forecasted discharge slightly increases, temporarily breaking the continuity of this five-month drought event.

In contrast, the FEWS-Rhine forecasts underestimate the discharge, resulting in an overestimation of the severity of drought conditions throughout the event. Additionally, FEWS-Rhine fails to accurately forecast the timing of both the start and end of the drought event. This inconsistency could be attributed to the fact that the forecast line fews_hbv_ens_ext of FEWS-Rhine does not undergo bias-correction using near real-time observations, leading to incorrect initial states. Moreover, the current operational system is not specifically designed for long-term and drought forecasting. It is initialized twice a week, but for this comparison, only the initialization closest to that of the DL models was chosen. This utilization does not fully leverage the twice-weekly forecasting capability of FEWS-Rhine.

The confusion matrix of DL model forecast results is shown in Table 5-3. The recall and precision values can be found in Table 5-4. For LT from 1 to 30 days, the DL model consistently achieves a recall value of one, indicating accurate prediction of drought events. However, for LTs from 31 to 40 days, only the recall value for the initialization on 2018-10-01 reaches one, while the recall values for other initializations are below 0.5 as the DL model wrongly predicts the drought as non-droughts (Type II errors) in these cases. On the other hand, the precision

values are generally high, with most of them reaching one. This indicates that the DL model has a low rate of false positives (FP).

Table 5-3 Confusion matrix of DL model forecast results for different initialization times and lead time (LT) bins.

Initialization time	LT 1 – 10 (10 days)				LT 11 – 20 (10 days)				LT 21 – 30 (10 days)				LT 31 – 40 (10 days)				LT 41 – 46 (6 days)			
	TP	FN	FP	TN	TP	FN	FP	TN	TP	FN	FP	TN	TP	FN	FP	TN	TP	FN	FP	TN
2018-07-01	0	0	3	7	6	0	3	1	10	0	0	0	1	9	0	0	0	6	0	0
2018-08-01	10	0	0	0	10	0	0	0	10	0	0	0	2	3	0	5	0	5	0	1
2018-09-01	3	1	0	6	4	6	0	0	0	9	0	1	0	10	0	0	0	6	0	0
2018-10-01	10	0	0	0	10	0	0	0	10	0	0	0	10	0	0	0	5	1	0	0
2018-11-01	10	0	0	0	10	0	0	0	10	0	0	0	1	2	0	7	6	0	0	0

Table 5-4 Recall and precision values derived from confusion matrix of DL model forecast results for different initialization times and lead time (LT) bins.

Initialization time	LT 1 – 10 (10 days)		LT 11 – 20 (10 days)		LT 21 – 30 (10 days)		LT 31 – 40 (10 days)		LT 41 – 46 (6 days)	
	Recall	Precision	Recall	Precision	Recall	Precision	Recall	Precision	Recall	Precision
2018-07-01	-	0	1	0.67	1	1	0.1	1	0	-
2018-08-01	1	1	1	1	1	1	0.4	1	0	-
2018-09-01	0.75	1	0.4	1	0	-	0	-	0	-
2018-10-01	1	1	1	1	1	1	1	1	0.83	1
2018-11-01	1	1	1	1	1	1	0.33	1	1	1

The confusion matrix of FEWS-Rhine forecast results is shown in Table 5-5. The recall and precision values can be found in Table 5-6. The recall values for all initializations and LTs equal one, indicating that FEWS-Rhine successfully forecasts all drought events. However, the precision values are not uniformly one. In comparison to the precision values of the DL model, FEWS-Rhine exhibits more Type I errors (false positives: FP), which suggests an overestimation of drought conditions. This finding is consistent with the time series plot in Appendix F, which indicates that FEWS-Rhine underestimates the discharge, leading to an overestimation of drought severity throughout the event.

Table 5-5 Confusion matrix of FEWS-Rhine forecast results for different initialization times and lead time (LT) bins. Note that the forecast results are missing after 2018-12-14, so for LT 41-46 of initialization 2018-11-01, results are only available for 3 days.

Initialization time	LT 1 – 10 (10 days)				LT 11 – 20 (10 days)				LT 21 – 30 (10 days)				LT 31 – 40 (10 days)				LT 41 – 46 (6 days)			
	TP	FN	FP	TN	TP	FN	FP	TN	TP	FN	FP	TN	TP	FN	FP	TN	TP	FN	FP	TN
2018-07-02	0	0	10	0	6	0	4	0	10	0	0	0	10	0	0	0	6	0	0	0
2018-08-02	10	0	0	0	10	0	0	0	10	0	0	0	5	0	5	0	5	0	1	0
2018-09-03	4	0	6	0	10	0	0	0	9	0	1	0	10	0	0	0	6	0	0	0
2018-10-01	10	0	0	0	10	0	0	0	10	0	0	0	10	0	0	0	6	0	0	0
2018-11-01	10	0	0	0	10	0	0	0	10	0	0	0	4	0	6	0	0	0	3	1

Table 5-6 Recall and precision values derived from confusion matrix of FEWS-Rhine forecast results for different initialization times and lead time (LT) bins.

Initialization time	LT 1 – 10 (10 days)		LT 11 – 20 (10 days)		LT 21 – 30 (10 days)		LT 31 – 40 (10 days)		LT 41 – 46 (6 days)	
	Recall	Precision	Recall	Precision	Recall	Precision	Recall	Precision	Recall	Precision
2018-07-01	-	0	1	0.6	1	1	1	1	1	1
2018-08-01	1	1	1	1	1	1	1	0.5	1	0.83
2018-09-01	1	0.4	1	1	1	0.9	1	1	1	1
2018-10-01	1	1	1	1	1	1	1	1	1	1
2018-11-01	1	1	1	1	1	1	1	0.4	-	0

5.3.4 Discussion of the results for SQ3

To address the third sub-research question (SQ3), three comparisons between the DL model and physically-based models, i.e., Wflow-Rhine and FEWS-Rhine, were designed and executed based on the availability of forecast forcings and existing results from physically-based models.

In Experiment 3A, the DL model demonstrates a notable performance compared to the wflow simulation. However, it is important to note that the DL model incorporates historical discharges as additional input to forecast 46 steps, whereas the wflow simulation only utilizes meteorology forcing to forecast streamflow one step ahead. This inclusion of historical discharge data improves the DL model's forecast skill. Although this comparison may not be entirely fair due to the different inputs, it highlights the DL model's ability to forecast longer lead times effectively. As the lead time increases, the influence of historical discharges diminishes. Even for lead times up to 46 days, the DL model exhibits notable forecast skill compared to the benchmark provided by the wflow simulation.

In Experiment 3B, it is important to acknowledge that the DL model may not have been compared to the best operational setting of the wflow model. The wflow results used in this comparison were not corrected using near real-time observations, which means that the wflow model does not start from the correct initial state. This discrepancy could have influenced the performance across the entire forecast horizon. If the wflow results were corrected for bias, it would provide a stronger basis for comparing the DL model's performance. Despite this limitation, the DL model still demonstrates skill for short LTs (1-11 days) across all discharge levels, and skill for discharge between 950 and 2200 m^3/s for all LTs.

Another crucial point to discuss is the choice of data sources in this experiment. In the training mode, ERA5 is used for both X1 (input for LSTM-1) and X2 (input for LSTM-2), while in the forecast mode, ERA5 is used for X1 along with SEAS5. SEAS5 data is not used for training due to the limited number of samples available. While this setup leaves the forecast mode vulnerable to biases in SEAS5, it is hypothesized that ERA5 and SEAS5 share some statistical characteristics since they are both generated using the atmospheric model ECMWF Integrated Forecast System (IFS), which might help mitigate the issue to some extent. Additionally, using SEAS5 in the forecast mode allows for testing the robustness of the DL model when utilizing a different dataset from the training phase. To further test this hypothesis, EOBS is used instead of ERA5 to conduct the same experiment for Experiment 3B. It should be noted that from past experiences, the data quality of EOBS has been found to be higher than ERA5. However, the results, presented in Appendix F, indicate a deterioration in the DL model's performance compared to the experiment with ERA5. Therefore, it is crucial to use the same or similar data sources for both the training and forecast phases.

Furthermore, there are methods available to further mitigate the influence of bias in SEAS5 during the forecast mode. For example, after training the DL model with ERA5, LSTM-2 can be fine-tuned using a two-year SEAS5 dataset. This approach allows for a better utilization of the limited forecast data.

In Experiment 3C, the DL model accurately predicts drought events for LTs up to 30 days, and has generally high precision values, i.e., low rate of false positives. FEWS-Rhine successfully forecasts drought events across all LTs, but exhibits more Type I errors (false positives) due to an underestimation of discharge. Despite using different input datasets for the DL model and FEWS-Rhine, the DL model exhibits skill along the forecast horizon. It correctly predicts the timing and trend of past drought events, indicating its potential in capturing streamflow patterns. However, for a fair comparison between the two models, it would be beneficial to have an ENS extended dataset available during dry seasons as input. This would provide a more comprehensive assessment of their performances.

Furthermore, in both Experiment 3B and Experiment 3C, the DL model is trained solely on data from 1979 to 2016, and this trained model is then used for all forecasts. However, for operational use, it is preferable to train the model using as much data as possible prior to initialization. Different training strategies could impact the forecast skill of the DL model, and it is hypothesized that utilizing more training data can improve its performance in the forecast mode. Although this study does not explore this aspect, it presents an opportunity for future improvements.

6 Limitations and recommendations

This study aims to explore the potential of the LSTM deep learning approach for operational streamflow drought forecasting for the Rhine River at Lobith with a lead time of up to 46 days. Based on the results and discussions, several limitations of this study are identified, along with recommendations for improvement and future research.

Regarding the input data, the study utilizes three meteorological parameters – total precipitation (*tp*), 2 meter temperature (*t2m*), and potential evaporation (*pev*) – to describe the meteorological conditions over time. However, the selection of these parameters is based on hydrological knowledge and physically-based models without exploring their detailed influences on the DL models. Therefore, it is suggested to conduct a sensitivity analysis to assess how these three parameters impact the model results. Additionally, as potential evaporation exhibits a high correlation with air temperature, and high correlation among inputs could potentially degrade the performance of DL models. To address this, it is recommended to explore the use of incoming shortwave radiation as an alternative to potential evaporation.

Moreover, the study uses a 270-day length of historical discharge (*Q_his*) as input, aligning it with the sequence length of other meteorological parameters. However, the contribution of *Q_his* as inputs to the forecast results and the rationale for choosing a 270-day length instead of other lengths have not been explored. Further investigation is necessary to understand the processes captured in *Q_his* data that may not be in meteorological parameters, and the effect of varying *Q_his* lengths on model performance as it can help identify and prevent input redundancy.

Furthermore, the water level at Lake Constance (*wl*) exhibits different performance when different target variables and model architectures are used. It is recommended to explore the impact and contribution of *wl* in future studies, as it integrates the operation of reservoirs into DL models and could help to understand how human interactions impact hydrological processes during droughts.

In addition to the variables utilized in this study, other valuable information can be incorporated into the DL models as input. For instance, integrating discharge data from upstream stations with a specified look-back window may enhance the model performance, considering the high correlation observed between the discharge at Lobith and the discharges from upstream stations (International Commission for the Protection of the Rhine, 2018). Furthermore, including static subbasin attributes such as area, mean elevation, land cover, drainage density and sand fraction as inputs might also improve the model performance (Kratzert et al., 2019).

Regarding the choice of loss function, the study uses the mean squared error (MSE), which is commonly employed in the DL models. However, it is recommended to explore alternative objective functions that are suitable for time series forecasting of non-stationary signals and multiple future steps prediction. One promising alternative is the DILATE function developed by Le Guen & Thome (2019). The DILATE function aims to accurately predict sudden changes and incorporates terms that facilitate precise shape and temporal change detection. This loss function would be beneficial for predicting the timing, specifically the start and end, of drought events in this study.

In terms of the model architectures, for all the experiments conducted for Model 4, the target variable of *Q* is used, as the architecture of Model 4 cannot be easily modified to model *delta_Q*. However, if it were possible to train Model 4 on *delta_Q*, it is anticipated that the model would yield improved performance, particularly for short lead times.

In terms of model training, in this study, the training datasets span from 1979 to 2016, while the testing datasets cover the period from 2017 to 2020. The selection of these periods is based on data availability and the need to generate drought events in 2018 for model performance evaluation. However, it should be noted that the training data has limited inclusion of severe drought events. DL models have the potential to perform better at predicting extreme events if they are exposed to such events during the training phase. Therefore, it is recommended to include a broader range of data, particularly encompassing extreme drought events such as the drought in 1976, to enhance the model performance in predicting extreme events during the inference phase.

To better interpret the DL model, examining the cell states of LSTM can help determine whether the model effectively captures hydrological knowledge. Specifically, investigating whether the model learns the patterns that indicate snow-dominated discharge during the dry season and rain-dominated discharge during the wet season would provide valuable insights.

To generalize the model's applicability to other locations, it is recommended to conduct tests in catchments of varying sizes. The study conducted by Nevo et al. (2022) on operational flood forecasting has shown that LSTM-based models perform better in large river systems, showing a positive correlation between model performance and watershed area. Following a similar approach, exploring different catchment characteristics can provide valuable insights into the model's effectiveness across diverse hydrological settings. Additionally, it is worth exploring the potential for forecasting ungauged basins through transfer learning, utilizing trained models from large samples of gauged basins with only meteorology parameters as inputs.

It is acknowledged that anthropogenic interactions can significantly influence hydrological processes during drought events. Although the study attempted to consider the operation of reservoirs by incorporating Lake Constance water level information, this addition did not improve the model performance significantly. Nevertheless, there is a need for further research to explore and integrate anthropogenic interactions into the DL model. By doing so, the impact of human activities such as construction of reservoirs, abstractions from surface water or groundwater and water diversion on the hydrological processes during drought can be better modeled and understood.

7 Conclusion

This research focuses on exploring the potential of the LSTM deep learning approach for operational streamflow drought forecasting for the Rhine River at Lobith, with a lead time (LT) of up to 46 days. The research objectives are to investigate optimal spatial resolution, input and target variables, and loss functions, develop suitable model architectures for operational frameworks, and compare the performance of LSTM-based models with physically-based models in forecasting streamflow drought. Three sub-research questions (SQ) have been formulated to align with the specific research objectives. The answer for each SQ is summarized below.

SQ1: What combinations of spatial resolution, input and target variables, and loss functions can be used to optimize the performance of LSTM-based models for drought forecasting?

The study found that using subbasin mean approach for meteorological forcing proves beneficial, enabling the model to capture information from various subbasins at different times. Incorporating historical discharge at Lobith improves the overall forecast skill. However, the addition of snow and lake level information do not provide significant added value. Additionally, training the model on time-differenced data as the target variable greatly improves forecast performance, particularly for short lead times. As for loss functions, Model 1, Model 2 and Model 4 exhibit different performance patterns with varying loss weights, while Model 3 shows relatively low sensitivity and no clear pattern to changes in loss weights.

SQ2: What LSTM-based model architectures are suitable for handling the various data sources available in an operational framework and how do they compare in performance?

To handle the various data sources available in an operational framework, two LSTMs in parallel or cascade are used, with one LSTM processing historical observation data and another LSTM processing forecast data. The cross-validation experiments demonstrate that Model 1 exhibits stable and robust performance for lead times up to around 20 days but becomes less stable beyond that threshold. In contrast, Models 2, 3, and 4 consistently demonstrate robust performance across the entire forecast horizon due to the inclusion of "transfer states" in their architectures, which allows the transfer of learned states from historical observations to future forecasting steps. Model 3, which employes "history representation", further enhances performance for longer lead times by utilizing the initial state obtained from the first LSTM for more steps.

SQ3: How does the performance of the LSTM-based model compared to physically-based models for drought forecasting?

Comparisons between the LSTM-based model (Model 3) and physically-based models in forecasting streamflow drought reveal that: 1) When using observed meteorology forcing from ERA5, the DL model demonstrates a notable performance compared to Wflow-Rhine simulation using the same forcing data. 2) When utilizing SEAS5 for forecasting, the DL model demonstrates skill over Wflow-Rhine in predicting discharge levels during the dry season up to 10 days ahead, as well as for discharges between 950 and 2200 m^3/s across the entire forecast horizon. However, for discharges between 700 and 950 m^3/s with longer LTs beyond 20 days, Wflow-Rhine shows skill over the DL model. 3) While FEWS-Rhine successfully forecasts drought events in 2018 throughout the forecast horizon, it tends to produce more

Type I errors (false positives) possibly due to the absence of state updates, leading to an underestimation of discharge. The DL model, forecasting with SEAS5, accurately predicts drought events in 2018 for LTs up to 30 days and generally has higher precision values. Despite using different forcing datasets, the DL model can predict the timing and trend of past drought events, indicating its potential in capturing streamflow patterns.

Overall, this study contributes to operational water management in the Netherlands by employing the LSTM deep learning approach in an operational framework for drought forecasting. These models leverage both historical observation data and forecasted meteorology forcing data, resulting in a skillful performance for streamflow drought forecasts. Future research could explore additional improvements to model performance, investigate the applicability of LSTM-based models in other river basins, and further validate the results in real operational settings. The findings of this study provide valuable insights for water managers and contribute to advancing the field of streamflow drought forecasting using DL models.

Bibliography

Aghelpour, P., Bahrami-Pichaghchi, H., & Varshavian, V. (2021). Hydrological drought forecasting using multi-scalar streamflow drought index, stochastic models and machine learning approaches, in northern Iran. *Stochastic Environmental Research and Risk Assessment*, *35*(8), 1615–1635. https://doi.org/10.1007/s00477-020-01949-z

Amanambu, A. C., Mossa, J., & Chen, Y.-H. (2022). Hydrological Drought Forecasting Using a Deep Transformer Model. *Water*, *14*(22). https://doi.org/10.3390/w14223611

Ben Taieb, S., Bontempi, G., Atiya, A. F., & Sorjamaa, A. (2012). A review and comparison of strategies for multi-step ahead time series forecasting based on the NN5 forecasting competition. *Expert Systems with Applications*, *39*(8), 7067–7083. https://doi.org/https://doi.org/10.1016/j.eswa.2012.01.039

Borji, M., Malekian, A., Salajegheh, A., & Ghadimi, M. (2016). Multi-time-scale analysis of hydrological drought forecasting using support vector regression (SVR) and artificial neural networks (ANN). *Arabian Journal of Geosciences*, *9*(19), 725. https://doi.org/10.1007/s12517-016-2750-x

Cammalleri, C., Naumann, G., Mentaschi, L., Bisselink, B., Gelati, E., De Roo, A., & Feyen, L. (2020). Diverging hydrological drought traits over Europe with global warming. *Hydrology and Earth System Sciences*, *24*(12), 5919–5935. https://doi.org/10.5194/hess-24-5919-2020

Cerqueira, V., Torgo, L., & Mozetič, I. (2020). Evaluating time series forecasting models: an empirical study on performance estimation methods. *Machine Learning*, *109*(11), 1997–2028. https://doi.org/10.1007/s10994-020-05910-7

de Bruin, H. (1987). From Penman to Makkink. *Evaporation and Weather*, *39*, 5–31.

de Vries, H., Kort, B., Teunis, B., Winters, M., & Beijk, V. (2021). *Landelijk draaiboek waterverdeling en droogte*. www.helpdeskwater.nl.

Deltares. (2019). *Integrated Overview of the effects of socio-economic scenarios on the discharge of the Rhine*.

Demirel, M. C., Booij, M. J., & Hoekstra, A. Y. (2013). Identification of appropriate lags and temporal resolutions for low flow indicators in the River Rhine to forecast low flows with different lead times. *Hydrological Processes*, *27*(19), 2742–2758. https://doi.org/https://doi.org/10.1002/hyp.9402

Dikshit, A., Pradhan, B., & Alamri, A. M. (2021). Long lead time drought forecasting using lagged climate variables and a stacked long short-term memory model. *Science of The Total Environment*, *755*, 142638. https://doi.org/https://doi.org/10.1016/j.scitotenv.2020.142638

Dikshit, A., Pradhan, B., & Santosh, M. (2022). Artificial neural networks in drought prediction in the 21st century–A scientometric analysis. *Applied Soft Computing*, *114*, 108080. https://doi.org/https://doi.org/10.1016/j.asoc.2021.108080

EEA. (2010). *Mapping the impacts of natural hazards and technological accidents in Europe An overview of the last decade*. https://doi.org/10.2800/62638

Fang, K., Shen, C., Kifer, D., & Yang, X. (2017). Prolongation of SMAP to Spatiotemporally Seamless Coverage of Continental U.S. Using a Deep Learning Neural Network. *Geophysical Research Letters*, *44*(21), 11, 11–30, 39. https://doi.org/https://doi.org/10.1002/2017GL075619

Fundel, F., Jörg-Hess, S., & Zappa, M. (2013). Monthly hydrometeorological ensemble prediction of streamflow droughts and corresponding drought indices. *Hydrology and Earth System Sciences*, *17*(1), 395–407. https://doi.org/10.5194/hess-17-395-2013

Gauch, M., Kratzert, F., Klotz, D., Nearing, G., Lin, J., & Hochreiter, S. (2021). Rainfall–runoff prediction at multiple timescales with a single Long Short-Term Memory network. *Hydrology and Earth System Sciences*, *25*(4), 2045–2062. https://doi.org/10.5194/hess-25-2045-2021

Hrachowitz, M., & Clark, M. P. (2017). HESS Opinions: The complementary merits of competing modelling philosophies in hydrology. *Hydrology and Earth System Sciences*, *21*(8), 3953–3973. https://doi.org/10.5194/hess-21-3953-2017

Hunt, K. M. R., Matthews, G. R., Pappenberger, F., & Prudhomme, C. (2022). Using a long short-term memory (LSTM) neural network to boost river streamflow forecasts over the western United States. *Hydrology and Earth System Sciences*, *26*(21), 5449–5472. https://doi.org/10.5194/hess-26-5449-2022

Ibrahim, K. S. M. H., Huang, Y. F., Ahmed, A. N., Koo, C. H., & El-Shafie, A. (2022). A review of the hybrid artificial intelligence and optimization modelling of hydrological streamflow forecasting. *Alexandria Engineering Journal*, *61*(1), 279–303. https://doi.org/https://doi.org/10.1016/j.aej.2021.04.100

Imhoff, R. O., van Verseveld, W. J., van Osnabrugge, B., & Weerts, A. H. (2020). Scaling Point-Scale (Pedo)transfer Functions to Seamless Large-Domain Parameter Estimates for High-Resolution Distributed Hydrologic Modeling: An Example for the Rhine River. *Water Resources Research*, *56*(4), e2019WR026807. https://doi.org/https://doi.org/10.1029/2019WR026807

International Commission for the Protection of the Rhine. (2018). *Inventory of the low water conditions on the Rhine*. www.iksr.org

Ionita, M., Tallaksen, L. M., Kingston, D. G., Stagge, J. H., Laaha, G., Van Lanen, H. A. J., Scholz, P., Chelcea, S. M., & Haslinger, K. (2017). The European 2015 drought from a climatological perspective. In *Hydrology and Earth System Sciences* (Vol. 21, Issue 3, pp. 1397–1419). Copernicus GmbH. https://doi.org/10.5194/hess-21-1397-2017

Khakbaz, B., Imam, B., Hsu, K., & Sorooshian, S. (2012). From lumped to distributed via semi-distributed: Calibration strategies for semi-distributed hydrologic models. *Journal of Hydrology*, *418–419*, 61–77. https://doi.org/https://doi.org/10.1016/j.jhydrol.2009.02.021

Kramer, N., Mens, M., Beersma, J., & Kielen, N. (2019). *Hoe extreem was de droogte van 2018?*

Kratzert, F., Klotz, D., Brenner, C., Schulz, K., & Herrnegger, M. (2018). Rainfall–runoff modelling using Long Short-Term Memory (LSTM) networks. *Hydrology and Earth System Sciences*, *22*(11), 6005–6022. https://doi.org/10.5194/hess-22-6005-2018

Kratzert, F., Klotz, D., Shalev, G., Klambauer, G., Hochreiter, S., & Nearing, G. (2019). Towards learning universal, regional, and local hydrological behaviors via machine learning applied to large-sample datasets. *Hydrology and Earth System Sciences*, *23*(12), 5089–5110. https://doi.org/10.5194/hess-23-5089-2019

Lam, R., Sanchez-Gonzalez, A., Willson, M., Wirnsberger, P., Fortunato, M., Pritzel, A., Ravuri, S., Ewalds, T., Alet, F., Eaton-Rosen, Z., Hu, W., Merose, A., Hoyer, S., Holland, G., Stott, J., Vinyals, O., Mohamed, S., & Battaglia, P. (2022). *GraphCast: Learning skillful medium-range global weather forecasting*. https://doi.org/10.48550/arXiv.2212.12794

Le Guen, V., & Thome, N. (2019). *Shape and Time Distortion Loss for Training Deep Time Series Forecasting Models*.

Middelkoop, H., & van Haselen. (1999). *Twice a river Rhine and Meuse in The Netherlands*.

Mosavi, A., Ozturk, P., & Chau, K. (2019). *Flood Prediction Using Machine Learning Models: Literature Review*.

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., … Matias, Y. (2022). Flood forecasting with machine learning models in an operational framework. *Hydrology and Earth System Sciences*, *26*(15), 4013–4032. https://doi.org/10.5194/hess-26-4013-2022

Prudhomme, C., Giuntoli, I., Robinson, E. L., Clark, D. B., Arnell, N. W., Dankers, R., Fekete, B. M., Franssen, W., Gerten, D., Gosling, S. N., Hagemann, S., Hannah, D. M., Kim, H., Masaki, Y., Satoh, Y., Stacke, T., Wada, Y., & Wisser, D. (2014). Hydrological droughts in the 21st century, hotspots and uncertainties from a global multimodel ensemble experiment. *Proceedings of the National Academy of Sciences of the United States of America*, *111*(9), 3262–3267. https://doi.org/10.1073/pnas.1222473110

Rijkswaterstaat. (2019). *Water management in the Netherlands*. http://www.helpdeskwater.nl/watermanagement

Sahoo, B. B., Jha, R., Singh, A., & Kumar, D. (2019). Long short-term memory (LSTM) recurrent neural network for low-flow hydrological time series forecasting. *Acta Geophysica*, *67*(5), 1471–1481. https://doi.org/10.1007/s11600-019-00330-1

Shah, S. M. S., O'Connell, P. E., & Hosking, J. R. M. (1996). Modelling the effects of spatial variability in rainfall on catchment response. 2. Experiments with distributed and lumped models. *Journal of Hydrology*, *175*(1), 89–111. https://doi.org/https://doi.org/10.1016/S0022-1694(96)80007-2

Shamshirband, S., Hashemi, S., Salimi, H., Samadianfard, S., Asadi, E., Shadkani, S., Kargar, K., Mosavi, A., Nabipour, N., & Chau, K.-W. (2020). Predicting Standardized Streamflow index for hydrological drought using machine learning models. *Engineering Applications of Computational Fluid Mechanics*, *14*(1), 339–350. https://doi.org/10.1080/19942060.2020.1715844

Shen, C. (2018). A Transdisciplinary Review of Deep Learning Research and Its Relevance for Water Resources Scientists. In *Water Resources Research* (Vol. 54, Issue 11, pp. 8558–8593). Blackwell Publishing Ltd. https://doi.org/10.1029/2018WR022643

Shen, C., Chen, X., & Laloy, E. (2021). Editorial: Broadening the Use of Machine Learning in Hydrology. *Frontiers in Water*, *3*. https://doi.org/10.3389/frwa.2021.681023

Sutanto, S. J., Van Lanen, H. A. J., Wetterhall, F., & Llort, X. (2020). Potential of pan-european seasonal hydrometeorological drought forecasts obtained from a multihazard early warning system. *Bulletin of the American Meteorological Society*, *101*(4), E368–E393. https://doi.org/10.1175/BAMS-D-18-0196.1

Tallaksen, L. M., & Van Lanen, H. (2004). *Hydrological Drought: Processes and Estimation Methods for Streamflow and Groundwater*.

Troutman, B. M. (1983). Runoff prediction errors and bias in parameter estimation induced by spatial variability of precipitation. *Water Resources Research*, *19*(3), 791–810. https://doi.org/https://doi.org/10.1029/WR019i003p00791

van der Wiel, K., Wanders, N., Selten, F. M., & Bierkens, M. F. P. (2019). Added Value of Large Ensemble Simulations for Assessing Extreme River Discharge in a 2 °C Warmer World. *Geophysical Research Letters*, *46*(4), 2093–2102. https://doi.org/10.1029/2019GL081967

Van Hateren, T. C., Sutanto, S. J., & Van Lanen, H. A. J. (2019). Evaluating skill and robustness of seasonal meteorological and hydrological drought forecasts at the catchment scale – Case Catalonia (Spain). *Environment International*, *133*. https://doi.org/10.1016/j.envint.2019.105206

Van Lanen, H. A. J., Wanders, N., Tallaksen, L. M., & Van Loon, A. F. (2013). Hydrological drought across the world: impact of climate and physical catchment structure. *Hydrology and Earth System Sciences*, *17*(5), 1715–1732. https://doi.org/10.5194/hess-17-1715-2013

Van Loon, A. F. (2015). Hydrological drought explained. *WIREs Water*, *2*(4), 359–392. https://doi.org/https://doi.org/10.1002/wat2.1085

Van Loon, A. F., Gleeson, T., Clark, J., Van Dijk, A. I. J. M., Stahl, K., Hannaford, J., Di Baldassarre, G., Teuling, A. J., Tallaksen, L. M., Uijlenhoet, R., Hannah, D. M., Sheffield, J., Svoboda, M., Verbeiren, B., Wagener, T., Rangecroft, S., Wanders, N., & Van Lanen, H. A. J. (2016). Drought in the Anthropocene. In *Nature Geoscience* (Vol. 9, Issue 2, pp. 89–91). Nature Publishing Group. https://doi.org/10.1038/ngeo2646

Van Loon, A. F., & Van Lanen, H. A. J. (2012). A process-based typology of hydrological drought. *Hydrology and Earth System Sciences*, *16*(7), 1915–1946. https://doi.org/10.5194/hess-16-1915-2012

van Verseveld, W., Visser, M., Boisgontier, H., Bootsma, H., Bouaziz, L., Buitink, J., Eilander, D., & Hegnauer, M. (2023). *Wflow.jl*. https://doi.org/10.5281/ZENODO.7687551

Vertessy, R. A., & Elsenbeer, H. (1999). Distributed modeling of storm flow generation in an Amazonian rain forest catchment: Effects of model parameterization. *Water Resources Research*, *35*(7), 2173–2187. https://doi.org/https://doi.org/10.1029/1999WR900051

Wanders, N., & Van Lanen, H. A. J. (2015). Future discharge drought across climate regions around the world modelled with a synthetic hydrological modelling approach forced by three general circulation models. *Natural Hazards and Earth System Sciences*, *15*(3), 487–504. https://doi.org/10.5194/nhess-15-487-2015

Weerts, A. (2009). *2009.06.11 Improving operational flood forecasting through data assimilation*. https://publications.deltares.nl/1200379_005.pdf

Wilhite, D., & Glantz, M. (1985). Understanding: the Drought Phenomenon: The Role of Definitions. *Water International - WATER INT*, *10*, 111–120. https://doi.org/10.1080/02508068508686328

WMO. (2020). *WMO statement on the state of the global climate in 2019*.

A Time terms

In operational forecasting, various time terms are used to specify different aspects of the forecast. To facilitate understanding, it is essential to introduce the relevant terms. Figure A-1 provides an illustration of these terms and notations in the context of this study which focuses on forecasting up to 46 days ahead.

- Validity time (d), which refers to the specific date and time associated with a particular streamflow state. It represents the point in time for which the forecast is intended.
- Forecast initialization time (t), which indicates the validity time of a forecast's initial inputs. It represents the starting point from which the forecast is generated.
- Forecast horizon (T), which represents the total number of steps in a forecast. It provides an indication of the length or duration of the forecast.
- Forecast lead time (LT), which represents the elapsed time in the forecast. It indicates the time difference between the forecast initialization time and the validity time of the forecast.

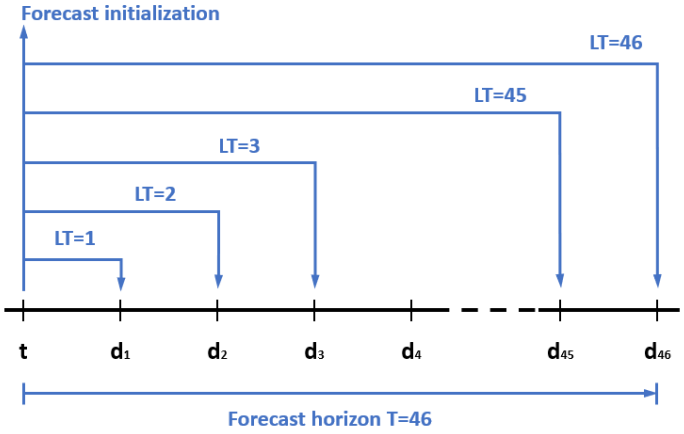


Figure A-1 Illustration of relevant time terms and notations in the context of this study forecasting up to 46 days ahead.

B DL model hyperparameters and settings

Table B-1 Model 1 hyperparameters and settings.

Hyperparameters and settings	Value
Optimizer	Adam
Loss	MSE
Batch size	256
Epochs	30
Sequence length	LSTM-1 : 270 days LSTM-2 : 46 days
Learning rate and scheduling	Epoch < 10 : 1e-3 10 ≤ epoch < 20 : 5e-4 20 ≤ epoch : 1e-4
Layers sizes	LSTM-1 : 64 hidden units; 0.25 dropout LSTM-2 : 64 hidden units; 0.20 dropout Dense : 46 hidden units

Table B-2 Model 2 hyperparameters and settings.

Hyperparameters and settings	Value
Optimizer	Adam
Loss	MSE
Batch size	256
Epochs	30
Sequence length	LSTM-1 : 270 days LSTM-2 : 46 days
Learning rate and scheduling	Epoch < 10 : 1e-3 10 ≤ epoch < 20 : 5e-4 20 ≤ epoch : 1e-4
Layers sizes	LSTM-1 : 128 hidden units FC : 128 hidden units LSTM-2 : 128 hidden units; 0.25 dropout Dense : 46 hidden units

Table B-3 Model 3 hyperparameters and settings.

Hyperparameters and settings	Value
Optimizer	Adam
Loss	MSE
Batch size	256
Epochs	30
Sequence length	LSTM-1 : 270 days LSTM-2 : 46 days
Learning rate and scheduling	Epoch < 10 : 1e-3 10 ≤ epoch < 20 : 5e-4 20 ≤ epoch : 1e-4
Layers sizes	LSTM-1 : 128 hidden units FC : 128 hidden units Dense-2 : 128 hidden units LSTM-2 : 128 hidden units; 0.25 dropout Dense-1 : 46 hidden units

Table B-4 Model 4 hyperparameters and settings.

Hyperparameters and settings	Value
Optimizer	Adam
Loss	MSE
Batch size	256
Epochs	30
Sequence length	LSTM-1 : 270 days LSTM-2 : 46 days
Learning rate and scheduling	Epoch < 10 : 1e-3 10 ≤ epoch < 20 : 5e-4 20 ≤ epoch : 1e-4
Layers sizes	LSTM-1 : 128 hidden units FC : 128 hidden units LSTM : 128 hidden units Dense-2 : 128 hidden units Dense : 1 hidden units

C Cross-validation method

An illustration and details of the cross-validation method used in this study are shown below.

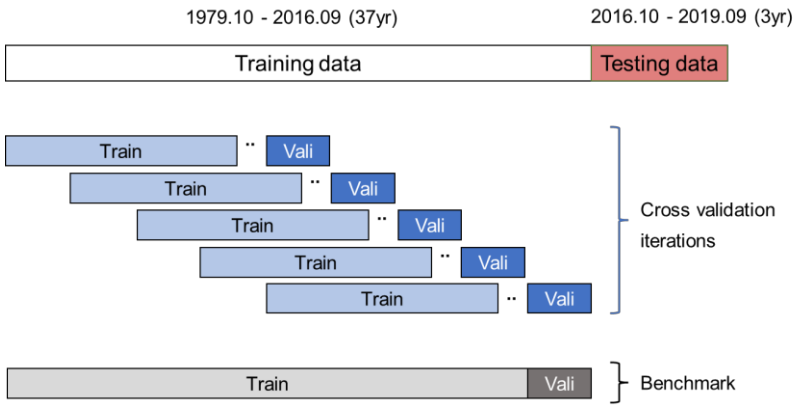


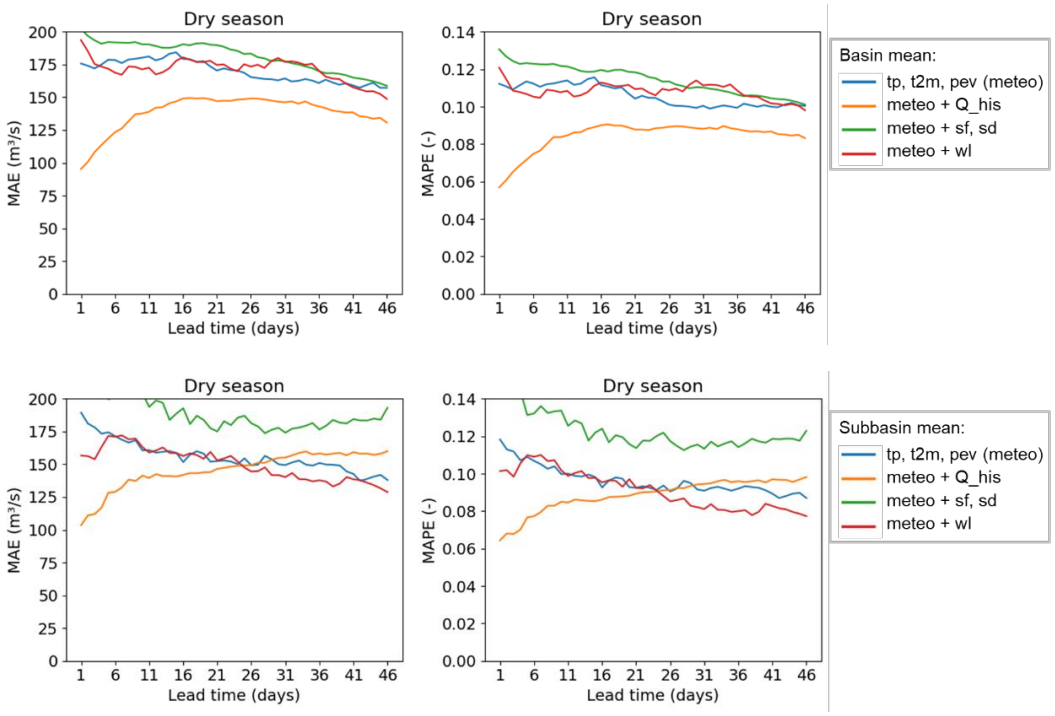
Figure C-1 Illustration of cross validation method “sliding window with gaps” used in this study.

The cross-validation process in this study consists of five iterations, where each iteration involves using 70% (around 25 years) of the total available time series data for training the model. Specifically, the 25-year data is divided into three segments: 21 years for training, a 1-year gap, and 3 years for validation. After the five iterations, five models are obtained. To provide a fair evaluation of their performance, the remaining 3 years of data (which has not been used in any of the iterations) are used as final testing data for each of the five models. Additionally, a sixth model is developed as a benchmark. This model is trained using all the available training data, without any cross-validation splitting, and tested also using testing data.

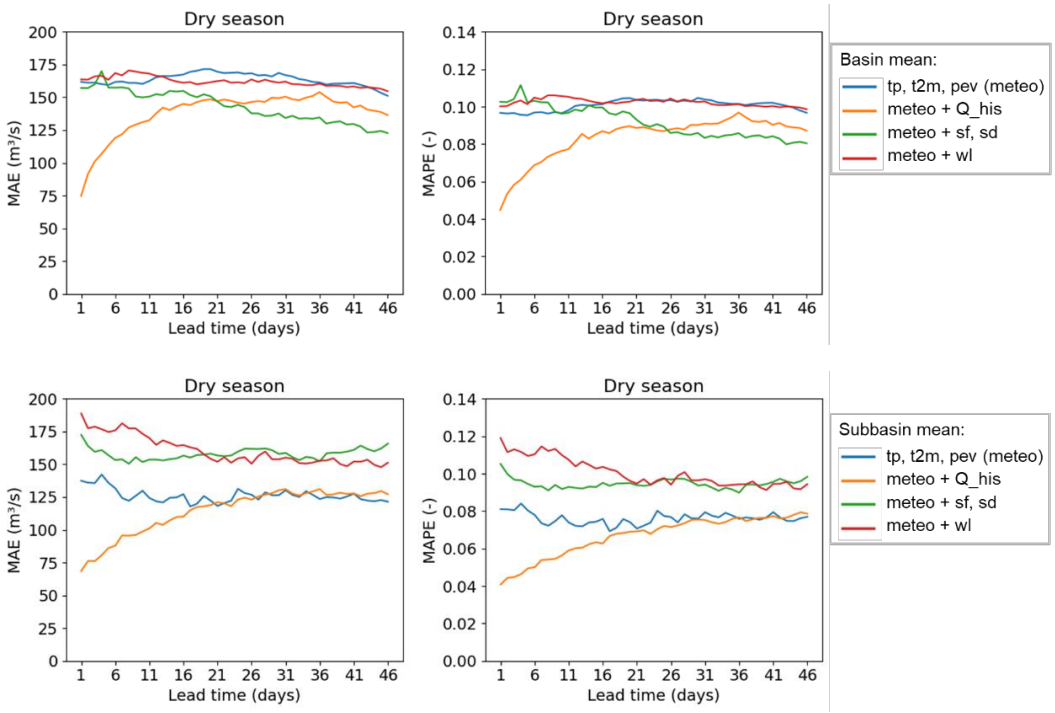
D Quantifying the impact of spatial resolution, input and target variables, and loss functions

D.1 Experiment 1A: spatial resolution

a) Model 1



b) Model 2



c) Model 4

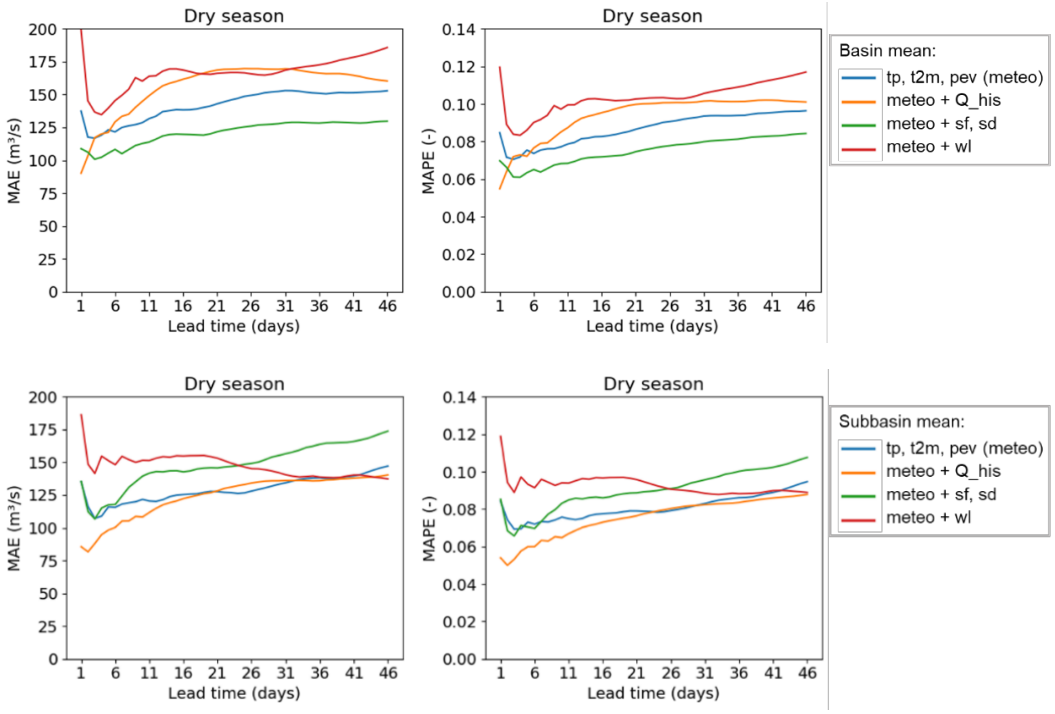
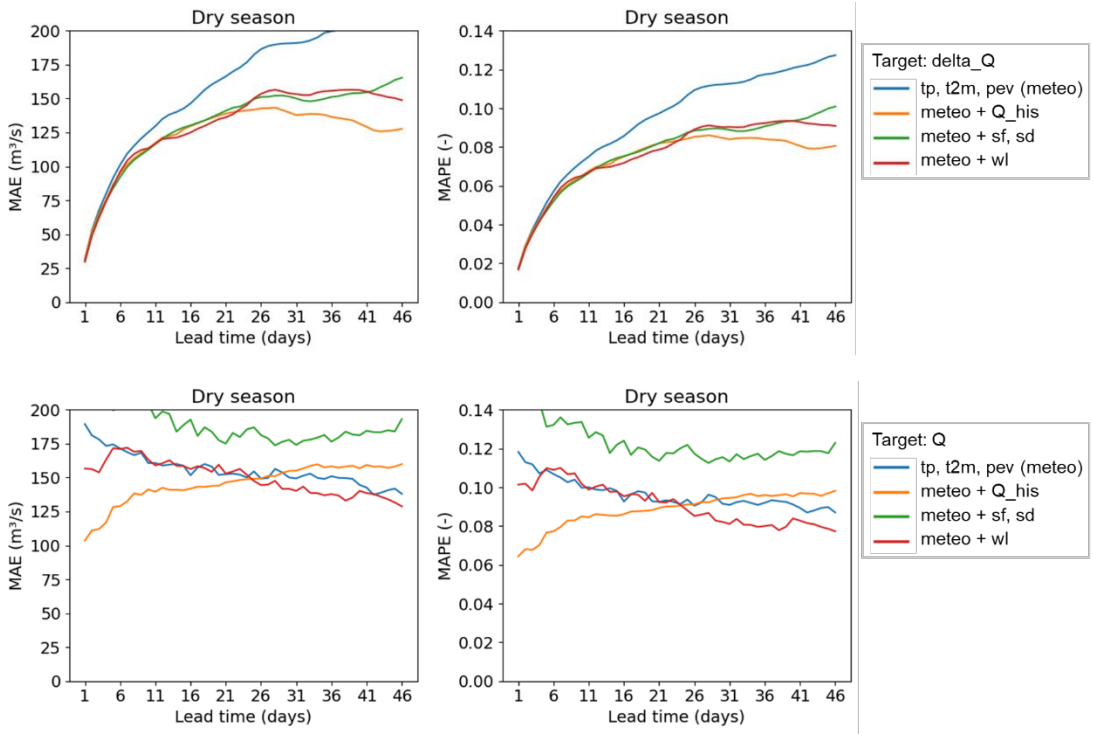


Figure D-1 MAE and MAPE of Experiment 1A results from basin mean approach and subbasin mean approach with various input variables for a) Model 1, b) Model 2, c) Model 4. Basin mean approach refers to the spatial resolution experiment where the spatially distributed variables are processed as mean values across the entire Rhine basin, while subbasin mean approach refers to the spatial resolution experiment where the spatially distributed variables are processed as mean values over the eight subbasins upstream of Lobith.

D.2 Experiment 1B: input and target variables

a) Model 1



b) Model 2

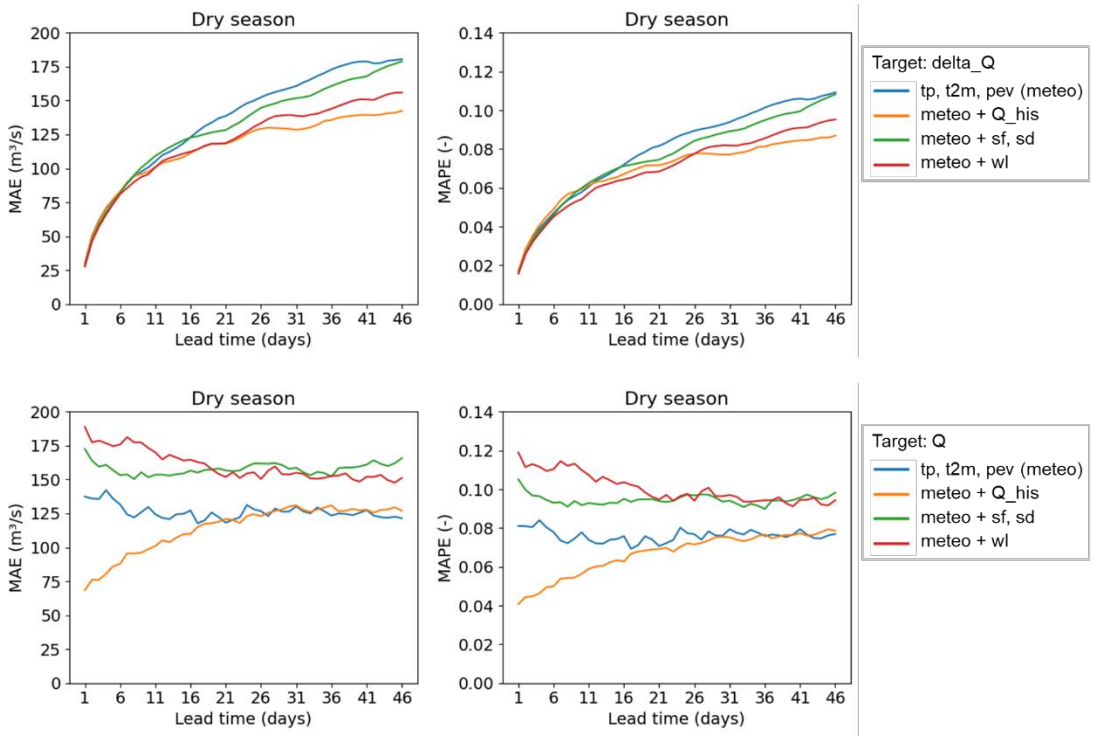
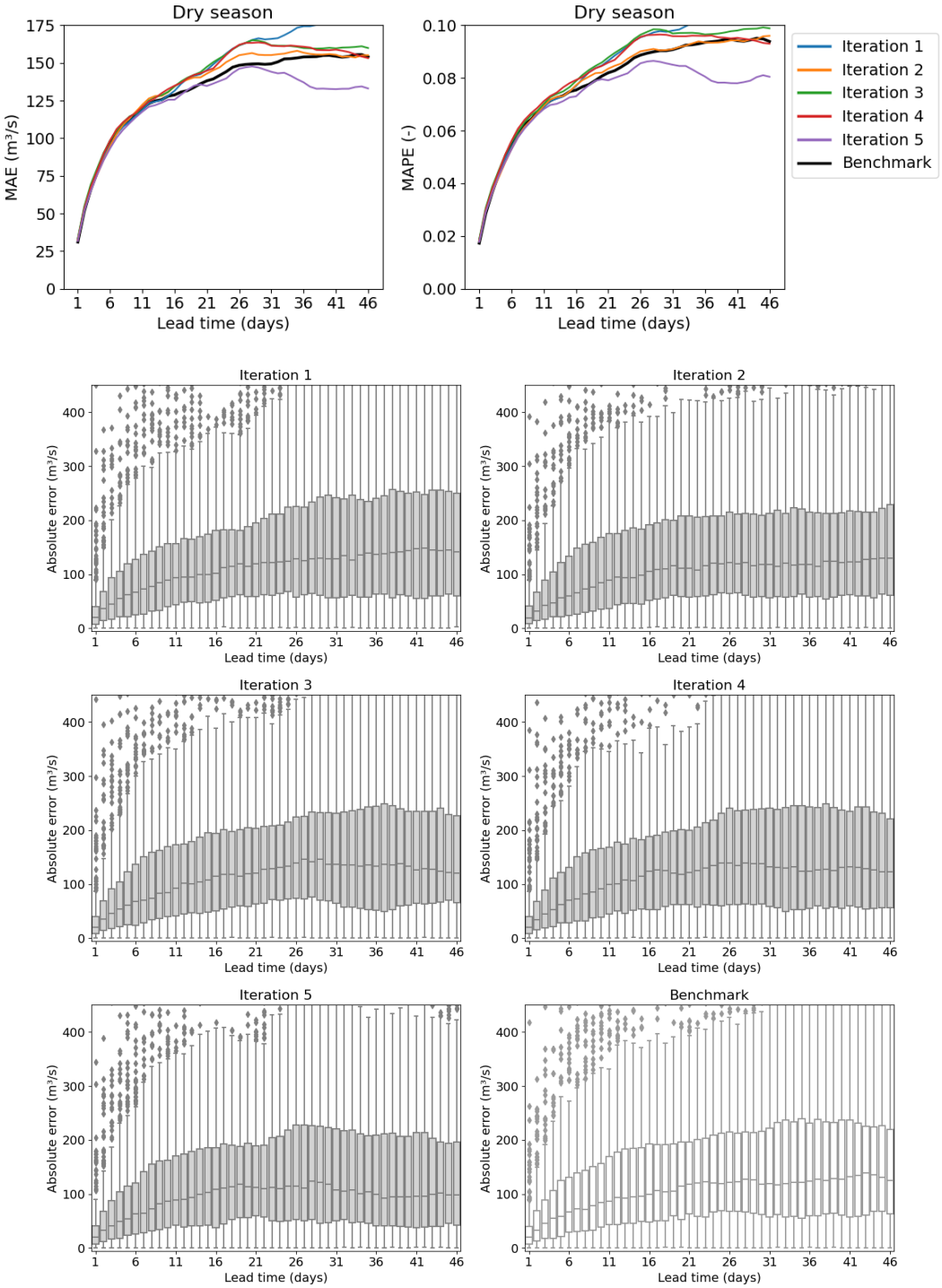


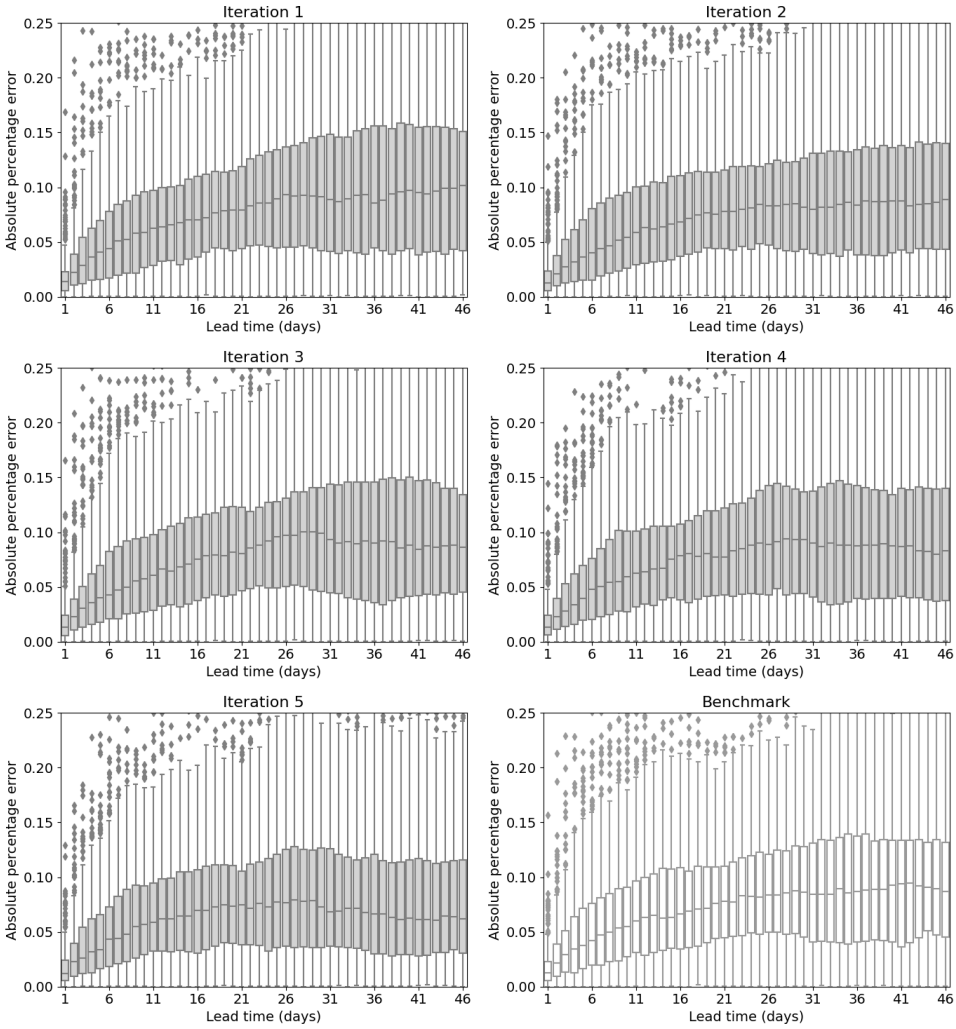
Figure D-2 MAE and MAPE of Experiment 1B results from training the model on different targets, i.e., the time-differenced data (delta_Q) or discharge (Q), with various input variables, for a) Model 1 and b) Model 2.

E Comparing different model architectures

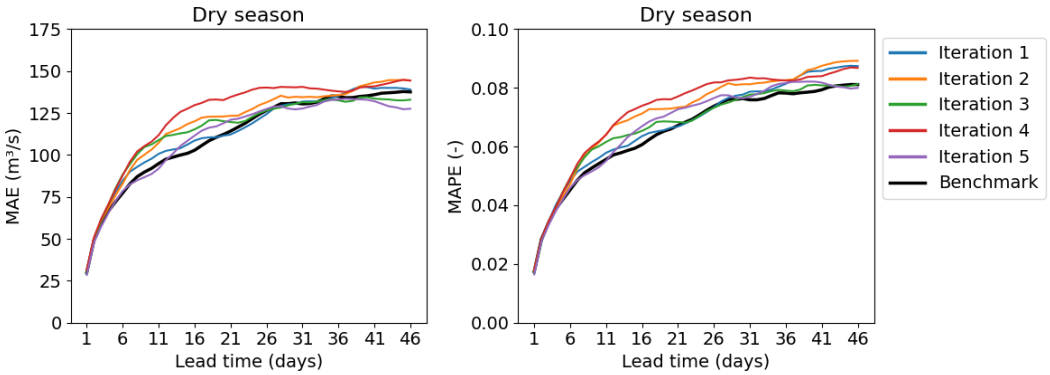
E.1 Experiment 2A: cross-validation

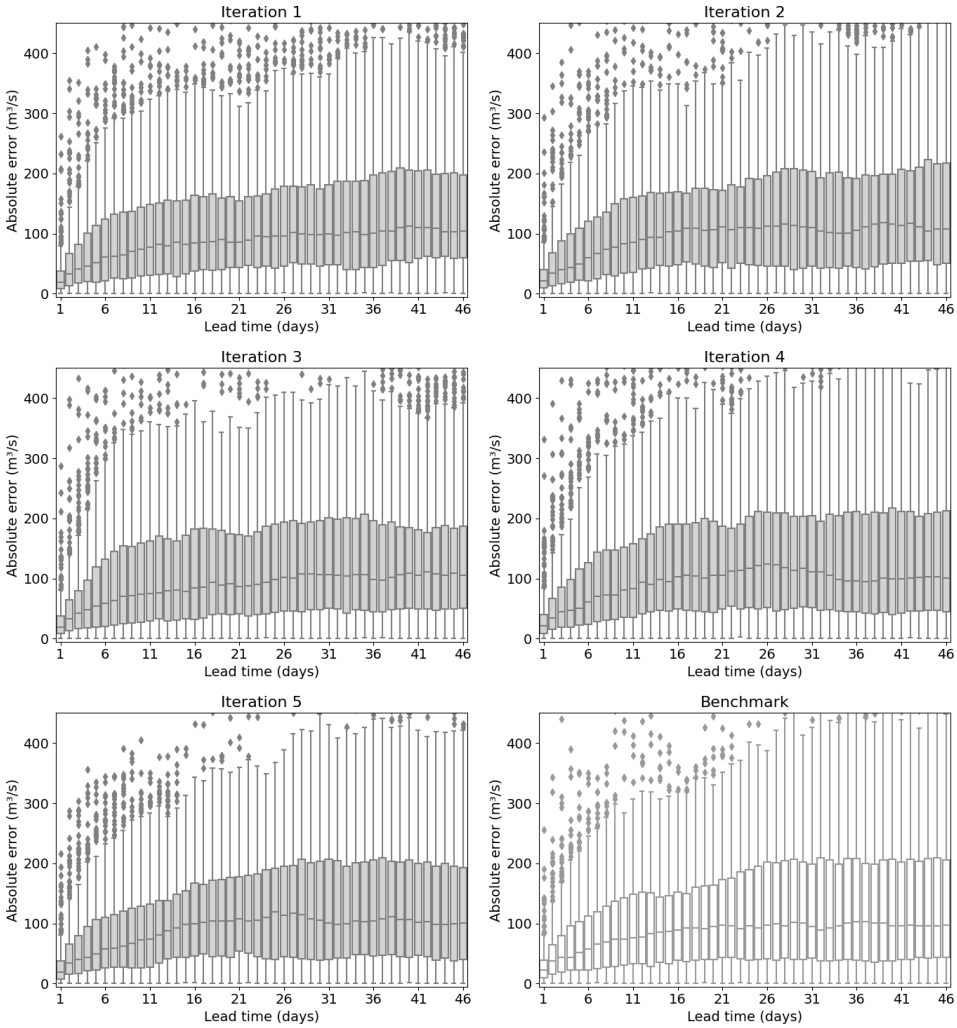
a) Model 1

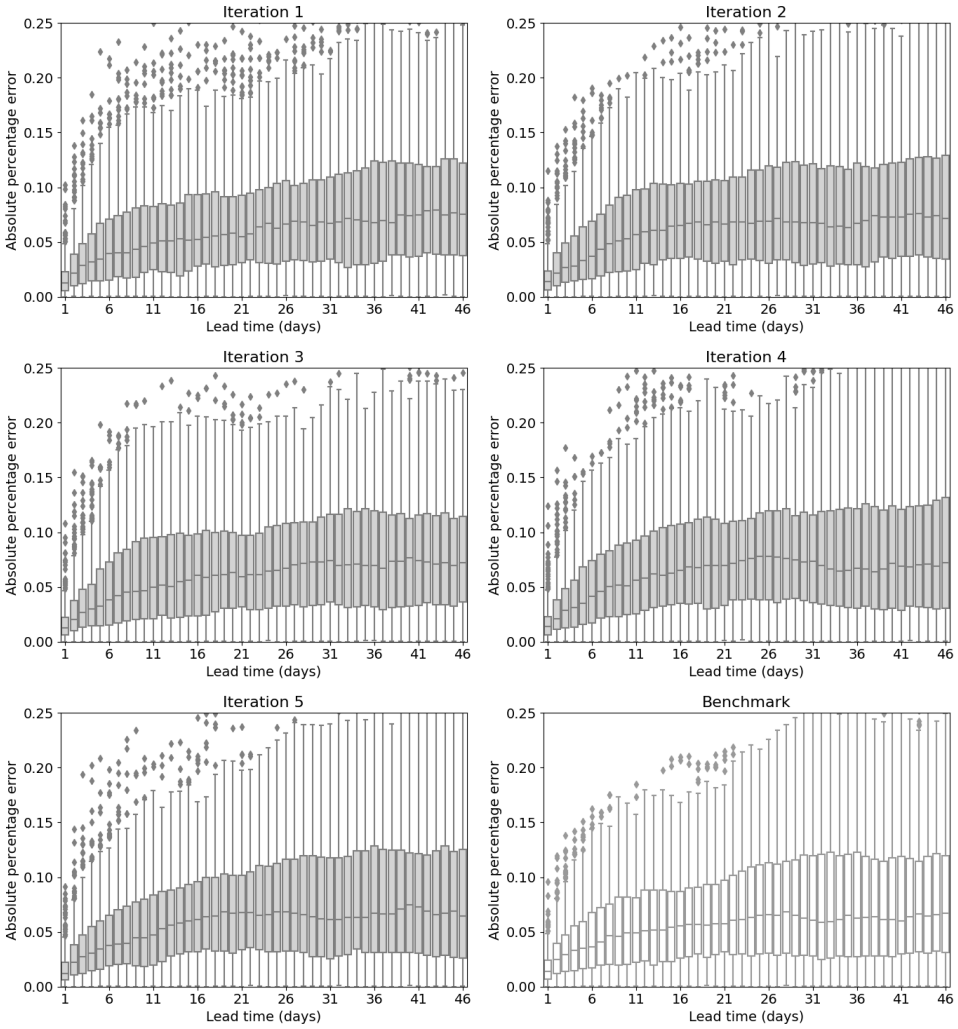



Iteration 1
Iteration 2
Iteration 3
Iteration 4
Iteration 5
Benchmark
Absolute percentage error
Lead time (days)

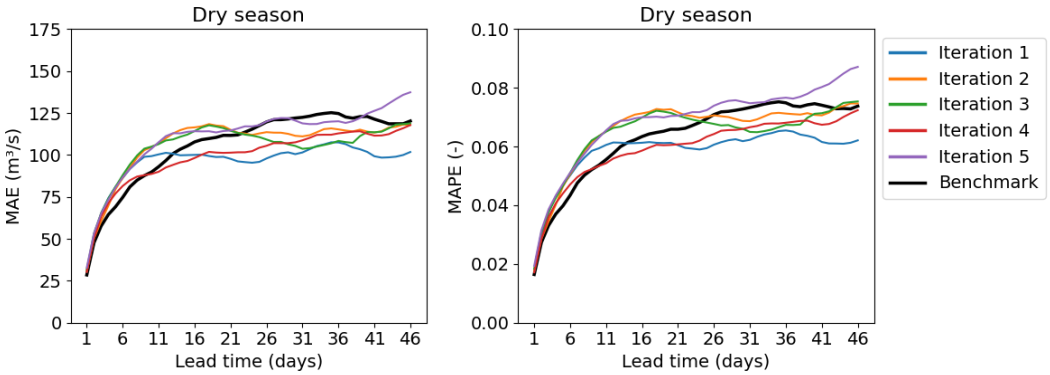
b) Model 2

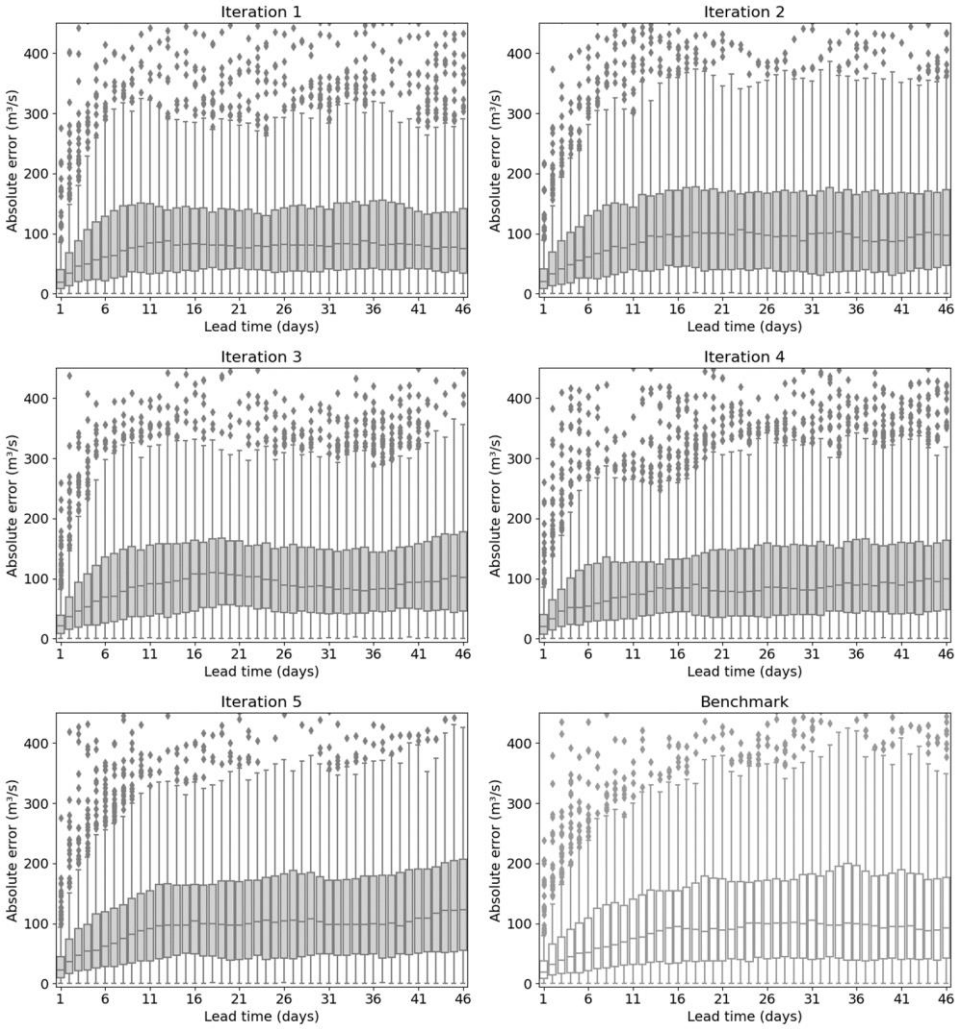

Dry season
Dry season
MAE (m³/s)
MAPE (-)
Lead time (days)
Iteration 1
Iteration 2
Iteration 3
Iteration 4
Iteration 5
Benchmark

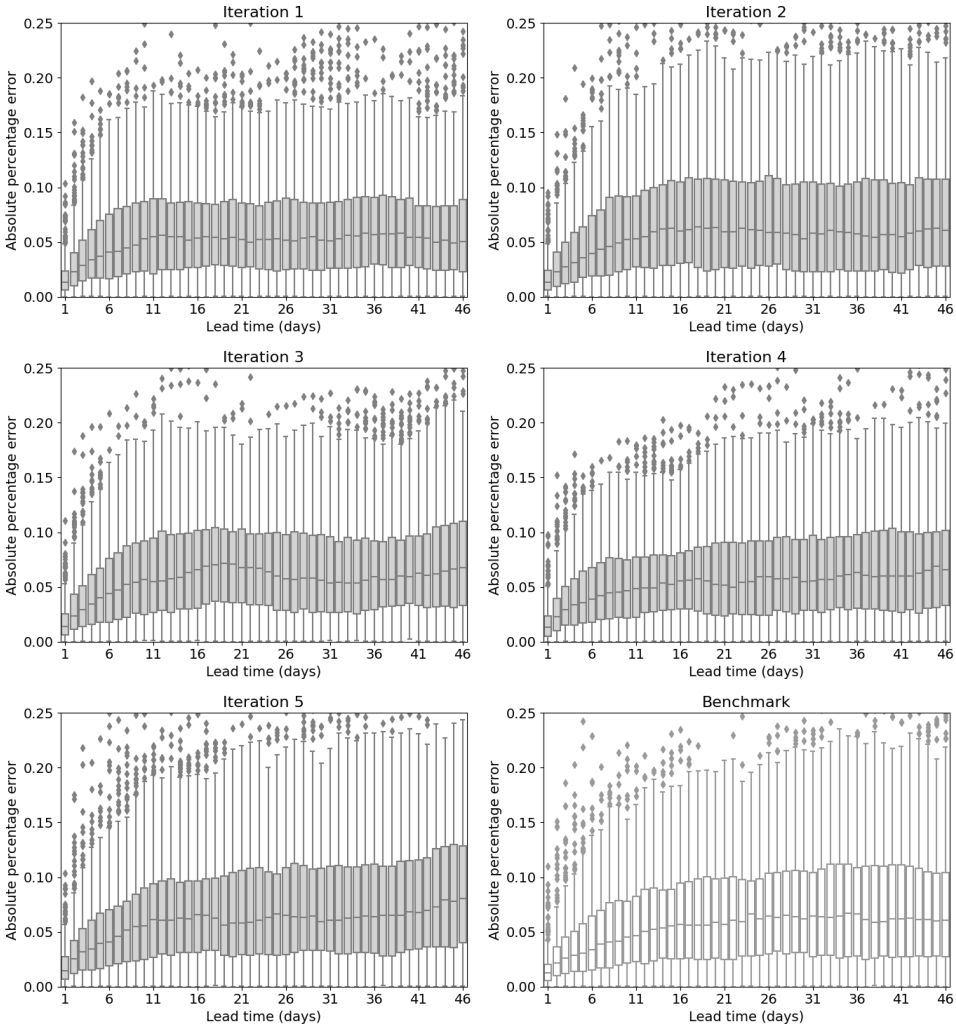

Iteration 1
Iteration 2
Iteration 3
Iteration 4
Iteration 5
Benchmark
Absolute error (m³/s)
Lead time (days)
0
100
200
300
400
1
6
11
16
21
26
31
36
41
46


Iteration 1
Iteration 2
Iteration 3
Iteration 4
Iteration 5
Benchmark
Absolute percentage error
Lead time (days)

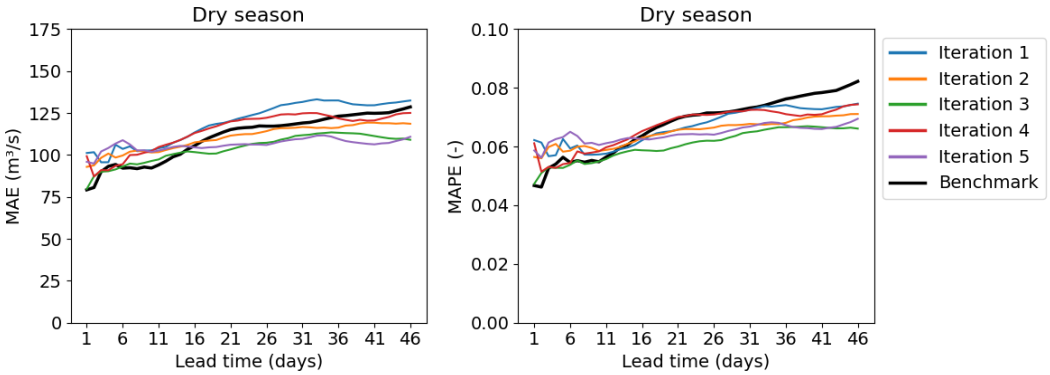
c) Model 3

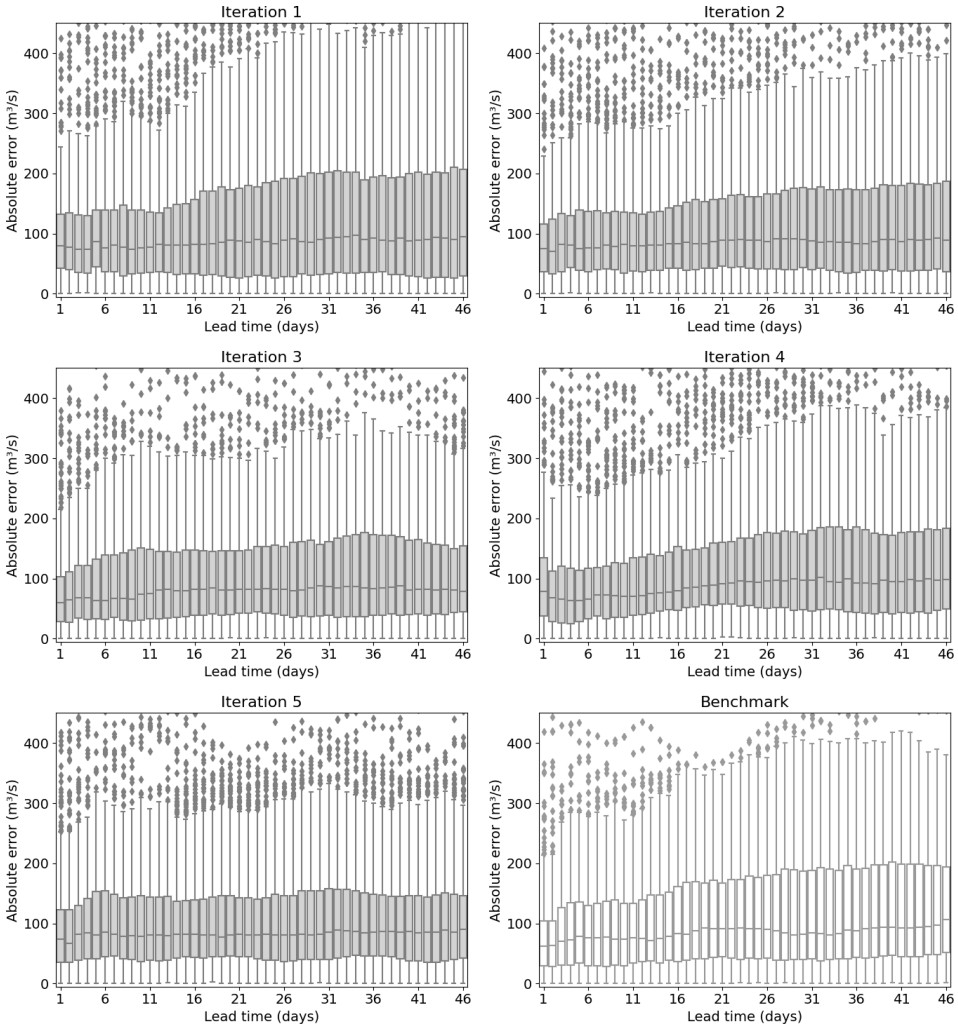

Dry season
Dry season
MAE (m³/s)
MAPE (-)
Lead time (days)
Iteration 1
Iteration 2
Iteration 3
Iteration 4
Iteration 5
Benchmark


Iteration 1
Iteration 2
Iteration 3
Iteration 4
Iteration 5
Benchmark
Absolute error (m³/s)
Lead time (days)
0
100
200
300
400
1
6
11
16
21
26
31
36
41
46


Iteration 1
Iteration 2
Iteration 3
Iteration 4
Iteration 5
Benchmark
Absolute percentage error
Lead time (days)

d) Model 4


Dry season
Dry season
MAE (m³/s)
MAPE (-)
Lead time (days)
Iteration 1
Iteration 2
Iteration 3
Iteration 4
Iteration 5
Benchmark


Iteration 1
Iteration 2
Iteration 3
Iteration 4
Iteration 5
Benchmark
Absolute error (m³/s)
Lead time (days)
0
100
200
300
400
1
6
11
16
21
26
31
36
41
46

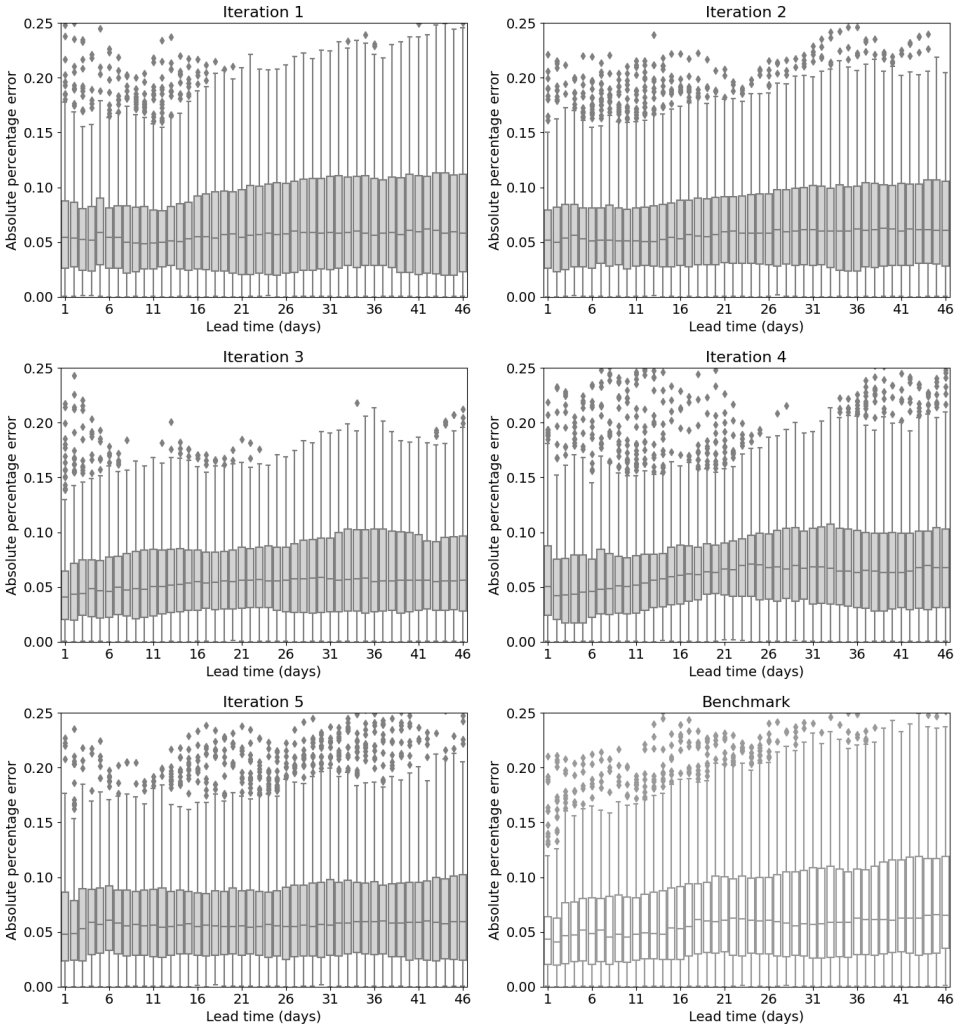


Figure E-1. Cross-validation results for a) Model 1, b) Model2, c) Model 3, and d) Model 4. Iteration 1 to 5 represent the five models trained on different cross-validation time series data splits. Benchmark represent the model trained using all the available training data without any cross-validation splitting.

E.2 Experiment 2B: comparative analysis

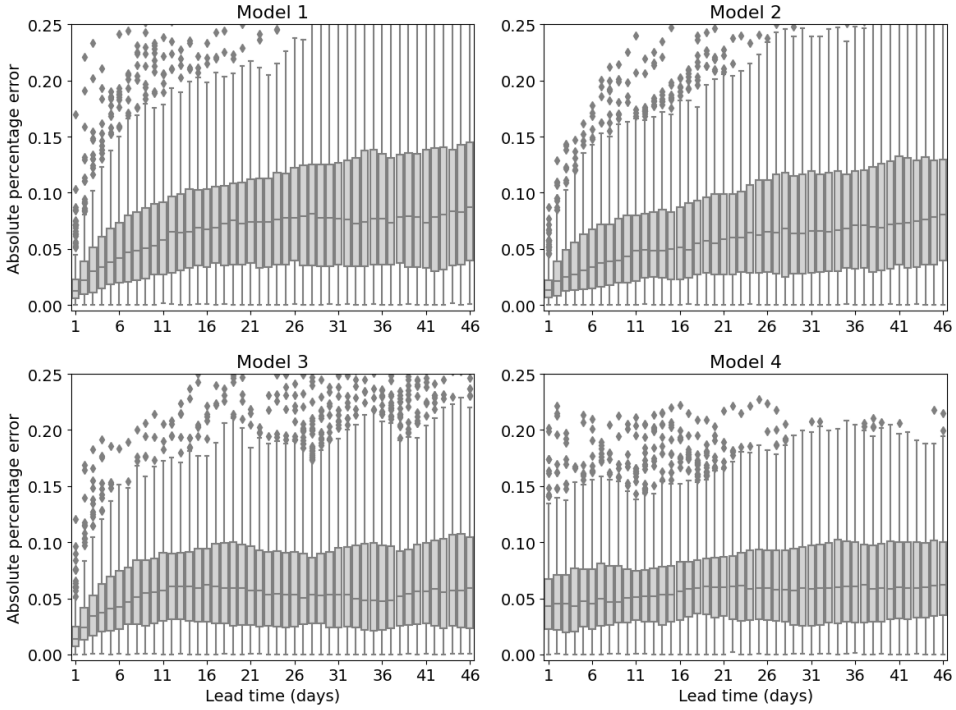


Figure E-2 APE results of different model architectures.

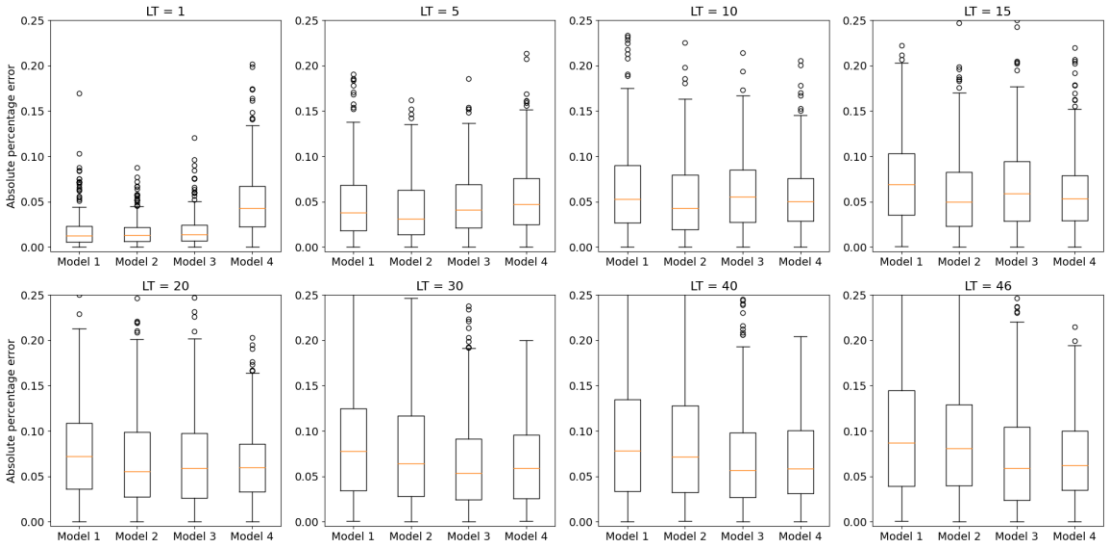


Figure E-3 APE results of different model architectures for several LTs.

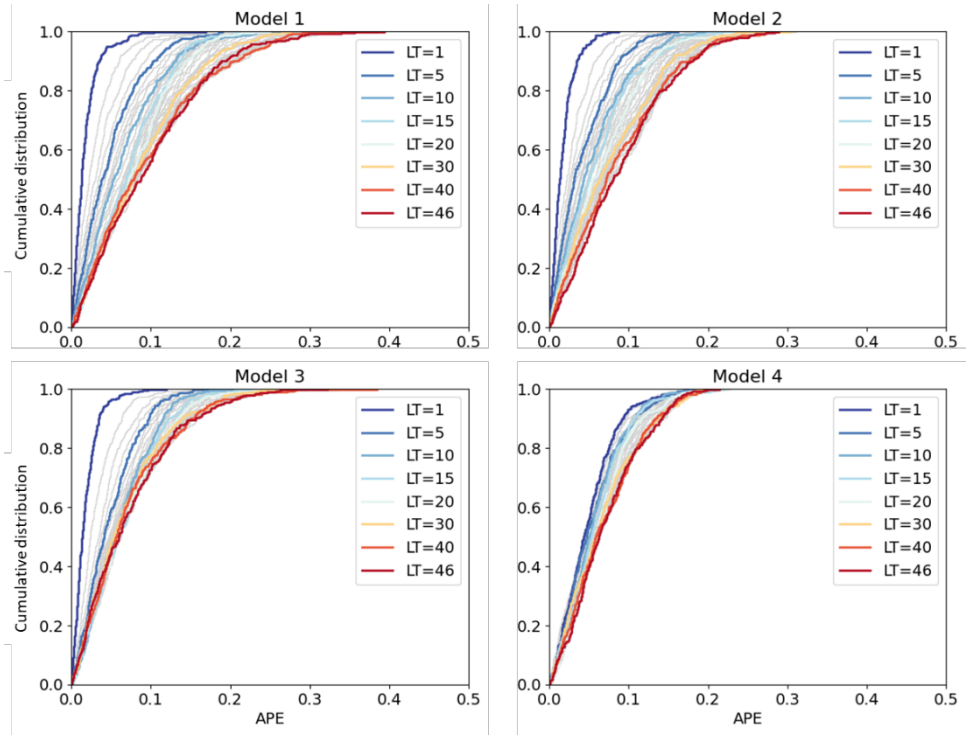


Figure E-4 CDF of APE results of different model architectures. Several LTs are highlighted in colors. The grey lines are the results for other LTs.

F Comparing the DL model with physically-based models

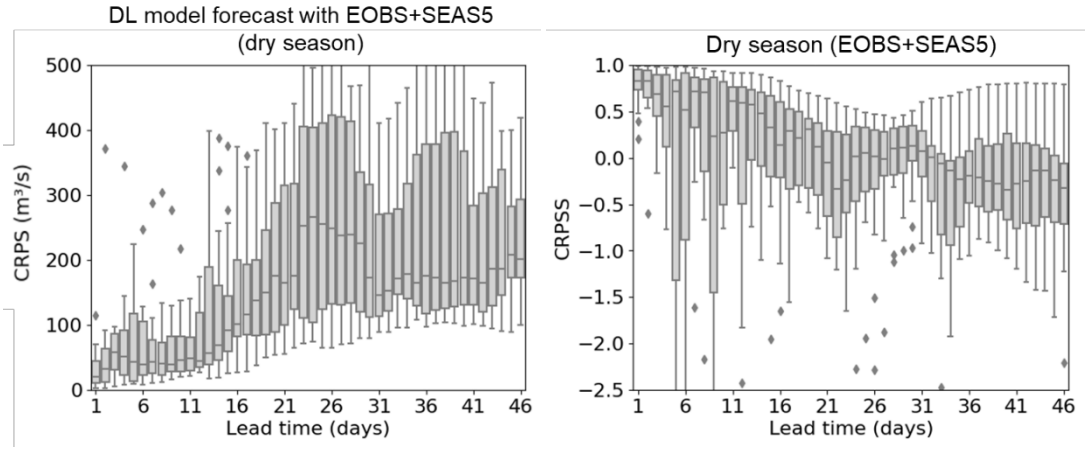


Figure F-1 CRPS of the DL model forecast results, and CRPSS of the results compared to wflow model forecasts. EOBS is used for both X1 (input for LSTM-1) and X2 (input for LSTM-2) in training mode, and used for X1 together with SEAS5 for X2 in the forecast mode.

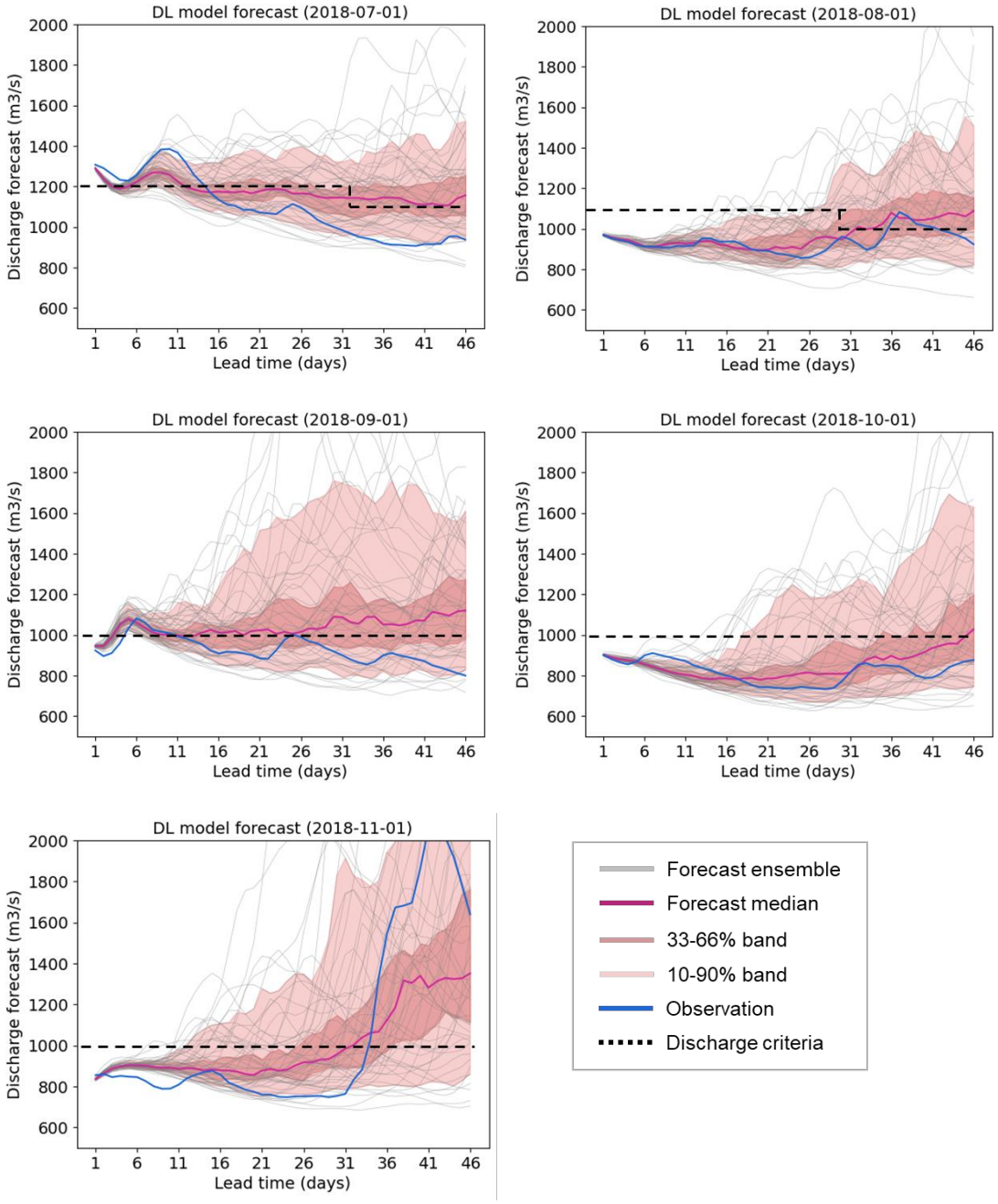


Figure F-2 DL model forecast results for the drought event in 2018 with SEAS5.

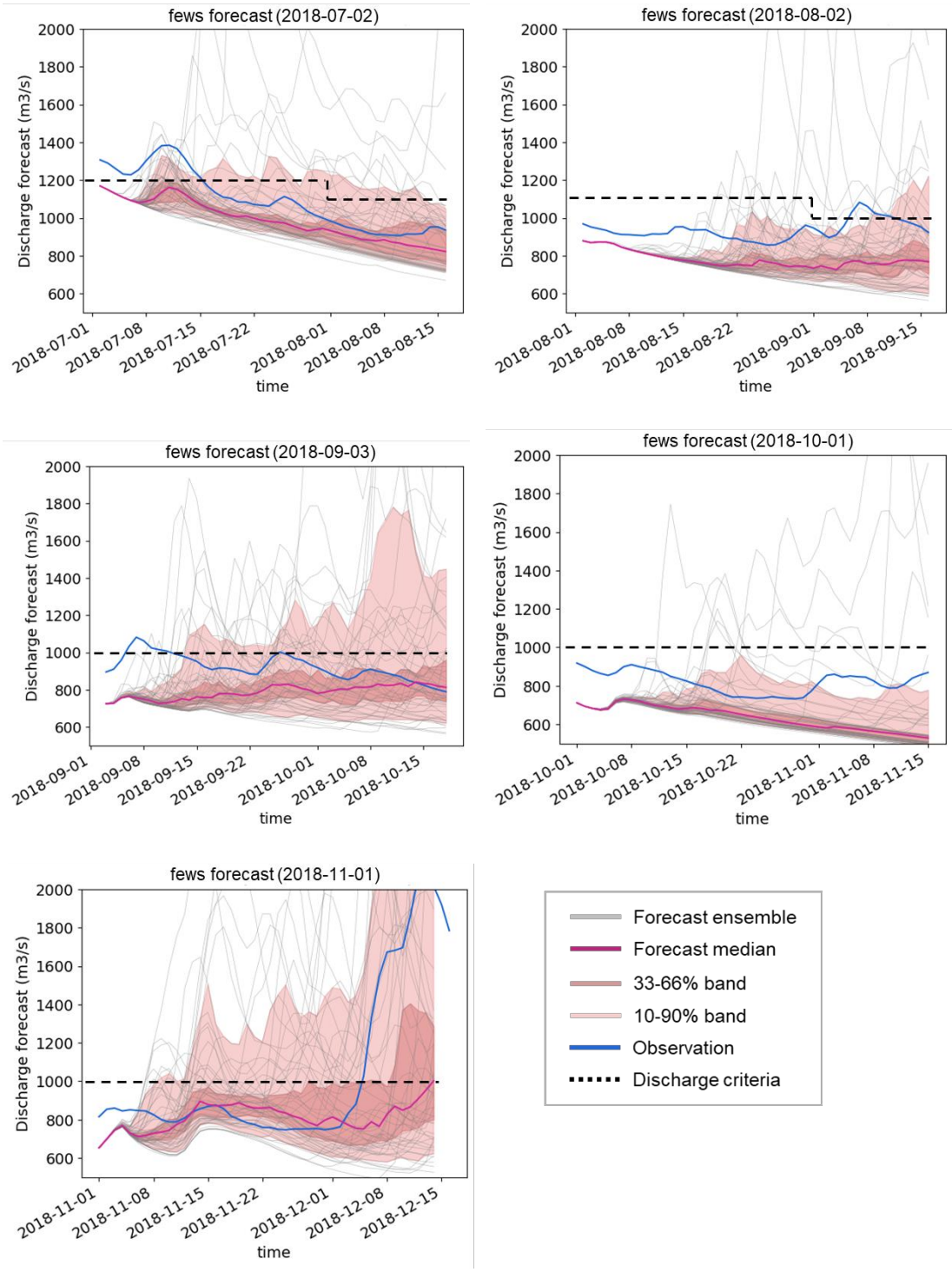


Figure F-3 FEWS-Rhine forecast results for the drought event in 2018 with ENS extended.