Advancing Operational Hydrology Through Deep Transfer Learning: Multi-Day Streamflow Forecasting in Central Asian Mountains

Master Thesis

Author: Nicolas Lazaro

Supervisors: Prof. Dr. Peter Molnar
Dr. Tobias Siegfried

Date: August 19, 2026

Abstract

Deep learning has revolutionized hydrological forecasting, yet its application in data-sparse regions remains challenging. This thesis examines whether transfer learning—pre-training models on data-rich source domains before fine-tuning them on target domains—can improve operational streamflow forecasting in mountainous Central Asian catchments. Through systematic experiments on 77 catchments (16 in Tajikistan, 61 in Kyrgyzstan), I evaluate four deep learning architectures (EA-LSTM, TFT, TSMixer, TiDE) across forecast horizons from 1 to 10 days, comparing regional transfer learning (pre-training on neighboring countries) and global transfer learning (pre-training on up to 6,690 catchments from the Caravan dataset). Results demonstrate that transfer learning consistently improves forecasting performance, with median Nash-Sutcliffe Efficiency gains ranging from 0.01 to 0.07, and greater benefits in data-limited Tajikistan compared to data-rich Kyrgyzstan. However, analysis using the Remaining Skill Captured metric reveals that these improvements primarily enhance temporal extrapolation of autoregressive streamflow signals rather than meteorological sensitivity. The comparison between volume-based and similarity-based source domain selection strategies reveals no consistent superiority; however, similarity-based selection achieves comparable performance with 81% fewer catchments. While simpler architectures (TSMixer, TiDE) show strong benefits from regional transfer learning, EA-LSTM demonstrates improvements primarily at shorter forecast horizons. In contrast, TFT shows minimal regional response, with both complex models achieving greater gains from global-scale pre-training datasets. All architectures converge to similar performance levels after global transfer learning, suggesting a performance ceiling imposed by strong temporal autocorrelation in mountainous streamflow. Methodologically, this work develops techniques for training on larger-than-RAM datasets and demonstrates that preprocessing choices critically impact model performance, with per-basin normalization causing catastrophic failures that transfer learning reduces by up to 90%. These findings suggest that while transfer learning provides consistent benefits for general water management applications, the model's enhanced reliance on the autoregressive pattern rather than improved sensitivity to meteorological forcings limits its utility for flood forecasting, which requires sensitivity to extreme meteorological events.

Acknowledgements

Tobias, thank you for the trust and freedom you gave me throughout this project. Your genuine excitement about my ideas and willingness to let me explore without constraints made this research possible.
Sandro, our discussions have been invaluable. Thank you for sharing your deep learning expertise and for always being willing to challenge my interpretations. When I got stuck, you helped me see alternative paths forward.
Professor Molnar, thank you for supervising my work. I appreciate your openness to new technologies.
Sophia, thank you for listening to my progress updates and asking the thoughtful questions that forced me to clarify my thinking.
Michelle, thank you for your patience with all the late evenings and weekend work, as well as for your constant support throughout this project.
Finally, I thank my family for their encouragement and support throughout my studies.

Contents

List of Figures

List of Tables

List of Abbreviations

AI Artificial Intelligence

AUS Australia

BR Brazil

CA Central Asia

CAMELS Catchment Attributes and Meteorology for Large-sample Studies

CAMELS-AUS CAMELS Australia

CAMELS-BR CAMELS Brazil

CAMELS-CH CAMELS Switzerland

CAMELS-CL CAMELS Chile

CAMELS-GB CAMELS Great Britain

CH Switzerland

CL Chile

CNN-LSTM Convolutional Neural Network - Long Short-Term Memory

DBA DTW Barycenter Averaging

DEM Digital Elevation Model

DTW Dynamic Time Warping

EA-LSTM Entity-Aware Long Short-Term Memory

ECMWF European Centre for Medium-Range Weather Forecasts

GB Great Britain

GDP Gross Domestic Product

GIS Geographic Information System

GoF Goodness of Fit

GPU Graphics Processing Unit

HII Human Influence Index

HYSETS Hydrometeorological Sandbox - École de Technologie Supérieure

IAHS International Association of Hydrological Sciences

KGE Kling-Gupta Efficiency

LamaH Large-Sample Data for Hydrology

LamaH-CE LamaH Central Europe

LSH Large Sample Hydrology

LSTM Long Short-Term Memory

MAE Mean Absolute Error

MoU Memorandum of Understanding

MSE Mean Squared Error

NHMS National Hydrometeorological Services

NLDAS North American Land Data Assimilation System

NSE Nash-Sutcliffe Efficiency

ODE Ordinary Differential Equation

PUB Predictions in Ungauged Basins

RAM Random Access Memory

RevIN Reversible Instance Normalization

RMSE Root Mean Square Error

RQ Research Question

RR-Former Rainfall-Runoff Former

RSC Remaining Skill Captured

SAC-SMA Sacramento Soil Moisture Accounting

SAPPHIRE Smart & Precise Prognostic Hydrology for Innovative Risk Management and Resource Use Efficiency

SRTM Shuttle Radar Topography Mission

TFT Temporal Fusion Transformer

TiDE Time Series Dense Encoder

TL Transfer Learning

TPE Tree-structured Parzen Estimator

TSMixer Time Series Mixer

USA United States of America

UTC Coordinated Universal Time

vCPU Virtual Central Processing Unit

VRAM Video Random Access Memory

1 Introduction

1.1 The Hydrological Regionalization Challenge

The International Association of Hydrological Sciences (IAHS), the world's oldest scientific society dedicated to hydrology, has long fostered advancements through global collaboration, research, and inclusive knowledge exchange (International Association of Hydrological Sciences, 2025). A key initiative, the Predictions in Ungauged Basins (PUB) Decade (2003–2012), aimed to improve hydrological prediction in areas with limited or no measurement data by enhancing process understanding and modelling approaches (Hrachowitz et al., 2013; Sivapalan et al., 2003). The subsequent Panta Rhei Decade (2013–2022) built upon this foundation, focusing on "Change in Hydrology and Society" to understand the dynamic, two-way interactions between hydrological systems and human activities (Montanari et al., 2013). Together, these initiatives represent two decades of concerted effort to address the longstanding challenge of hydrological model regionalisation.

Despite the universal consistency of the physical laws governing hydrology—such as the conservation of mass, momentum, and energy—a persistent paradox remains: hydrological models traditionally perform better when calibrated to individual catchments than when applied across multiple catchments (Hrachowitz et al., 2013; Nearing et al., 2021b). This highlights the difficulty of effectively transferring hydrological knowledge from gauged to ungauged basins (Prieto et al., 2019). Historically, various explanations were proposed. Beven (2000) highlighted the varying importance of different hydrological processes active at different time scales in different catchments, thereby emphasising the "uniqueness of place" as a consequence of natural variability. This perspective suggested that inherent catchment heterogeneity might fundamentally limit the transferability of hydrological models. Alternatively, some argued that the lack of regionalisation success stemmed from insufficient observations in terms of type, scale, and scope to discover underlying similarities between catchments (Hrachowitz et al., 2013). However, recent developments in deep learning have begun to challenge both explanations, suggesting that the information needed for successful generalisation may have been present in available data all along, but that the hydrological community lacked the tools to extract it effectively (Nearing et al., 2021b). This represents a paradigm shift from individual catchment calibration toward large-sample, data-driven approaches.

1.2 Large Sample Hydrology Enables Deep Learning

The Catchment Attributes and Meteorology for Large-sample Studies (CAMELS) initiative marks a cornerstone in this paradigm shift. Large Sample Hydrology (LSH) datasets leverage standardised data from large samples of catchments to derive robust conclusions about hydrological processes and models (Addor et al., 2020). Beginning with the CAMELS dataset for the contiguous United States (Addor et al., 2017; Newman et al., 2015), the project provided the research community with a comprehensive dataset spanning 671 catchments with minimal human disturbances across the United States. The dataset contains catchment-aggregated meteorological forcing data and observed streamflow at the daily timescale, with most records extending from 1980 to 2014. The meteorological data are derived from three gridded data sources—Daymet (1 km resolution), Maurer (12 km resolution), and North American Land Data Assimilation System (NLDAS) (12 km resolution)—and include precipitation, shortwave radiation, maximum and minimum temperature, vapour pressure, and snow water equivalent. Additionally, CAMELS provides 27 static catchment attributes describing topography, climate, geology, soil, and land cover characteristics. Following this standard, similar datasets were developed for other regions, including CAMELS Chile (CAMELS-CL) (Alvarez-Garreton et al., 2018) and CAMELS Switzerland (CAMELS-CH) (Höge et al., 2023). More recently, the Caravan dataset (Kratzert et al., 2023) has extended this principle globally, standardising and aggregating data from multiple LSH datasets to create a unified framework for global large-sample hydrology. This provision of standardised, high-quality data from large and diverse sam-

ples of catchments created the necessary foundation for applying data-intensive methods to hydrology, enabling a new generation of models that could learn from multiple catchments simultaneously rather than being calibrated to individual basins.

1.3 Deep Learning Success: From Single to Multiple Basins

Leveraging these large-sample datasets, Kratzert et al. (2018) published the seminal paper *Rainfall–runoff modelling using Long Short-Term Memory (LSTM) networks*. They showed that a relatively simple LSTM architecture (Hochreiter and Schmidhuber, 1997)—a sequence-to-sequence model with daily meteorological inputs from 241 catchments in the CAMELS dataset and streamflow as output—could match and, in many cases, surpass the performance of a process-based model (Sacramento Soil Moisture Accounting (SAC-SMA) coupled with Snow-17). LSTMs incorporate memory cells that function analogously to the state variables in traditional bucket-type hydrological models: they store and update an internal representation of catchment conditions over time in response to input signals, such as precipitation and temperature. Unlike traditional models, however, the LSTM learns the structure and dynamics of this memory directly from data without requiring prior specification of storage components, routing schemes, or process equations. This gives the model greater flexibility in capturing non-linear relationships and allows it to reproduce runoff dynamics across a wide range of catchments, despite (or perhaps because of (Nearing et al., 2021b)) including no hydrologically informed components (Kratzert et al., 2018). This seminal work established that deep learning is viable for rainfall-runoff modelling.

Kratzert et al. (2019a) extended this work by training LSTMs on larger datasets to evaluate performance in ungauged basins. Using 531 CAMELS catchments with meteorological forcing and static attributes, their single LSTM model achieved a median NSE of 0.69 in out-of-sample predictions, surpassing the locally calibrated SAC-SMA model (NSE = 0.64) despite having no access to in-basin training data. This demonstrated that LSTMs trained on multiple basins could generalise to unseen catchments more effectively than traditional models could be regionalised. This finding was later reinforced in a position paper by Kratzert et al. (2024), which argued that training on single basins is a methodological flaw that fails to leverage the primary strength of deep learning. Kratzert et al. (2019c) further refined this approach by systematically integrating static catchment attributes alongside meteorological time series, confirming that CAMELS attributes contain sufficient information to differentiate between diverse rainfall-runoff behaviours. They introduced the EA-LSTM, which features a separate static input gate that processes catchment attributes independently of dynamic meteorological inputs. This architecture learns catchment embeddings that activate different parts of the network for different basin types, enabling distinct rainfall-runoff responses while sharing learned representations among similar basins. The EA-LSTM outperformed both standard LSTMs and traditional hydrological models, while producing interpretable embeddings that align with established hydrological knowledge. The development of EA-LSTM marked a transition from applying generic deep learning architectures to designing models purpose-built for rainfall-runoff modelling, establishing that both multi-basin training and static attributes are essential for deep learning success in rainfall-runoff modelling.

1.4 Architectural Advances and Operational Capabilities

The success of the EA-LSTM as the first purpose-built rainfall-runoff model catalysed three parallel evolutionary paths in the field. **Purpose-built, data-driven** approaches continued to develop architectures specifically tailored for rainfall-runoff processes. For example, Yin et al. (2022) introduced Rainfall-Runoff Former (RR-Former), which adapts Transformer (Vaswani et al., 2017) attention mechanisms for rainfall-runoff modelling and achieves superior multi-day forecasting performance (median NSE of 0.8265 versus LSTM's 0.7448 for 7-day predictions). Anderson and Radić (2022) developed Convolutional Neural Network - Long Short-Term Memory (CNN-LSTM) networks that combine spatial

pattern recognition with temporal modelling to capture the spatiotemporal nature of meteorological forcing. **Hybrid physics-informed models** have emerged to bridge the performance-interpretability gap, including neural Ordinary Differential Equations (ODEs) (Höge et al., 2022), differentiable parameter learning frameworks (Feng et al., 2022), and physics-informed neural networks (He et al., 2025), which maintain physical consistency while approaching the performance of pure deep learning. **General time series models** from broader machine learning advances were systematically applied to rainfall-runoff forecasting, with researchers exploring Transformer variants, time series attention models, and large language models for rainfall-runoff forecasting tasks (Liu et al., 2025). Systematic benchmarking across these diverse architectures has revealed that different models excel at different rainfall-runoff tasks, with performance depending on task complexity and forecast horizon. Liu et al. (2025) demonstrated that LSTM models perform best in memory-dependent regression tasks. In contrast, attention-based models gradually surpass LSTM performance as tasks become more complex, with Transformers particularly excelling at longer forecast horizons where autoregressive signals weaken. This architectural diversification validates that the benefits of deep learning in rainfall-runoff modelling extend beyond LSTM's initial success, establishing that architecture choice should align with specific forecasting requirements and providing a foundation for robust operational forecasting systems.

To achieve the highest possible model performance and for operational water management applications to benefit from deep learning, it is essential to incorporate near-real-time streamflow observation data into the model input (Nearing et al., 2021a). Feng et al. (2020) demonstrated that integrating previous streamflow observations with meteorological forcing improves model performance, achieving record median NSE values of 0.86 across continental scales using an LSTM with data integration. This approach tested different methods for incorporating past streamflow—either directly as lagged inputs or through convolutional neural network units—showing that there are multiple effective ways to leverage the autocorrelation structure inherent in streamflow time series. The benefits of incorporating observed streamflow are particularly pronounced in regions with high flow autocorrelation, such as mountainous and snow-dominated catchments (Lin et al., 2024).

While the advances described thus far primarily focused on rainfall-runoff *modelling*—predicting current streamflow using data up to and including the present day (Nearing et al., 2021a)—operational water management requires forecasting capabilities that extend beyond the present to predict streamflow days or weeks in advance. Deep learning has successfully demonstrated multi-day ahead streamflow forecasting capabilities, with different architectures excelling at different forecast horizons. As mentioned previously, Yin et al. (2022) introduced RR-Former, which adapts Transformer attention mechanisms for rainfall-runoff forecasting and achieves superior multi-day performance. Yin et al. (2021) further established that sequence-to-sequence architectures can effectively handle multi-step-ahead predictions, enabling operational forecasting systems that provide early warnings for flood events and support water resource planning decisions.

1.5 Transfer Learning: Extending Deep Learning Globally

Building on these advances, Transfer Learning (TL) emerged as a technique to further enhance hydrological model performance by leveraging knowledge from data-rich regions to improve predictions in data-sparse regions (Tan et al., 2018). Transfer learning relaxes the traditional machine learning assumption that training and testing data must come from identical distributions, enabling models to generalise knowledge across varying hydrological conditions. Tan et al. (2018) classify transfer learning into four categories: instance-based methods that select and emphasise the most relevant examples from data-rich regions; mapping-based methods that transform data from different regions into a unified, comparable format; network-based methods that take a model trained on one region and adapt it for another region through weight initialisation and fine-tuning; and adversarial-based methods that train models to

focus on universal hydrological patterns rather than region-specific characteristics. Hydrological studies primarily employ network-based transfer learning, utilising pre-trained models for weight initialisation, followed by fine-tuning on target domain data, with varying approaches to updating network components during fine-tuning.

Ma et al. (2021) demonstrated that TL works effectively across continents by training LSTM models on the data-rich CAMELS dataset and transferring them to data-sparse regions in Asia, Europe, and South America. Their LSTM architecture took meteorological forcing and static catchment attributes as inputs (without incorporating observed streamflow data) and achieved state-of-the-art results with median NSE values of 0.75, 0.87, and 0.86 for China, Chile, and Great Britain, respectively. The results revealed that hydrological dynamics worldwide share commonalities that can be leveraged across different continents.

Khoshkalam et al. (2023) extended TL to operational forecasting by demonstrating its effectiveness with data integration—incorporating past observed streamflow to create autoregressive models. Their study demonstrated that TL models, which combine pre-training on CAMELS with data integration, achieved median NSE values of 0.953 in snow-dominated watersheds in Quebec, Canada. This establishes that transfer learning can improve predictions in gauged basins where streamflow observations are available, making it compatible with the autoregressive modelling approaches that underlie operational forecasting systems.

1.6 Literature Gap

Summarising: the literature of the past seven years has shown that deep learning works for rainfall-runoff modeling, that training on multiple basins improves performance, that static attributes enhance model performance, that different architectures excel at different tasks, that models can perform multi-day streamflow forecasting, that incorporating past observed streamflow is crucial for operational systems, and that transfer learning enables knowledge transfer across regions—**we have not systematically tested whether these advances work effectively together for operational forecasting applications**. Specifically, no study has comprehensively evaluated how transfer learning performs across multiple architectures, forecast horizons, and data availability scenarios in operational autoregressive forecasting contexts. This represents a gap between the individual advances in deep learning hydrology and their integrated application for real-world water resources management, where multi-day forecasting capabilities with autoregressive components are essential for flood warning systems and water resource planning.

1.7 Research Objectives and Contributions

To address this literature gap, this thesis provides an evaluation of transfer learning effectiveness across multiple deep learning architectures, forecast horizons, and data availability scenarios in operational autoregressive forecasting contexts. I investigate whether the individual advances demonstrated in recent deep learning hydrology research—multi-basin training, static attribute integration, multi-day forecasting capabilities, autoregressive data integration, and cross-regional transfer learning—work effectively together for real-world water resources management applications.

I conduct this research through two experiments applied to 77 mountainous catchments in Central Asia. The first experiment establishes the baseline effectiveness of transfer learning through regional knowledge transfer between neighboring countries (Kyrgyzstan and Tajikistan). The second experiment scales to global transfer learning, comparing strategies that prioritise data volume versus hydrological relevance when selecting source catchments from multiple continents. I evaluate four deep learning architectures, spanning purpose-built hydrological models (EA-LSTM) and state-of-the-art general time series forecast-

ing models (TFT, TSMixer, TiDE), across forecast horizons ranging 1, 5 and 10 days.

This thesis makes three primary contributions to the field. First, I provide the first systematic comparison of transfer learning effectiveness across different deep learning architectures for operational streamflow forecasting. Second, I develop and evaluate a framework for selecting source catchments in global transfer learning, directly comparing volume-based and similarity-based selection strategies and introducing technical methods for handling larger-than-RAM datasets. Third, to my knowledge, this represents the first application of transfer learning from the Caravan dataset to catchments outside the original dataset, extending beyond the single-day, non-autoregressive fine-tuning approaches demonstrated in recent work (Ryd and Nearing, 2025) to multi-day operational streamflow forecasting.

2 Study Area

2.1 Hydrology of Semi-Arid Central Asia

Central Asia (CA), encompassing Kazakhstan, Kyrgyzstan, Tajikistan, Turkmenistan, and Uzbekistan, is characterized by significant hydrological contrasts. The landscape consists of extensive arid plains interrupted by mountain ranges. From north to south, these include the Tien Shan ranges primarily in Kyrgyzstan, the Gissar-Alay ranges along the Kyrgyzstan-Tajikistan border, the Pamirs in eastern Tajikistan, and the Hindu Kush extending into Afghanistan. These mountains function as the region's *water towers* (Immerzeel et al., 2020), constituting a primary zone of runoff formation where precipitation accumulates as snow and ice before supplying major river systems such as the Syr Darya and the Amu Darya.

The region's hydrology can be differentiated into two distinct zones: the mountainous areas (predominantly in Kyrgyzstan and Tajikistan) serve as production zones where water resources originate, while the downstream plains (primarily in Kazakhstan, Turkmenistan, and Uzbekistan) function as consumption zones where water is utilized for irrigation agriculture (Siegfried et al., 2024). This geographic distribution creates a natural upstream-downstream relationship that influences water resource management challenges across the region, particularly considering the arid climate that dominates much of the territory (Bernauer and Siegfried, 2012).

2.2 Target Catchments

As part of recent efforts to modernize hydrological monitoring and forecasting in Central Asia, the Smart & Precise Prognostic Hydrology for Innovative Risk Management and Resource Use Efficiency (SAPPHIRE) project was launched in 2023. Funded by the Swiss Agency for Development and Cooperation (Swiss Agency for Development and Cooperation (SDC), 2022) and implemented by hydrosolutions Ltd. in collaboration with National Hydrometeorological Services (NHMS) in Kyrgyzstan, Tajikistan, Uzbekistan, and Kazakhstan, the project aims to enhance operational hydrological forecasting by integrating digital tools such as the iEasyHydro HF Operational Hydrology Block. This system supports automated data processing, quality control, and the generation of hydrological forecasts (Hydrosolutions Ltd., 2023).

Within the broader scope of SAPPHIRE, my thesis explores whether deep transfer learning techniques could enhance hydrological forecasting in the mountainous regions of CA, particularly in the runoff formation zones of Kyrgyzstan and Tajikistan. My analysis centres on 77 mountainous catchments—61 in Kyrgyzstan and 16 in Tajikistan. Their location is shown in Figure 1, where Kyrgyz basins are depicted in yellow and Tajik basins in purple. Some large catchments encompass smaller ones, which makes them appear darker in the plot.

2.2.1 Climate Patterns of Kyrgyzstan and Tajikistan

The climates of Kyrgyzstan and Tajikistan display distinct seasonal patterns driven by their mountainous topography, with mean catchment elevations of 2917.8 ± 461.1 m.a.s.l. in Kyrgyzstan and 3117.6 ± 625.6 m.a.s.l. in Tajikistan (Table 1). Figure 2 presents boxplots showing the distribution of median monthly values across the catchments in each country. The vertical spread of each boxplot represents the spatial heterogeneity of that variable across the different basins.

Temperature patterns are relatively similar between the two countries, reflecting their high-altitude environments. In cold winters (December-February), median basin temperatures fall below -10°C, particularly in January and December. The boxplots indicate that winter temperature variability across basins is greater in Tajikistan. Summer temperatures peak in July-August, with median basin values reaching

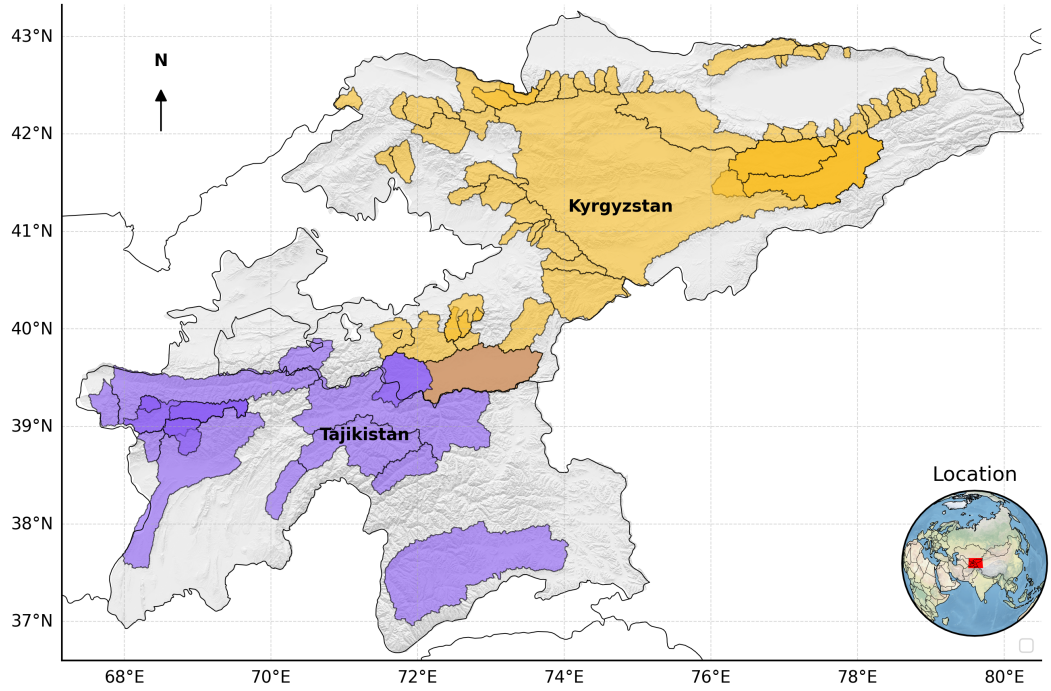

Figure 1: Map showing the location of the 77 mountainous catchments used in my thesis, situated in the runoff formation zones of Central Asia. Catchments in Kyrgyzstan (n=61) are highlighted in yellow, and catchments in Tajikistan (n=16) are highlighted in purple. The inset map shows the regional location. The hillshade layer was derived from a SRTM Global 1 arc-second DEM (Marti et al., 2023). The Map was created in Python using the Cartopy package (Met Office, 2010 - 2015).

10-13°C in both countries.

Precipitation distributions reveal notable differences. Kyrgyzstan's basins show a pronounced summer precipitation maximum, with the highest median values (3-5 mm/day) from May through August and considerable variability across basins during these peak months. Tajikistan's basins exhibit a more evenly distributed precipitation pattern throughout the year, with moderate peaks (1-2 mm/day) during late winter and spring (February-May) and less spatial variability than Kyrgyzstan. Reflecting the higher average elevation, a larger fraction of precipitation falls as snow in the studied Tajik catchments (mean snow fraction 0.6 $\pm$ 0.1) compared to the Kyrgyz catchments (mean snow fraction 0.4 $\pm$ 0.1), as shown in Table 1.

These climate patterns directly influence streamflow regimes. The streamflow boxplots demonstrate that basins in both countries experience peak flows during the summer months (May-August), with maximum median values shy of 4 mm/day for Tajikistan and 2 mm/day for Kyrgyzstan. The peak flows coincide with the melting of high-elevation snowpack and glaciers. Despite the pronounced spatial variability in summer precipitation across Kyrgyz catchments, the variability in summer streamflow among these catchments is not greater than that observed in Tajik basins during the same period.

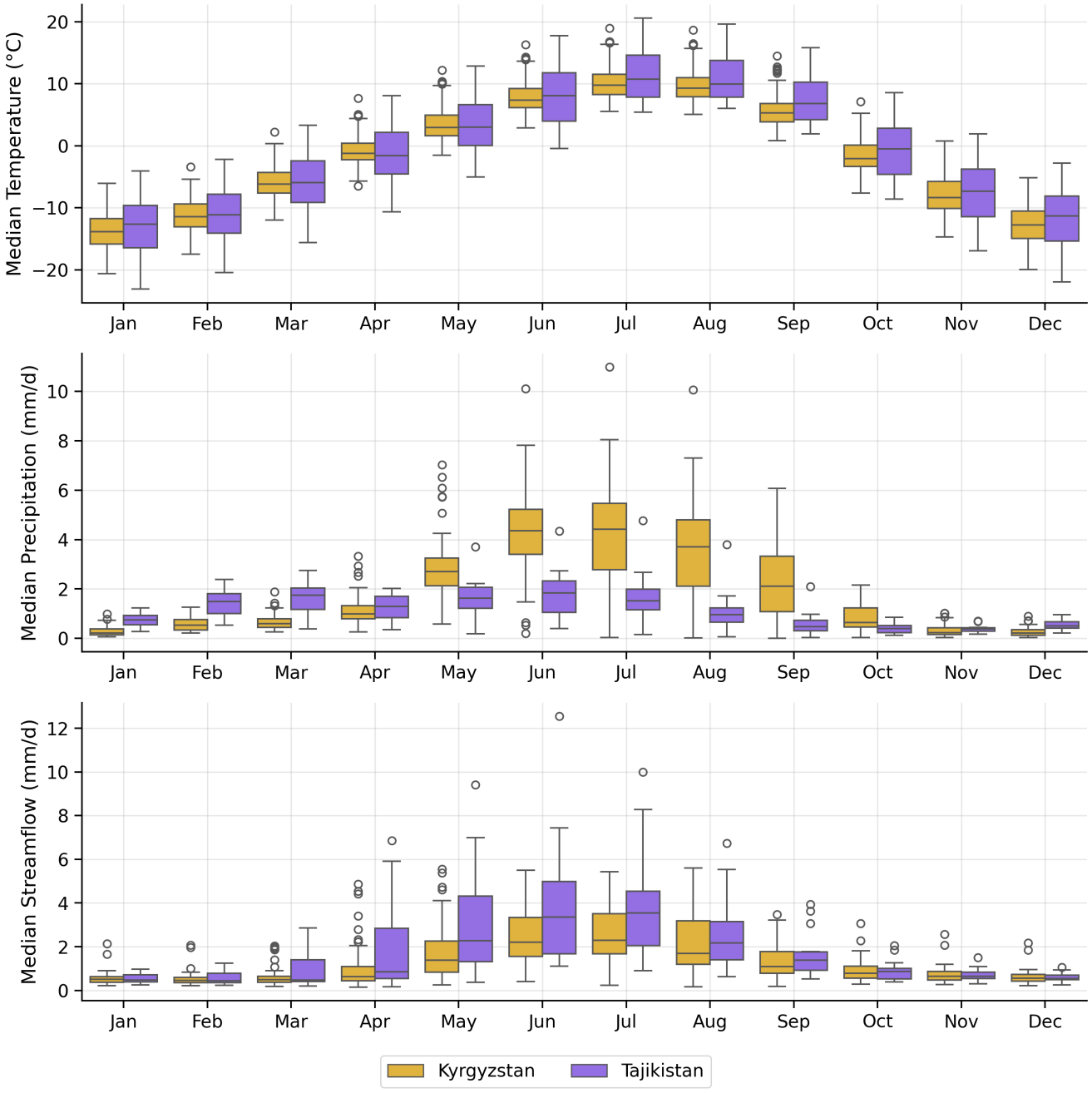

Figure 2: Seasonal climate and streamflow patterns in the target Central Asian catchments. Boxplots illustrate the distribution of median monthly values for temperature, precipitation, and streamflow aggregated across the 61 catchments in Kyrgyzstan (yellow) and 16 in Tajikistan (purple).

Table 1: Mean climate and physiographic attributes by country. For each attribute the mean and standard deviation are reported.

Country	Number of stations	Snow fraction (-)	Area (km^2)	Elevation (m.a.s.l)
Kyrgyzstan	61	0.4 ± 0.1	1868.4 ± 6040.9	2917.8 ± 461.1
Tajikistan	16	0.6 ± 0.1	5029.8 ± 5681.3	3117.6 ± 625.6

3 Data

I explore deep transfer learning for hydrological forecasting using a dataset comprising 16,131 catchments across nine distinct geographic regions. CA (77 catchments, including 16 in Tajikistan and 61 in Kyrgyzstan) represents my target domain, where the objective is to improve forecasting performance through knowledge transfer from source domains including Canada (HYSETS, 12,123 catchments), Brazil (BR) (870 catchments), Central Europe (LamaH, 859 catchments), the United States of America (USA) (671 catchments), Great Britain (GB) (671 catchments), Chile (CL) (502 catchments), Australia (AUS) (222 catchments), and Switzerland (CH) (135 catchments).

3.1 Data Harmonisation Process

Kratzert et al. (2023) developed Caravan, a large-sample hydrology dataset combining streamflow observations from multiple datasets, including CAMELS, CAMELS Australia (CAMELS-AUS), CAMELS Brazil (CAMELS-BR), CAMELS-CL, CAMELS Great Britain (CAMELS-GB), HYSETS, and LamaH Central Europe (LamaH-CE). They harmonised meteorological forcings across all catchments using ERA5-Land reanalysis and standardised catchment attributes with HydroATLAS. This harmonisation ensures data consistency across diverse geographic regions, facilitating comparative studies and the development of hydrological models. To democratise LSH dataset usage, they provided the methodology and Python code for extending the Caravan dataset globally to any catchment with available streamflow observations. I use the Caravan dataset for my research.

Additionally, building upon the methodology and publicly available Python codebase, I extend the Caravan dataset to include the Central Asian and CH catchments, creating a harmonised dataset across all regions. This harmonisation involves extracting ERA5-Land meteorological forcings via Google Earth Engine and computing catchment attributes through HydroATLAS. To ensure reliable derivation of static attributes (Kratzert et al., 2023), I exclude all catchments smaller than $70km^2$. The streamflow values are normalised by catchment area and expressed in units of mm/day.

3.2 Central Asian Data Acquisition and Preprocessing

I use streamflow data from catchments in Kyrgyzstan and Tajikistan, made available to hydrosolutions Ltd. through formal Memorandum of Understandings (MoUs) with the respective NHMSs. These agreements grant access to daily operational streamflow data for developing and testing forecasting models. The dataset consists of manually digitised versions of historical staff gauge readings, typically recorded twice daily at 08:00 and 20:00. hydrosolutions Ltd. performed a manual cleaning procedure to remove outliers and exclude catchments subject to strong anthropogenic influence. Only basins with a pronounced seasonal signal indicative of snowmelt-driven mountainous hydrology were retained. I use catchment boundaries from the publicly available CA-discharge dataset (Marti et al., 2023). These were delineated using WhiteboxTools v2.0.0, based on gauge locations manually positioned in a Geographic

Information System (GIS) and the SRTM Global 1 arc-second DEM.

Due to the sensitive nature of water data in Central Asia, the dataset is not publicly available. Data collection is highly labour-intensive and regarded as the intellectual property of the Hydromets. As stipulated in the MoUs with these institutions, data sharing is explicitly prohibited.

3.3 Streamflow Records

The streamflow datasets I use vary significantly in their number of catchments and temporal coverage, as summarised in Table 2. The source domains provide extensive data, including long historical records from regions such as Chile and Australia, as well as a large number of catchments from HYSETS. In contrast, the target domain of Central Asia is characterised by more recent records. While the table provides the overall date ranges, Figure 3 illustrates the dynamics of this availability by showing the total number of basins with valid streamflow data over time.

Table 2: Overview of streamflow datasets used in this study. The table details the geographic region, original dataset name, number of catchments, and the temporal coverage of the streamflow records.

Region	Dataset Name	Catchments	Temporal Coverage
Canada	HYSETS	12,123	Jan 1951 – Dec 2018
Brazil	CAMELS-BR	870	Jan 1980 – Dec 2018
Central Europe	LamaH-CE	859	Jan 1981 – Jan 2018
USA	CAMELS-US	671	Jan 1980 – Dec 2014
Great Britain	CAMELS-GB	671	Oct 1970 – Sep 2015
Chile	CAMELS-CL	502	Jan 1951 – Jun 2020
Australia	CAMELS-AUS	222	Jan 1951 – Dec 2014
Switzerland	*This study*	135	Jan 1981 – Dec 2020
Central Asia	*This study*	77	Jan 2000 – May 2024

3.4 Meteorological Forcing

I use meteorological forcings from ERA5-Land, a global reanalysis dataset produced by the European Centre for Medium-Range Weather Forecasts (ECMWF) that provides hourly land-surface variables at 9 km resolution (Muñoz-Sabater et al., 2021). I rely on the ERA5-Land data already included in the Caravan dataset for the Canadian, Brazilian, Central European,US, Great British, Chilean, and Australian catchments. For the Swiss and Central Asian catchments, I generate ERA5-Land forcings independently using Google Earth Engine, following the method described by Kratzert et al. (2023). The processing pipeline involves downloading spatially averaged hourly data for each catchment, disaggregating accumulated variables (such as precipitation), converting all variables into hydrologically meaningful units, and shifting timestamps from Coordinated Universal Time (UTC) to the local time zone of each gauge station. I then aggregate the data to daily resolution. Precipitation and potential evaporation are computed as daily sums, while temperature, radiation, pressure, and wind are summarised as daily means, minima, and maxima. Table 3 shows the meteorological forcing features used as model inputs.

Clerc-Schwarzenbach et al. (2024) demonstrated that ERA5-Land reanalysis data generally reduces hydrological model performance compared to the original national datasets used in CAMELS collections, with precipitation differences causing the most significant impact. Despite these limitations, the standardised approach of the Caravan dataset remains valuable for my transfer learning experiments as it ensures a consistent data source across all regions.

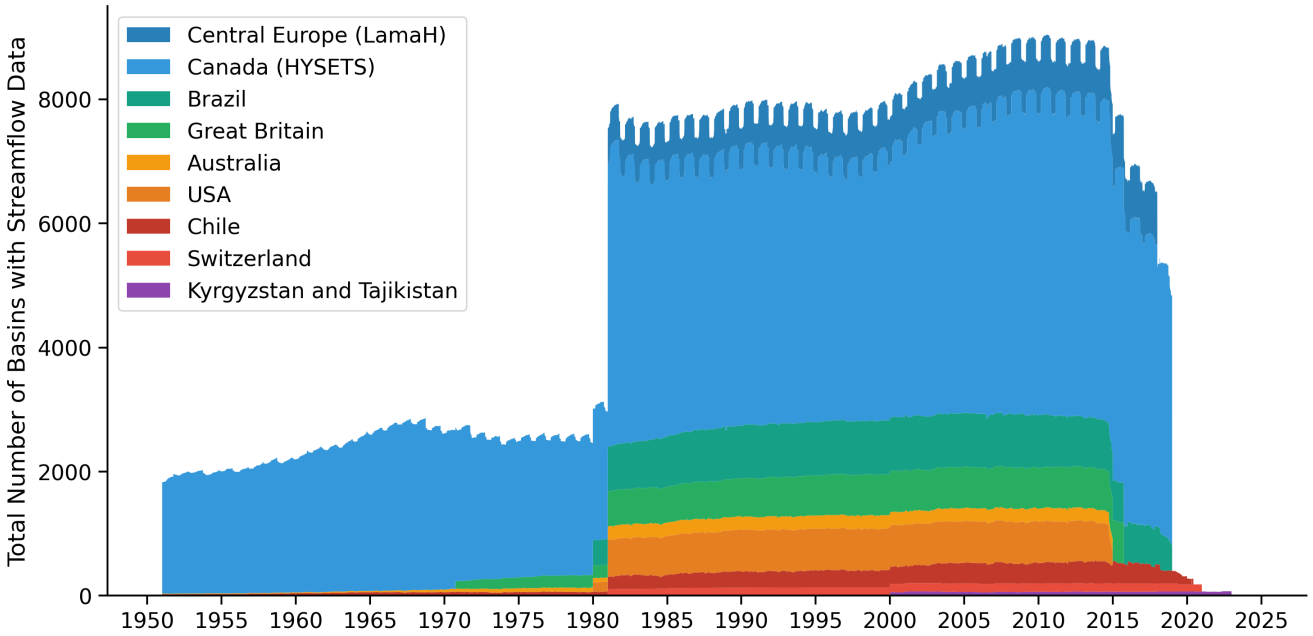

Figure 3: Total number of catchments with streamflow data through time. The plot shows data availability across multiple regions: Central Europe (LamaH), Canada (HYSETS), Brazil, Great Britain, Australia, USA, Chile, Switzerland, and Central Asia (Kyrgyzstan and Tajikistan).

Table 3: Meteorological forcing features used as model inputs.

Variable	Unit
Snow depth water equivalent (mean)	mm
Surface net solar radiation (mean)	W m^{-2}
Surface net thermal radiation (mean)	W m^{-2}
Potential evaporation (sum, ERA5-Land)	mm day^{-1}
Potential evaporation (sum, FAO Penman-Monteith)	mm day^{-1}
Temperature at 2m (min, max, mean)	°C
Total precipitation (sum)	mm day^{-1}

3.5 Catchment Attributes

Two sets of static catchment attributes are derived for each catchment across both source and target domains:

- **HydroATLAS attributes:** These were computed through the spatial intersection between catchment boundaries and the HydroATLAS dataset (Linke et al., 2019). For each catchment, HydroATLAS polygons (level 12, highest resolution) are intersected with the catchment boundary, and attributes are aggregated using area-weighted averaging. Discrete class variables (e.g., land cover and climate zones) are assigned using area-weighted majority voting. The attributes span multiple categories, including hydrology, physiography, climate, soils and geology, land cover, and anthropogenic influences. For a detailed description of each attribute, I refer the reader to catchmentATLAS Attributes (version 1.0)[1].
- **Climate indices:** Additional climate indices are calculated from the ERA5-Land time series, including aridity index, fraction of snow precipitation, moisture index, seasonality, and metrics

[1] https://data.hydrosheds.org/file/technical-documentation/catchmentATLAS_Catalog_v10.pdf

describing precipitation patterns (frequency and duration of high/low precipitation events) (Knoben et al., 2018; Kratzert et al., 2019c).

Table 4 shows the static catchment attributes used as model inputs. These correspond to the ten most important features listed in Kratzert et al. (2019c) (Table 4) that have an equivalent in the Caravan dataset.

Table 4: Static catchment attributes used as model inputs.

Description	Unit
Mean daily precipitation	mm day^{-1}
Average duration of high precipitation events	days
Frequency of high precipitation days	days
Aridity index (PET/P) (ERA5-Land and FAO Pennman-Monteith)	-
Fraction of precipitation falling as snow	-
Mean terrain slope	$^\circ$ (x10)
Catchment area	km^2
Mean elevation	m a.s.l.
Clay fraction in soil	%

4 Methods

4.1 Data Cleaning and Splitting

I apply data quality checks at the basin level to ensure sufficient data for model training. For target domain basins, I require that the designated training portion of the data contains at least 5 years of non-null streamflow observations. For source domain basins used in transfer learning experiments, I require at least 10 years of non-null streamflow observations in the training portion. Basins failing these criteria are discarded. For the remaining basins, gaps of missing data up to five consecutive days are imputed using forward filling.

Following the quality check, I split the data for each basin chronologically. Based on the temporal sequence of available non-null streamflow records, the data is split into a training set (the first 50%), a validation set (the next 25%), and a test set (the final 25%). This threefold split is best practice in time series forecasting with deep learning (Goodfellow et al., 2016). This per-basin splitting approach ensures that data from all basins contributes to model training, allowing the model to learn from the full spatial diversity of the dataset. An alternative approach would be to split data based on calendar dates across all basins simultaneously—for example, using all data from all basins from 2000 to 2015 for training, 2016 to 2020 for validation, and 2021 onwards for testing. However, I do not implement this date-based approach as it could result in some basins contributing only to training or only to testing, potentially reducing the spatial representativeness of each dataset partition.

The chronological splits serve distinct purposes: the training set is for optimising model parameters, the validation set is for monitoring overfitting, and the test set is reserved for a final, unbiased evaluation of model performance on unseen data. All performance metrics reported in this thesis are calculated on this test set.

4.2 Data Preprocessing

4.2.1 In-Model Normalisation: Reversible Instance Normalisation

Time series data often exhibit distribution shifts, where statistical properties such as mean and variance change between training and test periods. This phenomenon is relevant to hydrological forecasting, where long-term trends lead to a shift in the data distribution. Such distribution shifts lead to performance degradation when models encounter test data with statistical properties that differ from those of the training data.

Reversible Instance Normalization (RevIN) (Kim et al., 2021) addresses this challenge through a symmetric normalisation-denormalisation structure. The technique removes instance-specific statistics during model input, allowing the model to focus on learning the underlying patterns, then restores these statistics at the output to return predictions to the original scale.

For an input sequence $\mathbf{x}^{(i)} \in \mathbb{R}^{K \times T_x}$ with K variables and length T_x, RevIN first computes instance-specific statistics:

$$\mu_k^{(i)} = \frac{1}{T_x} \sum_{t=1}^{T_x} x_{k,t}^{(i)} \quad \text{and} \quad \sigma_k^{(i)} = \sqrt{\frac{1}{T_x} \sum_{t=1}^{T_x} \left(x_{k,t}^{(i)} - \mu_k^{(i)} \right)^2}$$

The normalisation step transforms the input as:

$$\hat{x}_{k,t}^{(i)} = \gamma_k \cdot \frac{x_{k,t}^{(i)} - \mu_k^{(i)}}{\sigma_k^{(i)} + \epsilon} + \beta_k$$

where $\gamma, \beta \in \mathbb{R}^K$ are learnable affine parameters optimized during model training and ϵ is a small constant for numerical stability. After the model produces predictions $\tilde{y}^{(i)}$, the denormalisation step restores the original scale:

$$\hat{y}_{k,t}^{(i)} = \frac{\tilde{y}_{k,t}^{(i)} - \beta_k}{\gamma_k} \cdot \sigma_k^{(i)} + \mu_k^{(i)}$$

I apply RevIN to the target streamflow feature within the input sequence and to the model's output predictions, treating the streamflow data with an encoder-decoder normalisation structure while preserving the original scale of meteorological forcings.

4.2.2 Offline Data Preprocessing Techniques

This subsection describes the data transformations that I apply before passing data to the neural network model. These offline preprocessing steps complement the in-model RevIN normalisation. I use three techniques to preprocess the time series data:

Z-Score Normalisation: This standardises features to have a mean of 0 and a standard deviation of 1:

$$\hat{x} = \frac{x - \mu}{\sigma}$$

I implement two variants: a **per-catchment** approach where normalisation statistics are computed independently for each catchment's time series, and a **global** approach where statistics are computed across all time steps and catchments in the entire training dataset.

Log Transformation: I apply a logarithmic transformation defined as:

$$q' = \log(q + 1)$$

This transformation is applied only to the streamflow data q. The "+1" ensures the argument is always positive, preventing mathematical errors for zero values.

Yeo-Johnson Power Transformation: To transform data towards a more Gaussian distribution, I use the Yeo-Johnson transformation (Weisberg, 2001), which extends the Box-Cox transform to handle negative values:

$$\psi(\lambda, x) = \begin{cases} \frac{(x+1)^\lambda - 1}{\lambda} & \text{if } \lambda \neq 0, x \geq 0 \\ \log(x+1) & \text{if } \lambda = 0, x \geq 0 \\ -\frac{(1-x)^{2-\lambda} - 1}{2-\lambda} & \text{if } \lambda \neq 2, x < 0 \\ -\log(1-x) & \text{if } \lambda = 2, x < 0 \end{cases}$$

where the parameter λ is estimated from the data to optimise normality.

Static Attribute Normalisation Across all experiments, I preprocess static catchment attributes using global Z-score normalisation. Given a set of static features $\mathbf{s} \in \mathbb{R}^{N \times M}$ for N catchments and M attributes, each attribute m for catchment n is transformed as:

$$\hat{s}_{n,m} = \frac{s_{n,m} - \mu_m}{\sigma_m}$$

where $s_{n,m}$ is the value of attribute m for catchment n, and the mean μ_m and standard deviation σ_m are computed across all catchments in the training set.

Experimental Preprocessing Pipelines The following subsections outline the specific combination of preprocessing techniques used for each experiment.

Experiment 1: Regional Transfer Learning For this experiment, I preprocess meteorological forcings using global Z-score normalisation, where statistics are computed across all catchments and time steps in the training set. I preprocess the target streamflow values using a two-stage approach: first applying a log transformation $\log(q + 1)$, then per-catchment Z-score normalisation. The model architecture incorporates the RevIN module.

Experiment 2, Phase 1: In-Memory Global Transfer Learning The offline preprocessing pipeline is identical to Experiment 1: global Z-score normalisation for forcings and log transformation, plus per-catchment Z-score normalisation for streamflow. I use RevIN in this experiment.

Experiment 2, Phase 2: Larger-than-RAM Global Transfer Learning This experiment tests a different preprocessing strategy designed to address catastrophic forgetting. Catastrophic forgetting occurs when a network forgets previous training when learning from new data, typically caused by distributional shifts between datasets (Kirkpatrick et al., 2017). Since this experiment processes data in chunks that may have different statistical properties, I implement a strategy to normalise all chunks to similar distributions. Both meteorological forcings and target streamflow undergo a two-step, per-catchment process: first, per-catchment Z-score normalisation, then Yeo-Johnson power transformation. The goal is to transform each chunk into an approximately normal distribution, theoretically reducing the distributional shift that causes catastrophic forgetting. As I demonstrate in the results section, this approach yields reduced model performance compared to the preprocessing pipeline employed in the previous experiments. I do not use RevIN in this experiment.

4.3 Model Training

4.3.1 Training Data Structure

Training Examples, Batches, and Epochs To train the forecasting models, I convert the time-series data into discrete training instances, or **examples**. For computational efficiency, these examples are processed in groups called **batches**.

A single example $e_{i,t}$, for catchment i and anchored at forecast-issue day t, is defined as:

$$e_{i,t} = \big(\underbrace{X_{i,t-L+1,\ldots,t}}_{\text{Past Inputs}}, \underbrace{S_i}_{\text{Static Attributes}}, \underbrace{F_{i,t+1,\ldots,t+H}}_{\text{Meteorological Forecast}}, \underbrace{Y_{i,t+1,\ldots,t+H}}_{\text{Target}}\big)$$

where L is the **lookback window length** (number of past days used as input) and H is the **forecast horizon** ($H = 10$ days). The components are:

- $X_{i,t-L+1,\ldots,t}$: The most recent L days of streamflow observations and meteorological data, ending on day t.
- S_i: Time-invariant catchment attributes.
- $F_{i,t+1,\ldots,t+H}$: Meteorological forecasts for the next H days.
- $Y_{i,t+1,\ldots,t+H}$: Observed streamflows for the next H days, which serve as the prediction target.

During training, examples are grouped into a batch $\mathcal{B}$ of size B. A batch can contain examples from different catchments and time points, allowing the model to learn from diverse hydrological conditions simultaneously. One complete pass over the entire training dataset is called an **epoch**. Training typically involves many epochs, with the model's parameters updated after processing each batch.

Examples are generated using a 1-day sliding window across each catchment's time series. For every day t with at least L preceding days and H subsequent days of data, an example is created.

Input Data Structure For a batch of B examples (daily resolution), the batched tensors have the following shapes:

- X (dynamic historical inputs): $[B, L, 10]$ — streamflow observations concatenated with nine meteorological variables over the lookback window of length L.
- S (static attributes): $[B, 10]$ — time-invariant catchment attributes.
- F (future forcings): $[B, H, 9]$ — forecasted meteorological variables for the next H=10 days.
- Y (target streamflow): $[B, H]$ — observed streamflow for each day in the forecast horizon, used only for training loss.

In this batched notation, I omit the catchment and time indices for clarity.

To include *autoregressive (lagged-target)* information, streamflow observations are incorporated as the leading feature in X. For operational evaluation, I assume perfect meteorological forecasts by using ERA5-Land reanalysis data for the future period (F). While this represents an idealised scenario, it isolates hydrological modelling performance from meteorological forecast uncertainty, allowing direct assessment of the transfer learning benefits for streamflow prediction.

The models employ *direct multi-step* prediction: given the historical dynamic forcing features (X), static attributes (S), and future forcings (F), they predict the entire H=10-day streamflow sequence (Y) simultaneously in a single forward pass (i.e., no recursive use of predicted flows). The optimal lookback window length L varies between 30 and 365 days and is determined through hyperparameter optimisation (see subsubsection 4.3.3).

4.3.2 Training Process and Optimisation

Deep learning models learn to predict streamflow through an iterative optimisation process that systematically adjusts hundreds of thousands of parameters to minimise prediction errors. This process enables the models to automatically discover non-linear patterns in the data without explicit programming of hydrological processes.

The Training Loop Each training iteration processes a batch of examples (X_i, S_i, F_i) through three stages:

1. **Forward Pass** – The model receives (X_i, S_i, F_i), applies RevIN normalisation (in experiments 1 and 2, Phase 1) to the target streamflow component, and propagates the inputs through multiple layers of learned transformations to produce streamflow predictions. Each layer applies trainable weights and biases to transform the data.

2. **Loss Computation** – The model's predictions $\tilde{y}_{b,t}$ are compared to observed streamflow values $y_{b,t}$ using the Mean Squared Error (MSE). Both $y_{b,t}$ and $\tilde{y}_{b,t}$ are in the preprocessed space (e.g., after log transformation and normalisation), not the original streamflow units (mm/day). For batch size B and forecast horizon T, the loss is:

$$\mathcal{L} = \frac{1}{B \times T} \sum_{b=1}^{B} \sum_{t=1}^{T} (y_{b,t} - \tilde{y}_{b,t})^2$$

3. **Backpropagation** – Gradients of the loss with respect to each parameter are computed via the chain rule, propagating errors backward through the network. These gradients indicate the direction and magnitude of parameter updates needed to reduce prediction error.

Parameter Optimisation with Adam I update parameters using the Adam (Adaptive Moment Estimation) optimiser (Kingma and Ba, 2014), which maintains individual adaptive learning rates for each parameter. Adam combines:

- **Momentum** – Tracks an exponentially decaying average of past gradients (first moment), helping traverse flat regions and avoid shallow minima.
- **Adaptive Learning Rates** – Tracks the exponentially decaying average of squared gradients (second moment), scaling each parameter's learning rate based on the historical magnitude of its gradients.

Given learning rate γ, bias-corrected first and second moment estimates $\hat{m}_t$ and $\hat{v}_t$, and stability constant ϵ, the parameter update rule for parameters θ is:

$$\theta_{t+1} = \theta_t - \gamma \cdot \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon} \tag{1}$$

Adaptive Learning Rate Scheduling and Early Stopping While Adam adapts learning rates per parameter, I also adjust the global learning rate γ using a `ReduceLROnPlateau` scheduler. This monitors validation loss after each epoch and halves the learning rate if the loss fails to improve for five consecutive epochs. The scheduler coordinates with early stopping: training terminates if validation loss fails to improve for 10 consecutive epochs, regardless of learning rate adjustments. Models typically train for 50-200 epochs before early stopping or convergence, depending on dataset size and architecture.

Computational Efficiency Through Batching Processing examples in batches rather than individually both improves computational efficiency and stabilises training. Modern Graphics Processing Units (GPUs) excel at parallel matrix operations, making batch processing significantly faster than sequential execution. Moreover, averaging gradients over multiple examples reduces noise in the updates, preventing oscillations near minima and enabling smoother convergence. In this work, I use a batch size of 2,048 sequences.

Training on Datasets Larger than RAM The Caravan and CAMELS-CH datasets exceed system memory, so I implement a *chunking strategy*. All basins are randomly partitioned into chunks small enough to fit in Random Access Memory (RAM) (1,500 in my experiments; see subsubsection 4.3.5). Training proceeds by loading one chunk at a time and processing all sequences from those basins as a single chunk-epoch. Once all chunks are processed (completing one full epoch over the entire dataset), the basins are reshuffled into new random partitions.

Validation loss is computed on a fixed set of 4,000 randomly selected basins throughout training. If fewer than 4,000 basins are available, the full validation set is used.

4.3.3 Hyperparameter Tuning

Model training optimises learnable parameters through the Adam optimiser. Hyperparameters are configuration settings that remain fixed during training but significantly impact model performance. Examples of hyperparameters include input length (determining the historical lookback window size), learning rate, and hidden layer dimensions. To find the optimal hyperparameters for each model, I employ Optuna (Akiba et al., 2019), an automated hyperparameter optimisation framework that efficiently explores the hyperparameter space.

Optuna utilises the Tree-structured Parzen Estimator (TPE) (Akiba et al., 2019; Bergstra et al., 2011), a Bayesian optimisation algorithm that maintains a history of trial results to suggest promising hyperparameter combinations intelligently. The framework employs two key strategies: intelligent sampling, which concentrates on regions showing better performance, and pruning, which automatically terminates unpromising trials early to reduce computational overhead. My implementation defines model-specific search spaces in separate Python modules, specifying parameter types, ranges, and sampling distributions. Each optimisation trial trains a complete model on the train split with suggested hyperparameters. The validation loss is returned as the objective. Upon completion, Optuna identifies the best-performing configuration and generates visualisation plots showing optimisation history and parameter importance.

In Table D3 in the Appendix, I show the search space for all tuned hyperparameters and the fixed values for all non-tuned hyperparameters used for each model. I tune the hyperparameters exclusively on the target domain data. Experiments 1 and 2, Phase 1 (described in section 5) share the same hyperparameters for Tajikistan, while Experiment 2, Phase 1, also tunes separate hyperparameters for Kyrgyzstan (Table D4). Experiment 2, Phase 2, employs a distinct set of hyperparameters for both countries (Table D5).

4.3.4 Transfer Learning

In my transfer learning experiments, I adopt a two-stage methodology comprising pre-training and fine-tuning. This approach is designed to transfer knowledge from a data-rich source domain to a target domain. The stages are:

1. **Phase 1: Pre-training (or weight initialisation).** The model is first trained on the source domain using the standard procedure: forward pass, loss computation, backpropagation, and parameter update. This phase aims to learn robust, generalised hydrological features from a diverse dataset. The final output of this stage is a model with optimised weights that encapsulate this foundational knowledge.

2. **Phase 2: Fine-tuning.** The pre-trained model's weights are used to initialise a new training session on the target catchments. For this phase, I reduce the learning rate by a factor of 25 from its initial value. This conservative update strategy aims to prevent *catastrophic forgetting* (Kirkpatrick et al., 2017) while enabling the model to adapt to the unique characteristics of the target domain. The fine-tuning process otherwise uses the same training configuration as the initial pre-training, with the reduced learning rate being the sole modification.

4.3.5 Computational Resources

Training deep learning models is computationally intensive, requiring extensive matrix operations and gradient computations across large datasets. It is standard practice to accelerate these computations using GPUs, which offer parallel processing capabilities. To access the necessary computational resources, I utilise RunPod. This cloud computing platform provides on-demand access to high-performance GPUs for Artificial Intelligence (AI) and machine learning workloads, eliminating the need for significant hardware investments (RunPod, 2025).

Specifically, I trained all models on NVIDIA RTX 4090 GPUs with 24 GB of Video Random Access Memory (VRAM). The computational environment provided 61 GB of system RAM and 16 Virtual Central Processing Units (vCPUs). The system RAM capacity determines the maximum dataset size that can be loaded into memory.

4.4 Model Evaluation

This section describes the evaluation framework. I implement the framework to assess model performance in an operational context.

4.4.1 Evaluation Framework

Multi-Horizon Performance Assessment I evaluate all models on the held-out test set using a rolling forecast approach. For each day in the test period, the model generates a complete 10-day forecast sequence from a single prediction. For each forecast horizon—from 1-day ahead to 10-day ahead—the corresponding predictions are isolated and compared with the observed values. Performance metrics are then calculated independently for each horizon. This approach enables me to assess how model performance deteriorates with increasing forecast horizon, without requiring the training of separate models for each lead time.

I evaluate the performance of all models over the growing season (March through October) when snowmelt-driven streamflow predictions are most relevant for water resources management.

4.4.2 Performance Metrics

Model performance is quantified using four Goodness of Fit (GoF) metrics:

Hydrological Metrics

- NSE: Measures the predictive skill relative to the mean of observations, with values ranging from $-\infty$ to 1, where 1 indicates perfect prediction.
- Kling-Gupta Efficiency (KGE): Decomposes model performance into correlation, bias, and variability components.

Machine Learning Metrics

- Root Mean Square Error (RMSE): Quantifies the average magnitude of prediction errors, giving higher weight to large errors.
- Mean Absolute Error (MAE): Provides the average absolute difference between predictions and observations, treating all errors equally.

The mathematical formulations of these metrics are provided in Appendix A.

4.4.3 Baseline Comparison

All deep learning models are benchmarked against a persistence baseline (dummy model) that naively repeats the last observed streamflow value for the entire forecast horizon. This baseline provides a reference for the minimum acceptable performance level—any sophisticated model should outperform this simple approach to justify its additional complexity.

4.5 Human Influence Index

I developed a composite Human Influence Index (HII) to quantify the degree of anthropogenic influence within each catchment. The index is derived from eight static attributes sourced from the HydroATLAS dataset: population density, urban extent, road density, nighttime light intensity, Gross Domestic Product (GDP), Human Development Index, degree of water regulation, and reservoir volume (see Table 5 for descriptions).

The calculation process begins with data preprocessing. To ensure comparability across different units and scales, each attribute is first normalised to a [0, 1] range using min-max scaling. Subsequently, weights are assigned to each attribute based on my judgment of their relative impact on hydrological regimes. For instance, direct infrastructural indicators, such as reservoir volume and degree of regulation, receive higher weights. These raw weights are then normalised by dividing each weight by the sum of all raw weights, ensuring the final values sum to one.

The HII for each catchment is computed as the weighted sum of the normalised attributes. To produce a final, interpretable score, the resulting composite index is scaled by its maximum value, bounding all HII scores between 0 and 1. It is important to note that because the normalisation and scaling are dependent on the specific catchments in the analysis, the HII scores are relative and only comparable within the current dataset. The index must be recalculated if the dataset is modified.

To facilitate interpretation, these continuous HII scores are used to classify catchments into three categories of human influence: 'Low' (below the 30th percentile), 'Medium' (between the 30th and 75th

percentiles), and 'High' (above the 75th percentile). These thresholds were selected to distinguish between catchments with minimal, moderate, and significant signs of human modification, which I verified by comparing classification results against catchment hydrographs.

Table 5: Anthropogenic attributes used for calculating the Human Influence Index (HII), with assigned attribute importance weights. Source: BasinATLAS Attributes (version 1.0).

Description (HydroATLAS name)	Unit	Importance Weight
Population density (ppd_pk_sav)	people per km^2	1
Urban extent (urb_pc_sse)	% cover	1
Road density (rdd_mk_sav)	km/km^2	1
Nighttime lights intensity (nli_ix_sav)	index value (x100)	1
Gross Domestic Product (gdp_ud_sav)	USD ($)	3
Human Development Index (hdi_ix_sav)	index value (x1000)	1
Degree of water regulation (dor_pc_pva)	percent (x10)	5
Reservoir volume (rev_mc_usu)	million cubic meters	5

4.6 Catchment Similarity

I cluster catchments by streamflow regimes using the shape-based time-series approach from Yang and Olivera (2023), which identifies hydrologically similar catchments based on hydrograph regimes rather than absolute flows. I hypothesise that catchments exhibiting similar hydrograph patterns share similar underlying hydrological processes and behaviours, making this classification particularly valuable for streamflow forecasting. I apply this clustering to all 16,053 catchments in the Caravan dataset (including my extension to CAMELS-CH, see subsection 3.1). I deliberately exclude CA catchments as I aim to develop a framework to identify similar source catchments applicable to both gauged and ungauged target regions. After clustering the gauged catchments, I train a random forest classifier to predict cluster membership using static catchment attributes. This classifier assigns CA catchments to clusters, allowing me to identify hydrologically similar source catchments for each target catchment. This section provides a detailed description of the clustering algorithm and the random forest model.

4.6.1 Hydrograph-based Catchment Clustering

This section is structured following the three fundamental components of clustering:

- A suitable distance measure to quantify the similarity between streamflow time series.
- A clustering algorithm to group catchments exhibiting similar hydrological behaviours.
- A method to determine the optimal number of clusters to ensure meaningful hydrological classification.

Similarity Measure

I prepare the data by aggregating daily streamflow values into weekly means for each catchment over the hydrological year. I adjust the water year definition based on hemisphere—October to September for the Northern Hemisphere and April to March for the Southern Hemisphere. This results in a hydrograph of 52 weekly values per catchment. To focus on the shape of these hydrographs rather than absolute magnitudes, I apply Z-score normalisation:

$$Z_{ij} = \frac{Q_{ij} - \mu_i}{\sigma_i}$$

Where Z_{ij} is the standardised flow at week j for catchment i, Q_{ij} is the weekly flow, and μ_i, σ_i are the mean and standard deviation of the annual streamflow for catchment i.

I implement Dynamic Time Warping (DTW) (Sakoe and Chiba, 1978) as the distance measure to measure the similarity between these weekly standardised hydrographs. DTW quantifies similarity between temporal sequences while accommodating variations in timing. Rather than comparing flow values at identical time indices, DTW identifies optimal alignments between hydrographs by mapping similar patterns that may occur with slight temporal offsets. Following Yang and Olivera (2023), I constrain the warping window, which limits the allowable shift in time when aligning hydrographs, using a relative window size of 2/52, which corresponds to approximately two weeks. This ensures meaningful hydrological comparisons while avoiding unrealistic temporal distortions.

Clustering Algorithm
After establishing the DTW as a suitable distance measure, I implement a partitional clustering approach using the `tslearn` library's implementation of `TimeSeriesKMeans`. The algorithm employs an iterative optimisation process to group catchments into non-overlapping clusters. The clustering process begins with initialisation using K-means++ to select initial cluster centres that are well-distributed across the feature space. This initialisation strategy helps avoid the poor local optima from random centroid selection (Arthur and Vassilvitskii, 2006). The algorithm assigns each catchment's standardised hydrograph for each iteration to its nearest cluster based on DTW distance. After the assignment, cluster centroids are recalculated using DTW Barycenter Averaging (DBA) (Petitjean et al., 2011). The iterative process continues until convergence occurs when either the cluster assignments stabilise (no catchments change clusters between iterations) or the change in total inertia (sum of squared DTW distances to cluster centroids) falls below a specified tolerance threshold (1×10^{-4}).

I implement parallelised distance calculations to optimise computational efficiency and employ Euclidean upper bound pruning to accelerate the clustering process without compromising accuracy (Silva and Batista, 2016). This pruning technique automatically sets the maximum distance threshold to the Euclidean distance between time series, allowing early termination of DTW calculations when this bound is exceeded. This allows the algorithm to effectively handle the substantial number of catchments in the dataset while maintaining reasonable runtime performance.

Determining the Optimal Number of Clusters
I employ the elbow method to identify the optimal number of clusters by plotting the inertia (within-cluster variance) against cluster counts ranging from 10 to 20. The resulting plot resembles a bent arm; the optimal number of clusters is located at the "elbow," the point where the curve bends, and the rate of inertia decrease slows dramatically.

4.6.2 Random Forest Classification for Cluster Prediction

Once clusters are defined based on streamflow regimes, I use them as training labels in a supervised learning task, enabling the projection of hydrological similarity onto (potentially) ungauged regions using only static attributes. To assign catchments in CA to clusters, I use a Random Forest classifier trained to predict cluster memberships from a set of 17 static catchment attributes, including area, latitude, longitude, and various climatic and land cover features. Table C1 shows the exact static catchment attributes used for cluster prediction in the Random Forest model. The model is trained on complete cases, meaning any catchments with missing attribute data are excluded from the training set.

While Yang and Olivera (2023) applied Random Forests solely for class prediction, I extend their approach by predicting probabilities rather than discrete cluster labels. The probabilistic outputs provide

insights into model confidence, serving as a proxy for hydrological similarity. Specifically, when a target catchment shows similar probabilities of belonging to multiple clusters, I assume it indicates that it shares hydrological characteristics with catchments across these clusters. Model hyperparameters are tuned using `RandomizedSearchCV` from the Python package `scikit-learn` (Pedregosa et al., 2011), exploring a search space that includes the number of estimators (100-300), maximum depth (10-30), and minimum samples for splits and leaves. The model is trained with ten-fold cross-validation, where I divide all source catchments into ten approximately equal groups. In each iteration, nine groups are used to train the model, while the remaining group serves as an independent test set. The classification accuracy is calculated for each test fold and averaged across all iterations to evaluate the model's performance.

4.7 Deep Learning Models

In my research, I evaluate four deep-learning architectures for multi-day streamflow forecasting. The first is an EA-LSTM network, which I adapt specifically for this study's operational context from a model originally designed for ungauged basins. The other three models—the TFT, TSMixer, and the TiDE—are selected because they form the core of the operational forecasting system recently deployed by hydrosolutions Ltd. in Kyrgyzstan and Tajikistan. To ensure full control and customization for my experiments, I implement all models in PyTorch.

4.7.1 Persistence Model (Sanity Check)

In addition to the deep learning models described in this section, I include a simple persistence model (referred to as the Dummy Model in section 6), which repeats the last observed streamflow value for the entire 10-day forecast horizon. This model serves as a sanity check and baseline for evaluating the performance of DL architectures.

4.7.2 Entity-Aware LSTM

As mentioned in the introduction, the EA-LSTM builds upon the standard LSTM architecture by integrating static catchment attributes—such as elevation, soil properties, and climate indices—directly into the model's structure. The EA-LSTM uses these static attributes to modulate the LSTM's input gate, which controls the flow of information into the model's memory cells. This mechanism allows the model to learn hydrological similarities and share learned information across catchments with similar physical characteristics (Kratzert et al., 2019b).

Model Adaptation for Operational Forecasting The original EA-LSTM is designed for same-day streamflow prediction (nowcasting) in ungauged basins, where a single model trained on multiple basins can predict streamflow in locations without local training data. However, my thesis aims to improve operational hydrological forecasting models for gauged catchments, which presents different requirements: (1) the need for multi-day forecasts (10 days ahead) rather than single-day predictions, (2) the availability of meteorological forecast data and (3) the operational context where historical streamflow measurements are available for forecasting. To adapt the EA-LSTM for these operational forecasting needs, I modify the code accompanying the publication by Kratzert et al. (2019b)[2]. I implement two innovations. First, the model architecture is extended to produce multi-day forecasts rather than single-day predictions by modifying the output projection layers. Second, I develop a Bidirectional EA-LSTM (BiEALSTM) inspired by the bidirectional LSTM architecture originally developed for speech recognition applications (Graves et al., 2013).

Bidirectional LSTM Background The bidirectional LSTM, introduced for phoneme classification in speech recognition, processes the same input sequence (e.g., an audio signal) through two separate standard LSTM networks simultaneously: one processes the sequence in the forward direction (from beginning to end), while the other processes the identical sequence in reverse (from end to beginning). In speech recognition tasks, the objective is to convert audio signals into sequences of phonemes (speech sounds). This dual processing enables the model to access both preceding and succeeding contextual information when classifying each phoneme, thereby improving recognition accuracy, as speech sounds depend on their surrounding acoustic context.

BiEALSTM Architecture My BiEALSTM adapts this bidirectional concept for hydrological forecasting while maintaining the core principle of bidirectional processing but redefines the directions in terms

[2] https://github.com/kratzert/ealstm_regional_modeling/blob/master/papercode/ealstm.py

of temporal domains rather than processing order. Instead of processing the same sequence forward and backward through time, the BiEALSTM processes information bidirectionally across the temporal divide between past and future. The first branch processes historical data, including observed meteorological forcings, observed streamflow values, and static catchment attributes up to the present time. The second branch processes information from the future, specifically meteorological forecast data for the 10-day forecast horizon combined with the same static catchment attributes. Both branches generate hidden states that are then concatenated and passed through a final projection layer that produces a sequence of 10 daily streamflow predictions. For simplicity, I will refer to this BiEALSTM architecture as EA-LSTM throughout the remainder of this thesis.

4.7.3 Operationally Deployed Models: TFT, TSMixer, and TiDE

In contrast to the EA-LSTM, which is adapted from a hydrological context, TFT, TSMixer, and TiDE are state-of-the-art architectures developed for general-purpose time series forecasting. A key strength that makes them well-suited for this research is their native ability to process a heterogeneous mix of inputs. Their architectures are designed with built-in mechanisms to handle past time series (e.g., historical streamflow), known future time-varying inputs (e.g., meteorological forecasts), and static covariates (e.g., catchment attributes). This flexibility allows them to be applied to streamflow forecasting with no fundamental architectural modifications. A brief description of these models is given below; for a deeper explanation, please refer to the respective publications.

Temporal Fusion Transformer (TFT) The Temporal Fusion Transformer (Lim et al., 2021) is a model that combines two powerful deep learning approaches: recurrent neural networks (like LSTMs) and attention mechanisms (from Transformer models). Conceptually, TFT processes hydrological data in two complementary ways: it uses LSTM components to capture local, short-term patterns in the meteorological and streamflow sequences, while attention mechanisms identify important relationships across longer periods. The model can automatically determine which input variables (e.g., precipitation, temperature, past streamflow) are most relevant for each prediction and how static catchment attributes should influence the forecasting process. A key feature is TFT's ability to provide insights into which historical periods and input variables contribute most to each forecast, offering valuable interpretability for hydrological understanding. For this study, I modify the original TFT to use mean squared error (MSE) loss and produce single-value predictions for each day rather than the original quantile regression approach that generates prediction intervals.

Time-Series Mixer (TSMixer) TSMixer (Chen et al., 2023) represents a fundamentally different approach, using only simple multi-layer perceptrons (MLPs) without any recurrent or attention components. The model operates through a conceptually straightforward "mixing" process, alternating between analysing patterns across time (time-mixing) and patterns across different input variables (feature-mixing). Time-mixing enables the model to learn temporal relationships in meteorological sequences and streamflow patterns, while feature-mixing allows it to understand how different variables (precipitation, temperature, and catchment attributes) interact with each other. This alternating process continues through multiple layers, gradually building up complex representations of the hydrological system. Despite its simplicity compared to LSTMs or Transformers, TSMixer is remarkably effective in time series forecasting across various domains.

Time-series Dense Encoder (TiDE) TiDE (Das et al., 2023) follows an encoder-decoder architecture using only MLPs. The encoder component processes all available historical information (meteorological observations, past streamflow, catchment characteristics) to create a compressed representation of the current catchment state. The decoder then combines this learned state representation with meteorological forecasts to generate predictions for streamflow. A key feature of TiDE is its "residual connection"

that preserves linear relationships between inputs and outputs—essentially ensuring that simple, direct relationships (such as immediate rainfall-runoff responses) are not lost within the more complex neural network processing.

5 Experiments Description

5.1 Research Objectives

In this thesis, I investigate the effectiveness of transfer learning for improving deep learning-based hydrological forecasting, addressing six key Research Questions (RQs):

1. **RQ1: Transfer Learning Effectiveness** - Does transfer learning improve hydrological forecasting performance compared to training solely on target domain data?
2. **RQ2: Forecast Horizon Impact** - Do transfer learning benefits increase at longer forecast horizons, indicating improved utilisation of meteorological forcing features beyond autoregressive signals?
3. **RQ3: Data Volume vs. Relevance** - Which strategy yields better performance: training on larger volumes of data or training on smaller, more hydrologically relevant datasets?
4. **RQ4: Scalability to Large Datasets** - Can transfer learning benefits scale to very large (larger-than-RAM) datasets, and what technical approaches enable this scalability?
5. **RQ5: Target Domain Data Availability** - How do the benefits of transfer learning change with data availability in the target domain?
6. **RQ6: Architecture-Dependent Transfer Learning** - How do different deep learning architectures respond to transfer learning?

5.2 Experiment 1: Regional Deep Transfer Learning

Objective: Address RQ1, RQ2, and RQ6 by demonstrating transfer learning effectiveness in a controlled regional context and comparing responses across different architectures.

In this experiment, I explore the potential of regional transfer learning to improve the hydrological forecasting capabilities of four deep learning models of varying complexity (EA-LSTM, TFT, TSMixer, and TiDE) by leveraging data from neighbouring countries. Specifically, I compare two training strategies: (1) training a model solely on data from 15 mountainous basins in Tajikistan (target domain) and (2) pre-training the model on a larger dataset comprising 61 mountainous basins from Kyrgyzstan (source domain), followed by fine-tuning on the Tajik data.

Hypotheses: I hypothesise that pre-training on Kyrgyz data will improve forecasting performance in Tajikistan, with greater gains at later horizons. For short-term predictions, performance is influenced by the strong predictive persistence in the autoregressive signal of streamflow observations. As the forecast horizon extends, the influence of this signal diminishes, and the model must instead rely on its learned internal representation of the rainfall-runoff processes to translate meteorological forcing into future streamflow. Pre-training on the larger Kyrgyz dataset provides this more robust internal representation.

Expected Outcomes: This experiment will establish baseline transfer learning effectiveness (RQ1), demonstrate whether benefits increase with forecast horizon (RQ2), and reveal how different architectures respond to transfer learning (RQ6).

5.3 Experiment 2: Global Deep Transfer Learning

Objective: Address RQ3, RQ4, RQ5, and RQ6 by testing data volume vs. relevance trade-offs, developing methods for large-scale transfer learning, evaluating performance across different target domain data availability scenarios, and comparing architectural responses to global transfer learning.

This experiment investigates whether pre-training on global data can improve hydrological forecasting in Central Asia. I structure the experiment into two phases: first, to test the approach with a standard in-memory dataset, and then to develop and evaluate a method for handling larger-than-RAM datasets.

5.3.1 Phase 1: In-Memory Global Transfer Learning

I first test the viability of the approach using a dataset of catchments from the USA, Chile, and Switzerland that fit into the virtual machine's RAM. The success of this phase motivates the development of a method to handle larger datasets.

5.3.2 Phase 2: Larger-than-RAM Global Transfer Learning

To test if performance can be improved with more data, I develop an approach to pre-train models on the whole Caravan and harmonised CAMELS-CH datasets, which exceeds available RAM.

In both phases, I compare two strategies for selecting source catchments and apply them to two different Central Asian target domains:

- **Volume-Based (Low Human Influence):** This strategy prioritizes data quantity. I select catchments with low to moderate human influence as the source catchments. The human influence is quantified by the HII, as described in subsection 4.5.

  Note on CAMELS-CH inclusion: During the preparation of the larger-than-RAM experiments, I inadvertently excluded CH basins from the volume-based selection due to an error in the HII classification process. This resulted in 91 CH basins with low and medium human influence being omitted from the pre-training dataset. However, I did include CH basins in the similarity-based selection approach (see next bullet point). Given time constraints, I did not retrain the models. I address this limitation in section 7.

- **Similarity-Based (Hydrologically Similar):** This strategy prioritizes data relevance. I select catchments with hydrograph shapes similar to those in Central Asia as the source catchments. The methodology for finding similar catchments is described in subsection 4.6.

I apply the models to two target domains to evaluate performance gains in different contexts: a data-limited case involving 15 basins in Tajikistan and a data-rich case utilising 59 basins in Kyrgyzstan. This comparison directly addresses RQ5 by examining how transfer learning benefits vary with target domain data availability.

Hypotheses: I hypothesise that the benefits of global transfer learning will be most pronounced for the data-limited Tajikistan case, as the relative information gain from pre-training is greatest for the smaller target dataset (addressing RQ5). Furthermore, I expect the similarity-based selection strategy to outperform the volume-based one, demonstrating that for the specific task of forecasting streamflow in mountainous Central Asia, the similarity of the training examples is more critical than the total data volume (addressing RQ3). Regarding architectural differences, I anticipate that more complex models (TFT

and EA-LSTM) will benefit more from transfer learning than simpler architectures (TSMixer and TiDE), as their sophisticated internal mechanisms—attention layers, bidirectional processing, and entity-aware components—are better equipped to extract and leverage rich representations from diverse pre-training data (addressing RQ6). This architectural advantage should be particularly evident when scaling to larger datasets, where complex models can more effectively capture and transfer nuanced hydrological patterns that simpler architectures might miss. Finally, I anticipate that leveraging the larger-than-RAM datasets in Phase 2 will yield further improvements over the in-memory datasets from Phase 1, with the most pronounced gains observed in complex architectures (addressing RQ4). This expectation is based on the fundamental principle that deep learning models are data-driven, and therefore more (high quality) data generally leads to better model performance.

Expected Outcomes: This experiment will determine whether data relevance trumps data volume (RQ3), establish the feasibility and effectiveness of large-scale transfer learning in hydrology (RQ4), quantify how transfer learning benefits change with target domain data availability (RQ5), and reveal architectural dependencies in transfer learning effectiveness (RQ6), while also documenting the technical methodology for handling larger-than-RAM datasets.

6 Results

6.1 Human Influence Index Classification

This section presents the results of the classification of the degree of human influence for basins in the Caravan and CAMELS-CH based on the HII. The classification is performed twice: once for Experiment 2, Phase 1, and a second time for Experiment 2, Phase 2.

6.1.1 Experiment 2 Phase 1

Figure B1 shows the distribution of HII value across the Chilean, US and Swiss catchments. The figure shows a bimodal pattern. The primary mode occurs between 0.15 and 0.2, just below the *Low* threshold of 0.17. The secondary mode appears between 0.35 and 0.4, coinciding with the *High* threshold at the point of 0.35. The histogram exhibits a long right tail extending to HII = 1.0. Two catchments from the USA have an HII value above 0.8.

Figure B2 shows the distribution of HII categories across three regions, ordered by increasing proportion of high human influence catchments. **Chile** has 483 catchments in the low and medium categories, corresponding to 95.6% of all catchments. **United States** contains 431 catchments in the low and medium categories, representing 64.2% of all catchments. **Switzerland** shows 50 catchments in the low and medium categories, accounting for 37.0% of all catchments. Switzerland exhibits the highest proportion of highly human-influenced catchments (63.0%), while Chile shows the lowest (4.4%). The United States has no catchments in the low-influence category, and Switzerland has no catchments in this category.

To provide a visual validation of the index, Figure B3 shows daily streamflow time series for three Chilean catchments from 2003 to 2004, representing each HII category. The catchments have HII values of 0.14 (Low), 0.21 (Medium), and 0.44 (High).

6.1.2 Experiment 2 Phase 2

The distribution of HII values across 16,038 catchments in the Caravan dataset is shown in Figure B4. This histogram shows a unimodal pattern. The mode occurs between 0.3 and 0.35, with the highest frequency bin containing over 2,500 catchments. The 30th percentile threshold falls at 0.29, while the 75th percentile threshold occurs at 0.37 just after the mode. The majority of catchments cluster around

HII values of 0.25 to 0.4, with 24 Canadian catchments exhibiting HII values above 0.8.

Figure B4 shows that **Brazil** has 868 catchments in the low and medium categories, corresponding to 99.8% of all catchments. **Chile** contains 487 catchments in the low and medium categories, representing 96.4% of all catchments. **Australia** shows 215 catchments in the low and medium categories, accounting for 96.8% of all catchments. **Central Europe** has 812 catchments in the low and medium categories, representing 94.5% of all catchments. **United States** contains 567 catchments in the low and medium categories, corresponding to 84.5% of all catchments. **Great Britain** shows 538 catchments in the low and medium categories, accounting for 80.2% of all catchments. **Canada** has 8,463 catchments in the low and medium categories, representing 69.6% of all catchments. Canada has the highest proportion of highly human-influenced catchments (30.4%), while Brazil has the lowest (0.2%).

6.2 Catchment Similarity

This section presents the results of the hydrograph-based catchment clustering and random forest classification, which aim to identify catchments in the Caravan and CAMELS-CH datasets that are similar to those in Central Asia.

Clustering: Based on the elbow plot analysis (see Figure C6 in the Appendix, which additionally shows the silhouette score), I determined that 11 clusters are optimal for the hydrograph-based catchment clustering. Figure 4 shows the resulting cluster centroids and sample hydrographs for each cluster.

Random Forest Classification: I trained a random forest classifier to predict cluster membership using the 17 static attributes described in Table C1 across 16,053 catchments from the 11 classes. The model achieved a cross-validation accuracy of 0.709 and a log loss score of 0.961. When applied to the 78 Central Asian basins, 77 basins (98.7%) had either cluster 2 or cluster 7 among their top two predicted clusters. Based on these results, I select all basins from clusters 2 and 7 as similar catchments, yielding a total of 2,929 catchments (1,914 from cluster 2 and 1,015 from cluster 7). As shown in Table C2 in the Appendix, cluster 2 represents high-elevation catchments (1,803.2 m.a.s.l) with high snow fractions (0.4), while cluster 7 represents catchments with the highest annual precipitation (1,244.6 mm/year) and largest average areas (11,962.8 km^2). Figure C7 in the Appendix shows a stacked bar chart with the distribution of basins from each country across all clusters.

6.3 Data Cleaning and Quality Checks

Following the data quality checks described in subsection 4.1, the number of basins meeting the minimum data requirements are as follows: For the target domains, 15 out of 16 Tajik basins (93.8%) and 59 out of 61 Kyrgyz basins (96.7%) pass the quality checks, which require at least 5 years of non-null streamflow observations in the training portion. For the source domains in the in-memory experiments, 161 out of 200 similarity-based basins (80.5%) and 852 out of 1,054 volume-based basins (80.8%) meet the requirement of having at least 10 years of non-null streamflow data. In the larger-than-RAM experiments, 1,850 out of 2,929 similarity-based basins (63.2%) and 6,690 out of 11,950 volume-based basins (56.0%) pass the quality checks.

6.4 Experiment 1: Regional Transfer Learning

This experiment evaluates the effectiveness of deep transfer learning in a regional data sharing scenario by comparing models trained solely on 15 Tajik basins (benchmark) against models pre-trained on 59 basins from neighbouring Kyrgyzstan and fine-tuned on the Tajik basins (regional TL). Performance is assessed across four deep learning architectures (EA-LSTM, TFT, TiDE, TSMixer) over the growing season (March to October) when operational forecasts are most valuable. This experiment directly addresses RQ1

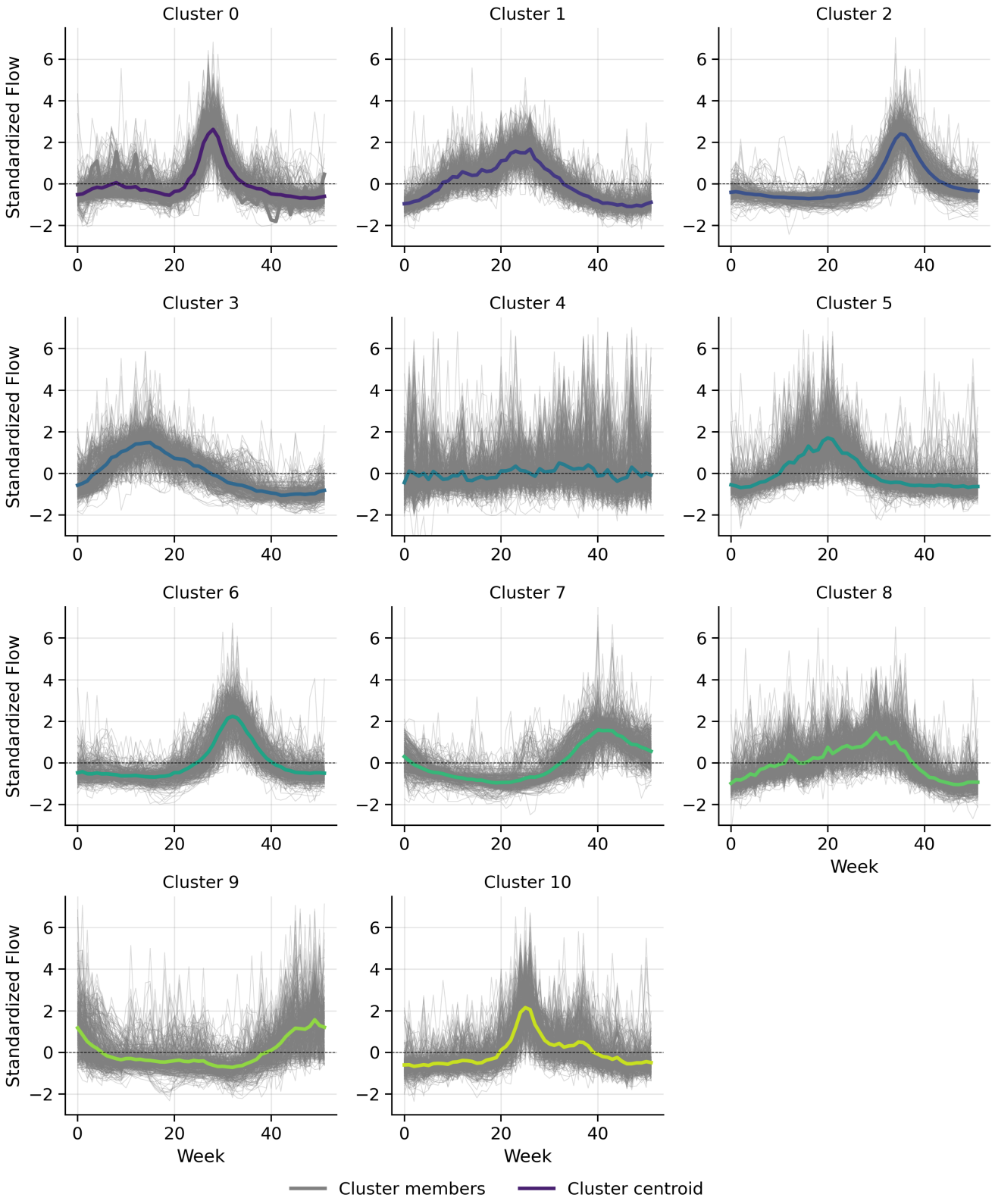

Figure 4: Cluster centroids and sample hydrographs showing the 11 optimal clusters identified through DTW-based clustering of standardised weekly streamflow data. Each subplot displays the cluster centroid (coloured line) and the hydrographs of sample members (grey lines).

(Transfer Learning Effectiveness), RQ2 (Forecast Horizon Impact), and RQ6 (Architecture-Dependent Transfer Learning).

6.4.1 RQ1: Transfer Learning Effectiveness

Table 6 presents a comparison of overall performance between benchmark and regional transfer learning approaches across four GoF metrics (NSE, KGE, RMSE, and MAE). The values report the median GoF across three forecast horizons (1, 5, and 10 days). The median NSE across all architectures improves from 0.85 ± 0.02 for the benchmark to 0.87 ± 0.01 for regional transfer learning. Similarly, the median KGE increases from 0.88 ± 0.02 to 0.90 ± 0.01. Error metrics show corresponding improvements, with median RMSE decreasing from 0.43 ± 0.04 mm/day to 0.39 ± 0.01 mm/day and median MAE reducing from 0.28 ± 0.04 mm/day to 0.25 ± 0.01 mm/day.

Examining individual architectures reveals distinct transfer learning responses (RQ6). EA-LSTM shows modest gains in KGE (0.89 ± 0.10 to 0.90 ± 0.08) while maintaining comparable NSE performance (0.85 ± 0.14 to 0.85 ± 0.12). TFT exhibits minimal changes in NSE (0.88 ± 0.10 for both) and a slight decrease in KGE from 0.93 ± 0.07 to 0.91 ± 0.07, with RMSE increasing from 0.35 ± 0.33 to 0.39 ± 0.33 mm/day and MAE from 0.22 ± 0.20 to 0.24 ± 0.20 mm/day. TiDE shows the most substantial gains, with median NSE increasing from 0.84 ± 0.09 to 0.87 ± 0.11 and median KGE improving from 0.87 ± 0.06 to 0.90 ± 0.06. TSMixer exhibits similar improvements, with median NSE rising from 0.85 ± 0.12 to 0.88 ± 0.10. Performance variability changes differ by architecture: EA-LSTM and TSMixer show reduced NSE standard deviation with transfer learning (0.14 to 0.12 and 0.12 to 0.10, respectively), TiDE exhibits increased variability (0.09 to 0.11), while TFT maintains consistent variability (0.10).

Table 6: Median model performance metrics (median ± standard deviation) across basins and forecasting horizons for two experimental scenarios: Benchmark (trained on 15 Tajik basins only) and Regional TL (pre-trained on Kyrgyz data and fine-tuned on Tajik). OVERALL represents the median performance across all architectures for each scenario. The performance metric of the best models within each architecture is shown in **bold**.

Architecture	Variant	NSE (-)	KGE (-)	RMSE (mm/d)	MAE (mm/d)
EALSTM	Benchmark	0.85 ± 0.14	0.89 ± 0.10	0.42 ± 0.42	0.26 ± 0.24
	Regional TL	**0.85 ± 0.12**	**0.90 ± 0.08**	**0.40 ± 0.36**	**0.26 ± 0.21**
TFT	Benchmark	**0.88 ± 0.10**	**0.93 ± 0.07**	**0.35 ± 0.33**	**0.22 ± 0.20**
	Regional TL	0.88 ± 0.10	0.91 ± 0.07	0.39 ± 0.33	0.24 ± 0.20
TIDE	Benchmark	0.84 ± 0.09	0.87 ± 0.06	0.45 ± 0.35	0.29 ± 0.22
	Regional TL	**0.87 ± 0.11**	**0.90 ± 0.06**	**0.40 ± 0.34**	**0.25 ± 0.21**
TSMIXER	Benchmark	0.85 ± 0.12	0.87 ± 0.08	0.45 ± 0.37	0.32 ± 0.24
	Regional TL	**0.88 ± 0.10**	**0.91 ± 0.07**	**0.38 ± 0.31**	**0.25 ± 0.19**
OVERALL	Benchmark	0.85 ± 0.02	0.88 ± 0.02	0.43 ± 0.04	0.28 ± 0.04
	Regional TL	**0.87 ± 0.01**	**0.90 ± 0.01**	**0.39 ± 0.01**	**0.25 ± 0.01**

Figure 5 presents the distribution of NSE values across the 15 Tajik basins for both benchmark and regional transfer learning variants at the three forecast horizons (1, 5, and 10 days), alongside the performance of the dummy model baseline. At the 1-day forecast horizon, all models achieve median NSE values that closely approach the dummy model's performance of 0.97. Regional transfer learning shows consistent improvements across three architectures: the median NSE value for EA-LSTM increases from 0.96 to 0.97, TFT from 0.97 to 0.97, TiDE from 0.94 to 0.96, and TSMixer from 0.93 to 0.95. At the

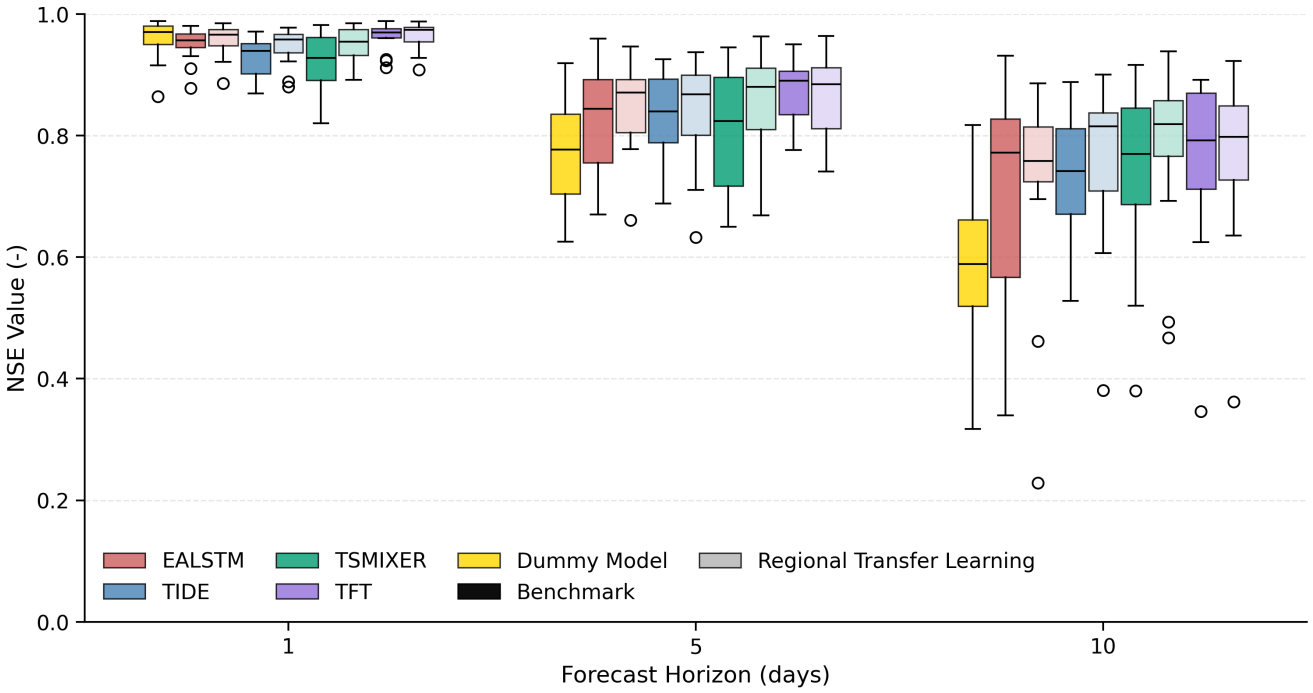

Figure 5: NSE values across 15 Tajik basins for four deep learning architectures (EA-LSTM, TiDE, TFT, TSMixer) at three forecast horizons (1, 5, and 10 days). Each architecture is presented in two variants: benchmark models (darker colours), trained solely on Tajik data, and regional transfer learning models (lighter colours), pre-trained on 59 Kyrgyz basins and fine-tuned on Tajik basins. The yellow boxplots represent the baseline performance of the dummy model.

5-day horizon, where the median dummy model performance declines to 0.78, the deep learning models maintain substantially higher performance. Regional transfer learning demonstrates median NSE improvements for all architectures: EA-LSTM from 0.84 to 0.87, TFT from 0.89 to 0.89, TiDE from 0.84 to 0.87, and TSMixer from 0.82 to 0.88. The 10-day forecast horizon reveals the most substantial differences between training approaches. While the median dummy model performance declines to 0.59, the deep learning models sustain median NSE values between 0.74 and 0.82. Regional transfer learning yields mixed results: EA-LSTM decreases slightly from 0.77 to 0.76, TFT improves marginally from 0.79 to 0.80, TiDE shows the largest improvement from 0.74 to 0.82, and TSMixer increases from 0.77 to 0.82.

The dummy model maintains median NSE values of 0.97, 0.78, and 0.59 at the 1-day, 5-day, and 10-day horizons, respectively. No deep learning model outperforms the dummy model at the 1-day lead time. TFT exhibits minimal responsiveness to regional transfer learning, with median performance remaining unchanged or showing slight decreases across several metrics as highlighted by the bold values in Table 6.

6.4.2 RQ2: Forecast Horizon Impact

Figure E8 presents the relationship between benchmark model performance (NSE) and the relative improvement achieved through regional transfer learning across three forecast horizons (1, 5, and 10 days). Two patterns emerge: First, a negative relationship between benchmark performance and relative improvement across all horizons. Second, the spread of relative improvements increases substantially at longer forecast horizons.

Examining relative NSE improvements alone is insufficient due to the bounded nature of NSE (maximum value of 1.0). This makes relative improvements inherently easier to achieve at lower baseline performance levels. To address this limitation, I introduce the Remaining Skill Captured (RSC) metric:

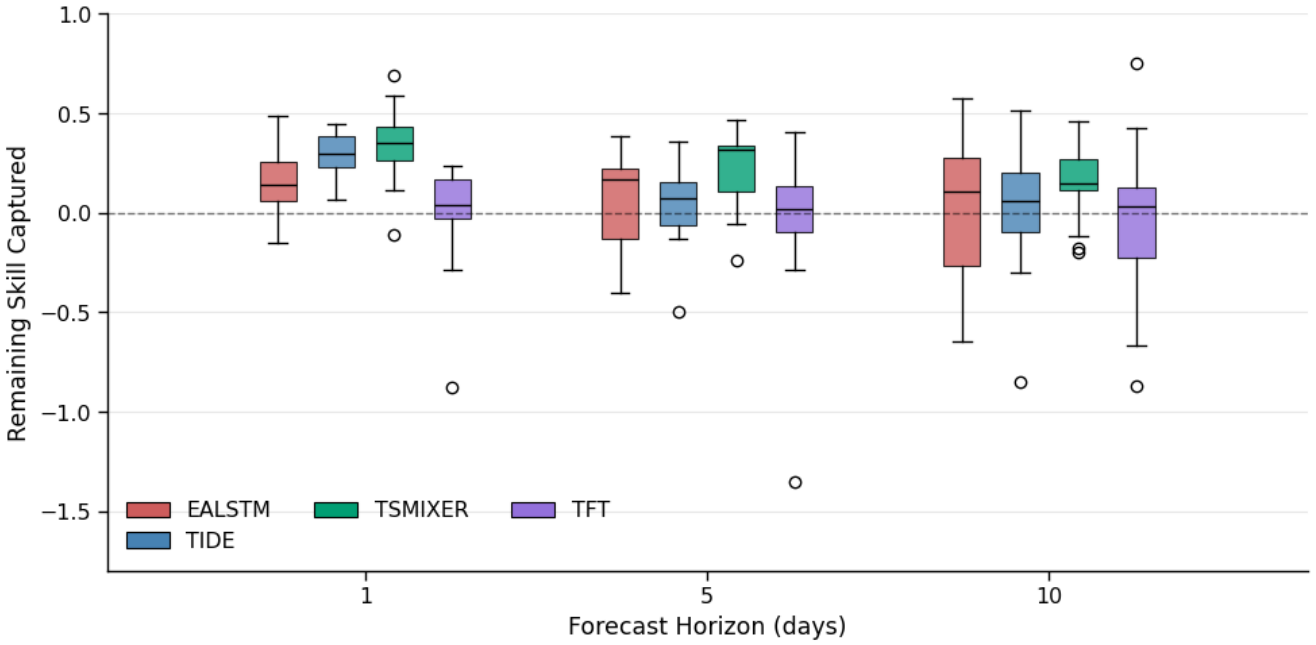

Figure 6: RSC values for four deep learning architectures across three forecast horizons (1, 5, and 10 days). RSC quantifies the proportion of potential Nash-Sutcliffe Efficiency improvement achieved through regional transfer learning, calculated as the ratio of actual improvement to maximum possible improvement. Each boxplot represents the distribution of RSC values across the 15 Tajik basins for each architecture-horizon combination.

$$\text{RSC } (-) = \frac{\text{NSE}_{\text{regional TL}} - \text{NSE}_{\text{benchmark}}}{1 - \text{NSE}_{\text{benchmark}}}$$

This metric quantifies the proportion of performance improvement that transfer learning achieves, accounting for the diminishing returns near NSE's theoretical maximum.

Figure 6 shows RSC values across forecast horizons for all architectures. Three of the four architectures (TSMixer, TiDE, and EA-LSTM) demonstrate consistently positive median RSC values across all forecast horizons. TFT shows minimal positive RSC values, consistent with its limited responsiveness to regional transfer learning observed in subsubsection 6.4.1. RSC values are consistently higher at shorter forecast horizons. At the 1-day horizon, TSMixer achieves the highest RSC of 0.35, followed by TiDE (0.30), EA-LSTM (0.14), and TFT (0.04). RSC values decrease progressively with forecast horizon: at 5 days, values drop to 0.32, 0.07, 0.17, and 0.02 for TSMixer, TiDE, EA-LSTM, and TFT, respectively, while at 10 days, they further decline to 0.14, 0.06, 0.10, and 0.03.

6.5 Experiment 2: Global Transfer Learning

6.5.1 Phase 1: In-Memory Global Transfer Learning

This phase evaluates global transfer learning by comparing benchmark models against volume-based global transfer learning (852 catchments) and similarity-based global transfer learning (159 catchments). I pre-train the models on data from catchments in the USA, Chile, and Switzerland. The target domains comprise Tajikistan (15 basins) and Kyrgyzstan (59 basins). This phase addresses RQ3 (Data Volume vs. Relevance), RQ5 (Target Domain Data Availability), and RQ6 (Architecture-Dependent Transfer Learning).

Tajikistan Analysis (15 basins) Table 7 presents four performance metrics (NSE, KGE, RMSE, MAE) comparing benchmark models trained solely on Tajik data against volume-based and similarity-based global transfer learning. The overall median NSE values are 0.85 for the benchmark, 0.88 for volume-based, and 0.89 for similarity-based selection. The benchmark models are identical to those in Experiment 1. Both global transfer learning strategies outperform the benchmark across all architectures and metrics.

Examining individual architectures reveals distinct transfer learning responses (RQ6). EA-LSTM achieves NSE of 0.87 (volume-based) vs. 0.86 (similarity-based), with KGE showing the opposite pattern (0.89 vs. 0.91). TiDE achieves NSE of 0.88 (volume-based) vs. 0.89 (similarity-based). TFT shows equivalent NSE performance (0.90 for both strategies). TSMixer achieves NSE of 0.87 (volume-based) vs. 0.88 (similarity-based).

Table 7: Median model performance metrics (median ± standard deviation) across basins and forecasting horizons for three experimental scenarios: Benchmark (trained on 15 Tajik basins only), Volume-Based global TL (pre-trained on 852 basins across Chile, Switzerland and the US) and Regional global TL (pre-trained on 161 basins across Chile, Switzerland and the US). The performance metric of the best models within each architecture is shown in **bold**. The OVERALL section shows summary statistics (median of architecture medians ± standard deviation) across all architectures for each scenario.

Architecture	Variant	NSE (-)	KGE (-)	RMSE (mm/d)	MAE (mm/d)
EALSTM	Benchmark	0.85 ± 0.14	0.89 ± 0.10	0.42 ± 0.42	0.26 ± 0.24
	Volume-Based	**0.87 ± 0.10**	0.89 ± 0.07	**0.38 ± 0.32**	0.25 ± 0.20
	Similarity-Based	0.86 ± 0.09	**0.91 ± 0.07**	0.39 ± 0.34	**0.24 ± 0.23**
TFT	Benchmark	0.88 ± 0.10	**0.93 ± 0.07**	0.35 ± 0.33	0.22 ± 0.20
	Volume-Based	**0.90 ± 0.07**	0.91 ± 0.06	0.35 ± 0.31	**0.22 ± 0.19**
	Similarity-Based	0.90 ± 0.09	0.92 ± 0.06	**0.35 ± 0.30**	0.22 ± 0.19
TIDE	Benchmark	0.84 ± 0.09	0.87 ± 0.06	0.45 ± 0.35	0.29 ± 0.22
	Volume-Based	0.88 ± 0.11	0.91 ± 0.06	**0.34 ± 0.40**	0.23 ± 0.22
	Similarity-Based	**0.89 ± 0.09**	**0.91 ± 0.05**	0.37 ± 0.33	**0.22 ± 0.20**
TSMIXER	Benchmark	0.85 ± 0.12	0.87 ± 0.08	0.45 ± 0.37	0.32 ± 0.24
	Volume-Based	0.87 ± 0.12	0.89 ± 0.08	0.39 ± 0.36	**0.25 ± 0.25**
	Similarity-Based	**0.88 ± 0.09**	**0.89 ± 0.06**	**0.38 ± 0.31**	0.25 ± 0.19
OVERALL	Benchmark	0.85 ± 0.02	0.88 ± 0.02	0.43 ± 0.04	0.28 ± 0.04
	Volume-Based	0.88 ± 0.01	0.90 ± 0.01	**0.36 ± 0.02**	0.24 ± 0.02
	Similarity-Based	**0.89 ± 0.01**	**0.91 ± 0.01**	0.38 ± 0.01	**0.23 ± 0.01**

Figure 7 presents the distribution of NSE values across the 15 Tajik basins for benchmark, volume-based, and similarity-based transfer learning variants at three forecast horizons, alongside the dummy model baseline. At the 1-day forecast horizon, the dummy model achieves a median NSE of 0.97. Both transfer learning strategies achieve comparable performance across architectures: EA-LSTM shows 0.96 (benchmark), 0.97 (volume-based), and 0.97 (similarity-based); TiDE shows 0.94 (benchmark), 0.98 (volume-based), and 0.97 (similarity-based); TFT shows 0.97 (benchmark), 0.97 (volume-based), and 0.97 (similarity-based); TSMixer shows 0.93 (benchmark), 0.97 (volume-based), and 0.96 (similarity-based).

At the 5-day horizon, the dummy model median NSE declines to 0.78. EA-LSTM achieves 0.84 (benchmark), 0.87 (volume-based), and 0.87 (similarity-based). TiDE shows 0.84 (benchmark), 0.89 (volume-

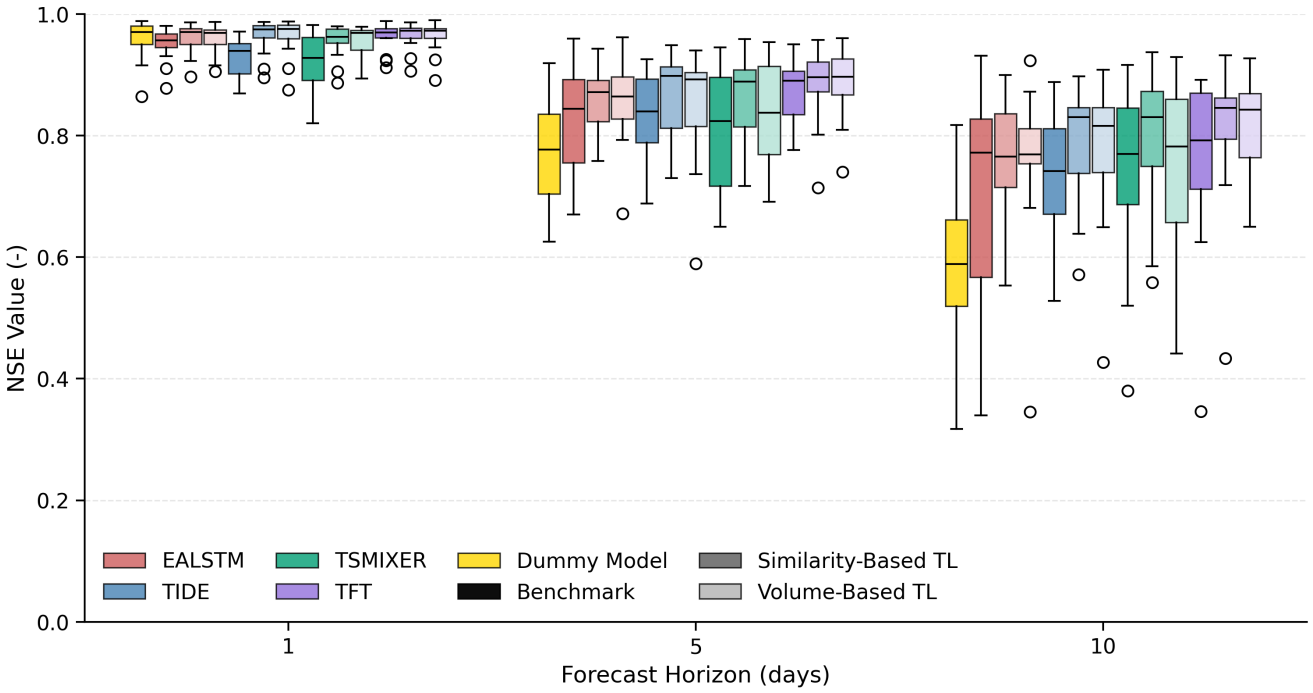

Figure 7: NSE values across 15 Tajik basins for four deep learning architectures (EA-LSTM, TiDE, TFT, TSMixer) at three forecast horizons (1, 5, and 10 days). Each architecture is shown in three variants: benchmark models (darkest colors) trained solely on Tajik data, volume-based global transfer learning models (medium colors) pre-trained on 852 catchments from the USA, Chile, and Switzerland and fine-tuned on Tajik data, and similarity-based global transfer learning models (lightest colors) pre-trained on 159 hydrologically similar catchments from the same regions and fine-tuned on Tajik basins. The yellow boxplots represent the baseline performance of the dummy model.

based), and 0.90 (similarity-based). TFT achieves 0.89 (benchmark), 0.90 (volume-based), and 0.90 (similarity-based). TSMixer shows 0.82 (benchmark), 0.84 (volume-based), and 0.89 (similarity-based).

At the 10-day forecast horizon, the dummy model median NSE reaches 0.59. EA-LSTM shows 0.77 (benchmark), 0.77 (volume-based), and 0.77 (similarity-based). TiDE achieves 0.74 (benchmark), 0.82 (volume-based), and 0.83 (similarity-based). TFT shows 0.79 (benchmark), 0.84 (volume-based), and 0.85 (similarity-based). TSMixer achieves 0.77 (benchmark), 0.78 (volume-based), and 0.83 (similarity-based). Transfer learning generally shifts the interquartile ranges upward compared to benchmark models across all horizons.

Kyrgyzstan Analysis (59 basins) Table 8 presents the four performance metrics comparing benchmark models trained solely on 59 Kyrgyz basins with those trained using volume-based and similarity-based global transfer learning. The overall median NSE values are 0.85 for the benchmark, 0.86 for both volume-based and similarity-based selection methods. Both global transfer learning strategies outperform the benchmark, though the overall NSE improvement of 0.01 is smaller than the 0.04 observed in Tajikistan.

For individual architectures, EA-LSTM achieves an NSE of 0.86 for all three variants (benchmark, volume-based, and similarity-based), though KGE favours volume-based selection (0.89) over similarity-based (0.88). TiDE shows NSE of 0.84 (benchmark), 0.86 (volume-based), and 0.86 (similarity-based). TFT achieves NSE of 0.85 (benchmark), 0.87 (volume-based), and 0.87 (similarity-based). TSMixer shows NSE of 0.83 (benchmark), 0.83 (volume-based), and 0.84 (similarity-based). Compared to Tajikistan, where similarity-based selection achieved the highest overall NSE (0.89), Kyrgyzstan shows equivalent

performance between selection strategies (both 0.86).

Table 8: Median model performance metrics (median ± standard deviation) across basins and forecasting horizons for three experimental scenarios: Benchmark (trained on 59 Kyrgyz basins only), Volume-Based global TL (pre-trained on 852 basins across Chile, Switzerland and the US) and Similarity-Based global TL (pre-trained on 161 basins across Chile, Switzerland and the US). The performance metric of the best models within each architecture is shown in **bold**. The OVERALL section shows summary statistics (median of architecture medians ± standard deviation) across all architectures for each scenario.

Architecture	Variant	NSE (-)	KGE (-)	RMSE (mm/d)	MAE (mm/d)
EALSTM	Benchmark	0.86 ± 0.12	0.88 ± 0.09	0.37 ± 0.26	0.23 ± 0.15
	Volume-Based	**0.86 ± 0.12**	**0.89 ± 0.09**	0.36 ± 0.26	**0.23 ± 0.15**
	Similarity-Based	0.86 ± 0.12	0.88 ± 0.08	**0.36 ± 0.26**	0.23 ± 0.15
TFT	Benchmark	0.85 ± 0.12	0.89 ± 0.09	0.37 ± 0.26	0.23 ± 0.15
	Volume-Based	0.87 ± 0.14	**0.91 ± 0.08**	0.34 ± 0.25	**0.21 ± 0.14**
	Similarity-Based	**0.87 ± 0.12**	0.90 ± 0.08	**0.34 ± 0.25**	0.21 ± 0.14
TIDE	Benchmark	0.84 ± 0.17	0.85 ± 0.12	0.40 ± 0.29	0.26 ± 0.17
	Volume-Based	**0.86 ± 0.13**	**0.90 ± 0.10**	**0.36 ± 0.26**	**0.22 ± 0.15**
	Similarity-Based	0.86 ± 0.15	0.88 ± 0.11	0.38 ± 0.27	0.23 ± 0.16
TSMIXER	Benchmark	0.83 ± 0.19	0.88 ± 0.11	0.39 ± 0.29	0.24 ± 0.17
	Volume-Based	0.83 ± 0.21	0.88 ± 0.11	0.39 ± 0.29	0.24 ± 0.16
	Similarity-Based	**0.84 ± 0.16**	**0.88 ± 0.09**	**0.38 ± 0.28**	**0.24 ± 0.16**
OVERALL	Benchmark	0.85 ± 0.01	0.88 ± 0.02	0.38 ± 0.01	0.24 ± 0.01
	Volume-Based	**0.86 ± 0.01**	**0.89 ± 0.01**	**0.36 ± 0.02**	**0.23 ± 0.01**
	Similarity-Based	0.86 ± 0.01	0.88 ± 0.01	0.37 ± 0.02	0.23 ± 0.01

Figure 8 presents the distribution of NSE values across the 59 Kyrgyz basins over the three forecasting horizons. At the 1-day forecast horizon, the dummy model achieves a median NSE of 0.97. Most architectures show equivalent performance across all variants: EA-LSTM achieves 0.96 (benchmark), 0.97 (volume-based), and 0.97 (similarity-based); TiDE shows 0.96 (benchmark), 0.97 (volume-based), and 0.97 (similarity-based); TFT achieves 0.96 (benchmark), 0.97 (volume-based), and 0.97 (similarity-based); TSMixer shows 0.96 (benchmark), 0.97 (volume-based), and 0.96 (similarity-based).

At the 5-day horizon, the dummy model median NSE is 0.79. EA-LSTM achieves 0.85 (benchmark), 0.86 (volume-based), and 0.86 (similarity-based). TiDE shows 0.83 (benchmark), 0.85 (volume-based), and 0.84 (similarity-based). TFT achieves 0.85 (benchmark), 0.86 (volume-based), and 0.87 (similarity-based). TSMixer shows 0.84 (benchmark), 0.84 (volume-based), and 0.85 (similarity-based).

At the 10-day forecast horizon, the dummy model median NSE is 0.58. EA-LSTM shows 0.77 (benchmark), 0.75 (volume-based), and 0.77 (similarity-based). TiDE achieves 0.73 (benchmark), 0.75 (volume-based), and 0.76 (similarity-based). TFT shows 0.75 (benchmark), 0.75 (volume-based), and 0.78 (similarity-based). TSMixer achieves 0.72 (benchmark), 0.70 (volume-based), and 0.70 (similarity-based) performance, showing a reduced median performance compared to the benchmark with both transfer learning strategies.

RQ5: Target Domain Data Availability The benefits of transfer learning differ between the data-limited Tajikistan case (15 basins) and the data-rich Kyrgyzstan case (59 basins). For Tajikistan, median

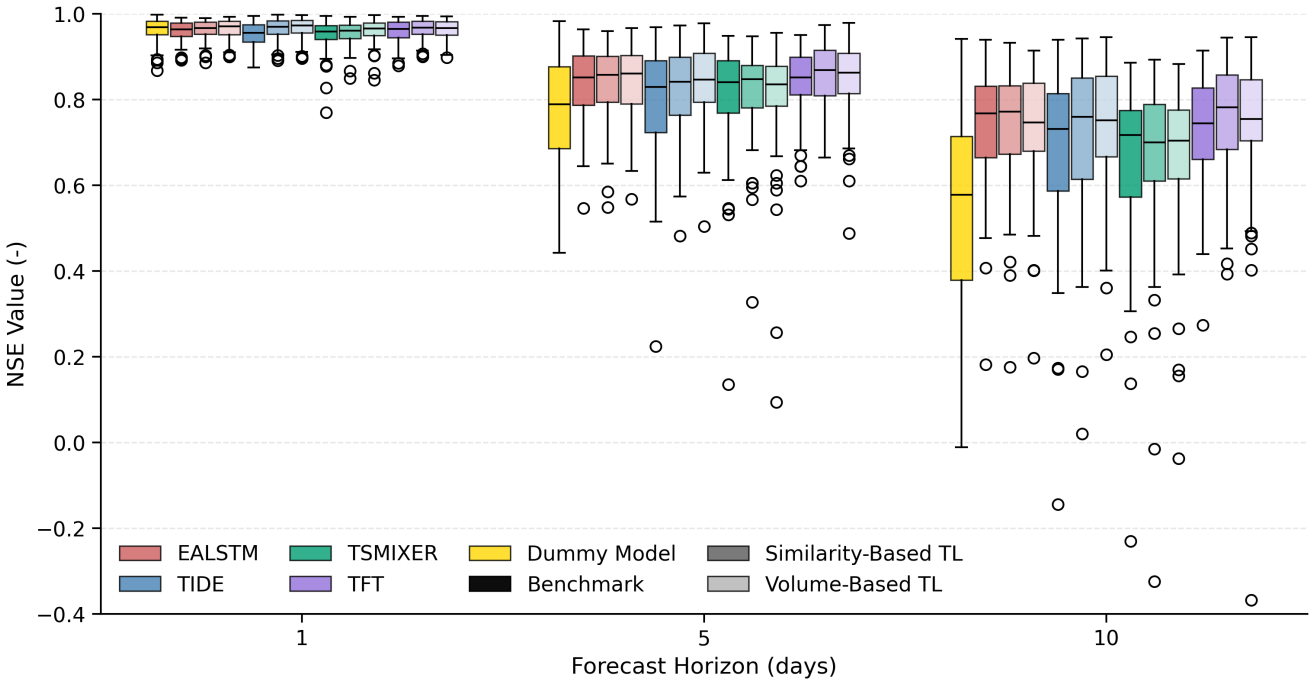

Figure 8: NSE values across 59 Kyrgyz basins for four deep learning architectures (EA-LSTM, TiDE, TFT, TSMixer) at three forecast horizons (1, 5, and 10 days). Each architecture is shown in three variants: benchmark models (darkest colors) trained solely on Kyrgyz data, volume-based global transfer learning models (medium colors) pre-trained on 852 catchments from the USA, Chile, and Switzerland and fine-tuned on Kyrgyz data, and similarity-based global transfer learning models (lightest colors) pre-trained on 159 hydrologically similar catchments from the same regions and fine-tuned on Kyrgyz basins. The yellow boxplots represent the baseline performance of the dummy model.

NSE across all architectures and horizons improves from 0.85 to 0.89 (0.04 improvement), while Kyrgyzstan shows an improvement from 0.85 to 0.86 (0.01 improvement).

Architecture-specific NSE improvements in Tajikistan range from 0.02 to 0.05: TiDE increases from 0.84 to 0.89, TSMixer from 0.85 to 0.88, EA-LSTM from 0.85 to 0.87, and TFT from 0.88 to 0.90. In Kyrgyzstan, improvements are smaller: TFT and TiDE improve by 0.02, TSMixer by 0.01, while EA-LSTM shows no improvement. The optimal transfer learning strategy varies across domains: similarity-based selection yields the highest overall NSE in Tajikistan (0.89). In contrast, volume-based selection performs best in Kyrgyzstan, though only when evaluated beyond two decimal places: the two selection strategies yield similar median NSE values (0.86). Performance variability, measured by standard deviation, decreases from 0.02 to 0.01 in both domains with transfer learning.

RSC Analysis Detailed RSC figures are presented in the Appendix (Figure E9 for Tajikistan and Figure E10 for Kyrgyzstan). Tajikistan exhibits higher RSC values than Kyrgyzstan across all architectures and forecast horizons.

For Tajikistan at the 1-day horizon, volume-based selection achieves higher median RSC for TiDE (0.58 vs. 0.55) and TSMixer (0.41 vs. 0.37), while similarity-based selection performs better for EA-LSTM (0.17 vs. 0.09). TFT shows minimal RSC for both strategies (-0.01 for similarity-based, 0.00 for volume-based). At the 5-day horizon, similarity-based selection outperforms volume-based for three architectures: EA-LSTM (0.09 vs. 0.13), TiDE (0.23 vs. 0.14), and TSMixer (0.29 vs. 0.16). At the 10-day horizon, volume-based selection achieves higher RSC for TiDE (0.24 vs. 0.18), while similarity-based selection performs better for TSMixer (0.19 vs. 0.08) and TFT (0.17 vs. 0.15).

For Kyrgyzstan, median RSC values are consistently lower. At the 1-day horizon, volume-based selection achieves higher RSC for all architectures: EA-LSTM (0.12 vs. 0.07), TiDE (0.32 vs. 0.29), and TSMixer (0.18 vs. 0.11). TFT shows minimal positive RSC (0.04 for volume-based, 0.07 for similarity-based). At the 5-day horizon, volume-based selection maintains superiority for TiDE (0.13 vs. 0.08), while similarity-based selection performs better for TSMixer (0.05 vs. -0.02) and TFT (0.10 vs. 0.07). At the 10-day horizon, volume-based selection achieves higher RSC for TiDE (0.14 vs. 0.10), while similarity-based selection shows advantages for TSMixer (0.00 vs. -0.01) and TFT (0.11 vs. 0.07).

RSC values decrease with forecast horizon for most architecture-strategy combinations. The decline from 1-day to 5-day horizons is generally larger than from 5-day to 10-day horizons, with some architectures showing stabilisation or slight increases between 5 and 10 days. TFT represents the only exception: in Tajikistan, RSC increases from 1-day to 10-day horizons.

6.5.2 Phase 2: Larger-than-RAM Global Transfer Learning

This phase evaluates global transfer learning by comparing benchmark models against volume-based global transfer learning (6,690 catchments) and similarity-based global transfer learning (1,850 catchments). The key distinction from Phase 1 is the use of larger-than-RAM datasets, which necessitates a modified preprocessing pipeline. Specifically, both meteorological forcings and target streamflow undergo per-catchment Z-score normalisation followed by Yeo-Johnson power transformation, and RevIN is not employed (see subsubsection 4.2.2). The target domains remain Tajikistan (15 basins) and Kyrgyzstan (59 basins). This phase addresses RQ3 (Data Volume vs. Relevance), RQ4 (Scalability to Large Datasets), RQ5 (Target Domain Data Availability), and RQ6 (Architecture-Dependent Transfer Learning).

Tajikistan Analysis

Transfer Learning Performance Table 9 presents the performance metrics comparing benchmark models trained solely on 15 Tajik basins against volume-based global transfer learning (6,690 catchments) and similarity-based global transfer learning (1,850 catchments). The overall median NSE values are 0.82 for the benchmark, 0.89 for volume-based, and 0.89 for similarity-based selection. The benchmark performance decreased from 0.85 in Phase 1 to 0.82 in Phase 2 for the same architectures and data, with only the preprocessing steps differing between phases. Both transfer learning strategies achieve a median NSE of 0.89, outperforming the benchmark by 0.07.

Examining individual architectures reveals distinct transfer learning responses (RQ6). EA-LSTM achieves NSE of 0.90 (volume-based) vs. 0.89 (similarity-based). TiDE achieves NSE of 0.89 (volume-based) vs. 0.87 (similarity-based). TFT exhibits the opposite pattern, with an NSE of 0.89 (volume-based) compared to 0.90 (similarity-based). TSMixer achieves NSE of 0.87 (volume-based) vs. 0.88 (similarity-based).

Regarding RQ4 (Scalability to Large Datasets), Phase 2 utilises datasets that are 8-11 times larger than those in Phase 1 (6,690 and 1,850 catchments versus 852 and 159). Comparing transfer learning performance between phases: similarity-based selection maintains a median NSE of 0.89 in both phases, while volume-based selection shows an NSE of 0.88 in Phase 1 and 0.89 in Phase 2. For individual architectures comparing Phase 1 to Phase 2 performance: EA-LSTM volume-based improves from 0.87 to 0.90, TiDE volume-based increases from 0.88 to 0.89, TFT volume-based decreases from 0.90 to 0.89, and TSMixer volume-based remains at 0.87.

Table 9: Median model performance metrics (median ± standard deviation) across basins and forecasting horizons for three experimental scenarios: Benchmark (trained on 15 Tajik basins only), Volume-Based global TL (pre-trained on 6690 basins in Caravan and CAMELS-CH) and Regional global TL (pre-trained on 1850 basins in Caravan and CAMELS-CH). The performance metric of the best models within each architecture is shown in **bold**. The OVERALL section shows summary statistics (median of architecture medians ± standard deviation) across all architectures for each scenario.

Architecture	Variant	NSE (-)	KGE (-)	RMSE (mm/d)	MAE (mm/d)
EALSTM	Benchmark	0.85 ± 5.61	0.84 ± 0.73	0.48 ± 2.09	0.30 ± 0.32
	Volume-Based	**0.90 ± 0.20**	**0.92 ± 0.11**	**0.36 ± 0.46**	0.22 ± 0.21
	Similarity-Based	0.89 ± 7.38	0.91 ± 0.77	0.38 ± 2.26	**0.22 ± 0.29**
TFT	Benchmark	0.88 ± 13.57	0.89 ± 1.16	0.38 ± 3.34	0.24 ± 0.30
	Volume-Based	0.89 ± 0.10	0.90 ± 0.09	0.37 ± 0.29	0.24 ± 0.18
	Similarity-Based	**0.90 ± 0.09**	**0.92 ± 0.06**	**0.34 ± 0.33**	**0.21 ± 0.19**
TIDE	Benchmark	0.79 ± 11.29	0.85 ± 1.06	0.50 ± 2.89	0.29 ± 0.42
	Volume-Based	**0.89 ± 0.08**	**0.91 ± 0.08**	**0.34 ± 0.32**	**0.22 ± 0.20**
	Similarity-Based	0.87 ± 1.87	0.90 ± 0.37	0.39 ± 1.14	0.24 ± 0.21
TSMIXER	Benchmark	0.76 ± 18.58	0.80 ± 1.50	0.60 ± 4.05	0.38 ± 0.44
	Volume-Based	0.87 ± 8.95	0.87 ± 0.78	0.46 ± 2.25	0.27 ± 0.24
	Similarity-Based	**0.88 ± 0.76**	**0.89 ± 0.20**	**0.38 ± 0.73**	**0.23 ± 0.20**
OVERALL	Benchmark	0.82 ± 0.04	0.84 ± 0.03	0.49 ± 0.08	0.29 ± 0.05
	Volume-Based	**0.89 ± 0.01**	0.91 ± 0.02	**0.37 ± 0.04**	0.23 ± 0.02
	Similarity-Based	0.89 ± 0.01	**0.91 ± 0.01**	0.38 ± 0.02	**0.23 ± 0.01**

Catastrophic Model Failures Several architectures exhibit substantially increased performance variability compared to Phase 1, as evidenced by the standard deviations in Table 9. The benchmark models exhibit particularly high variability: TSMixer displays a standard deviation of 18.58 for NSE, TFT shows 13.57, and TiDE exhibits a standard deviation of 11.29. These high standard deviations generally decrease with transfer learning. However, some architectures maintain considerable variability—TSMixer with volume-based selection shows a standard deviation of 8.95 for NSE, while EA-LSTM with similarity-based selection exhibits a standard deviation of 7.38.

The high variability in benchmark models is due to catastrophic model failures (NSE < 0) for specific basin-horizon combinations. Table E6 summarises the number of basins experiencing catastrophic failures across all forecast horizons for each model variant. Benchmark models exhibit 12 total catastrophic failures across all architectures, out of 45 (15 basins over three horizons). In contrast, volume-based and similarity-based transfer learning reduce this to 1 and 2 failures, respectively. The dummy model shows zero catastrophic failures across all basins and horizons. Among benchmark models, catastrophic failures occur most frequently at the 10-day horizon (5 out of 12 failures), followed by 1-day and 5-day horizons (4 and 3 failures, respectively).

Performance Across Forecast Horizons Figure 9 presents the distribution of NSE values across the 15 Tajik basins for benchmark, volume-based, and similarity-based transfer learning variants at three forecast horizons, alongside the dummy model baseline. At the 1-day forecast horizon, the dummy model achieves a median NSE of 0.97. EA-LSTM shows median NSE of 0.91 (benchmark), 0.97 (volume-based), and 0.97 (similarity-based). TiDE shows 0.92 (benchmark), 0.98 (volume-based), and 0.98 (similarity-based). TFT achieves 0.96 (benchmark), 0.97 (volume-based), and 0.97 (similarity-based). TSMixer

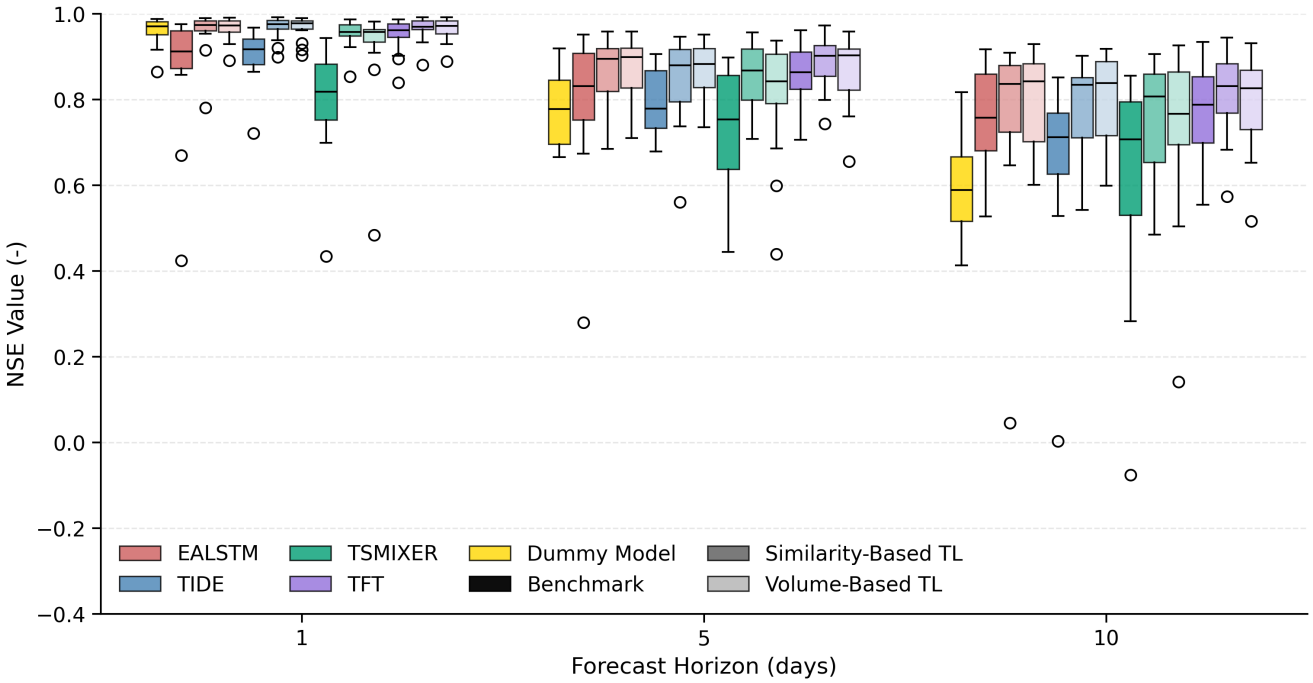

Figure 9: NSE values across 15 Tajik basins for four deep learning architectures (EA-LSTM, TiDE, TFT, TSMixer) at three forecast horizons (1, 5, and 10 days). Each architecture is shown in three variants: benchmark models (darkest colours) trained solely on Tajik data, volume-based global transfer learning models (medium colours) pre-trained on 6,690 catchments across Caravan and CAMELS-CH and fine-tuned on Tajik data, and similarity-based global transfer learning models (lightest colours) pre-trained on 1850 hydrologically similar catchments from the same regions and fine-tuned on Tajik basins. The yellow boxplots represent the baseline performance of the dummy model.

shows 0.82 (benchmark), 0.96 (volume-based), and 0.96 (similarity-based).

At the 5-day horizon, the dummy model median NSE declines to 0.78. EA-LSTM achieves median NSE of 0.83 (benchmark), 0.90 (volume-based), and 0.89 (similarity-based). TiDE shows 0.78 (benchmark), 0.88 (volume-based), and 0.88 (similarity-based). TFT achieves 0.86 (benchmark), 0.90 (volume-based), and 0.90 (similarity-based). TSMixer shows 0.75 (benchmark), 0.84 (volume-based), and 0.87 (similarity-based).

At the 10-day forecast horizon, the dummy model median NSE reaches 0.58. EA-LSTM shows median NSE of 0.76 (benchmark), 0.84 (volume-based), and 0.84 (similarity-based). TiDE achieves 0.71 (benchmark), 0.84 (volume-based), and 0.83 (similarity-based). TFT shows 0.79 (benchmark), 0.83 (volume-based), and 0.83 (similarity-based). TSMixer achieves 0.71 (benchmark), 0.77 (volume-based), and 0.81 (similarity-based). Transfer learning improves median model performance and shifts the interquartile ranges upward compared to benchmark models across all horizons.

Kyrgyzstan Analysis

Transfer Learning Performance Table 10 presents the performance metrics comparing benchmark models trained solely on 59 Kyrgyz basins against volume-based global transfer learning (6,690 catchments) and similarity-based global transfer learning (1,850 catchments). The overall median NSE values are 0.81 for the benchmark, 0.86 for the volume-based approach, and 0.85 for the similarity-based approach. The NSE improvement from benchmark to best transfer learning strategy is 0.05 for Kyrgyzstan (0.81 to 0.86) versus 0.07 for Tajikistan (0.82 to 0.89). Volume-based selection achieves higher

overall performance than similarity-based selection (0.86 vs. 0.85), differing from Tajikistan, where both strategies achieved a similar overall median NSE performance of 0.89.

For individual architectures, EA-LSTM achieves an NSE of 0.87 (volume-based) compared to 0.86 (similarity-based). TiDE shows NSE of 0.85 for both approaches, with KGE higher for volume-based selection (0.89 vs. 0.89 when examined beyond two decimal places). TFT achieves NSE of 0.87 for both strategies, with KGE favouring similarity-based selection (0.91 vs. 0.90). TSMixer shows NSE of 0.84 for both approaches, but achieves higher KGE with volume-based selection (0.89 vs. 0.87).

Comparing Phase 1 and Phase 2 performance for Kyrgyzstan: Phase 1 showed minimal NSE improvements of 0.01 from benchmark to transfer learning, while Phase 2 demonstrates larger gains of 0.05. Benchmark performance decreased from 0.85 in Phase 1 to 0.81 in Phase 2. The architectures showing the largest NSE improvements from benchmark to volume-based selection in Phase 2 are EA-LSTM (0.81 to 0.87, improvement of 0.06) and TSMixer (0.80 to 0.84, improvement of 0.04). In Phase 1, improvements were uniform at 0.01 across most architectures.

Table 10: Median model performance metrics (median ± standard deviation) across basins and forecasting horizons for three experimental scenarios: Benchmark (trained on 59 Kyrgyz basins only), Volume-Based global TL (pre-trained on 6690 basins in Caravan and CAMELS-CH) and Similarity-Based global TL (pre-trained on 1850 basins in Caravan and CAMELS-CH). The performance metric of the best models within each architecture is shown in **bold**. The OVERALL section shows summary statistics (median of architecture medians ± standard deviation) across all architectures for each scenario.

Architecture	Variant	NSE (-)	KGE (-)	RMSE (mm/d)	MAE (mm/d)
EALSTM	Benchmark	0.81 ± 8.42	0.87 ± 0.66	0.41 ± 0.90	0.27 ± 0.18
	Volume-Based	**0.87 ± 14.69**	**0.90 ± 0.96**	**0.34 ± 1.52**	**0.22 ± 0.17**
	Similarity-Based	0.86 ± 15.21	0.90 ± 0.89	0.36 ± 1.60	0.22 ± 0.18
TFT	Benchmark	0.85 ± 393.75	0.88 ± 4.81	0.37 ± 7.97	0.24 ± 0.41
	Volume-Based	**0.87 ± 1.11**	0.90 ± 0.24	0.34 ± 0.51	0.21 ± 0.15
	Similarity-Based	0.87 ± 2.04	**0.91 ± 0.34**	**0.34 ± 0.54**	**0.21 ± 0.15**
TIDE	Benchmark	0.81 ± 386.82	0.88 ± 4.02	0.43 ± 8.50	0.27 ± 0.41
	Volume-Based	0.85 ± 6.18	**0.89 ± 0.57**	0.39 ± 0.67	0.24 ± 0.16
	Similarity-Based	**0.85 ± 17.50**	0.89 ± 0.86	**0.38 ± 1.00**	**0.23 ± 0.16**
TSMIXER	Benchmark	0.80 ± 338.58	0.84 ± 4.45	0.42 ± 7.65	0.27 ± 0.47
	Volume-Based	**0.84 ± 76.93**	**0.89 ± 1.78**	**0.37 ± 3.11**	**0.22 ± 0.21**
	Similarity-Based	0.84 ± 28.11	0.87 ± 1.14	0.39 ± 2.07	0.23 ± 0.22
OVERALL	Benchmark	0.81 ± 0.02	0.87 ± 0.02	0.42 ± 0.02	0.27 ± 0.01
	Volume-Based	**0.86 ± 0.01**	**0.90 ± 0.01**	**0.36 ± 0.02**	**0.22 ± 0.01**
	Similarity-Based	0.85 ± 0.01	0.89 ± 0.01	0.37 ± 0.02	0.22 ± 0.01

Catastrophic Model Failures As observed for Tajikistan, several architectures exhibit substantially increased performance variability, as evidenced by the standard deviations in Table 10. The benchmark models exhibit higher variability than those in Tajikistan: TFT displays a standard deviation of 393.75 for NSE, TiDE shows 386.82, and TSMixer shows 338.58. Transfer learning reduces this variability, though some architectures maintain considerable standard deviations—TSMixer with volume-based selection shows 76.93 for NSE, while similarity-based selection exhibits 28.11.

Table E7 shows that the benchmark models in Kyrgyzstan exhibit 33 out of 177 (59 basins over three

horizons) total catastrophic failures across all architectures, compared to 12 (out of 45) in Tajikistan. Volume-based and similarity-based transfer learning reduce this to 17 and 20 failures, respectively, compared to 1 and 2 in Tajikistan. The dummy model maintains zero catastrophic failures. Among benchmark models in Kyrgyzstan, catastrophic failures occur most frequently at the 10-day horizon (16 out of 33 failures), followed by the 5-day horizon (12 failures) and 1-day horizon (5 failures).

Performance Across Horizons Figure 10 presents the distribution of NSE values across the 59 Kyrgyz basins for benchmark, volume-based, and similarity-based transfer learning variants at three forecast horizons, alongside the dummy model baseline. At the 1-day forecast horizon, the dummy model achieves a median NSE of 0.97. EA-LSTM shows median NSE of 0.95 (benchmark), 0.97 (volume-based), and 0.97 (similarity-based). TiDE shows 0.95 (benchmark), 0.97 (volume-based), and 0.97 (similarity-based). TFT achieves 0.96 (benchmark), 0.97 (volume-based), and 0.97 (similarity-based). TSMixer shows 0.94 (benchmark), 0.96 (volume-based), and 0.96 (similarity-based).

At the 5-day horizon, the dummy model median NSE declines to 0.79. EA-LSTM achieves median NSE of 0.81 (benchmark), 0.85 (volume-based), and 0.86 (similarity-based). TiDE shows 0.82 (benchmark), 0.84 (volume-based), and 0.85 (similarity-based). TFT achieves 0.85 (benchmark), 0.86 (volume-based), and 0.87 (similarity-based). TSMixer shows 0.80 (benchmark), 0.85 (volume-based), and 0.85 (similarity-based).

At the 10-day forecast horizon, the dummy model median NSE reaches 0.58. EA-LSTM shows median NSE of 0.71 (benchmark), 0.76 (volume-based), and 0.77 (similarity-based). TiDE achieves 0.69 (benchmark), 0.74 (volume-based), and 0.74 (similarity-based). TFT shows 0.73 (benchmark), 0.79 (volume-based), and 0.78 (similarity-based). TSMixer achieves 0.70 (benchmark), 0.76 (volume-based), and 0.76 (similarity-based). The largest improvements from benchmark to transfer learning are observed for TSMixer and TFT, both of which show increases of 0.06 NSE.

RSC Analysis The RSC figures are presented in the Appendix (Figure E11 for Tajikistan and Figure E12 for Kyrgyzstan). Tajikistan exhibits higher RSC values than Kyrgyzstan across all architectures and forecast horizons.

For Tajikistan at the 1-day horizon, TSMixer achieves the highest median RSC (0.82 for similarity-based, 0.71 for volume-based), while TFT shows the lowest (0.30 for both strategies). EA-LSTM yields 0.62 for both strategies, while TiDE achieves 0.68 (similarity-based) and 0.65 (volume-based). At the 5-day horizon, median RSC values decline to: TSMixer (0.46 similarity-based, 0.38 volume-based), TiDE (0.35 similarity-based, 0.41 volume-based), EA-LSTM (0.22 similarity-based, 0.23 volume-based), and TFT (0.18 similarity-based, 0.07 volume-based). At the 10-day horizon, values further decline to: TSMixer (0.35 similarity-based, 0.29 volume-based), TiDE (0.35 similarity-based, 0.38 volume-based), EA-LSTM (0.12 similarity-based, 0.27 volume-based), and TFT (0.15 similarity-based, -0.01 volume-based).

For Kyrgyzstan, median RSC values are consistently lower. At the 1-day horizon, TiDE exhibits the highest median RSC values (0.41 similarity-based, 0.44 volume-based), contrasting with Tajikistan, where TSMixer achieved the highest values. EA-LSTM shows 0.37 (similarity-based) vs. 0.41 (volume-based), TSMixer achieves 0.27 (similarity-based) vs. 0.33 (volume-based), and TFT shows 0.19 (similarity-based) vs. 0.11 (volume-based). At the 5-day horizon, values decline to: EA-LSTM (0.17 similarity-based, 0.18 volume-based), TiDE (0.13 similarity-based, 0.11 volume-based), TSMixer (0.14 similarity-based, 0.15 volume-based), and TFT (0.10 similarity-based, 0.09 volume-based). At the 10-day horizon: EA-LSTM (0.10 similarity-based, 0.14 volume-based), TiDE (0.14 similarity-based, 0.12 volume-based), TSMixer

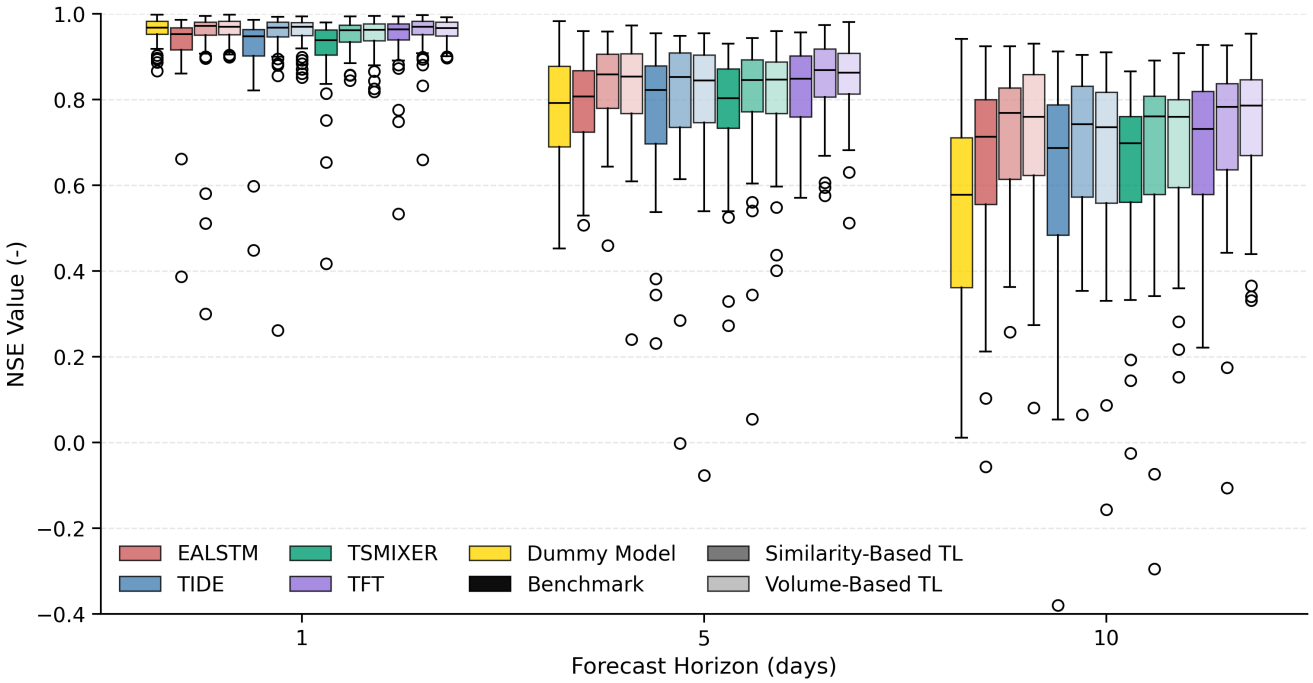

Figure 10: NSE values across 59 Kyrgyz basins for four deep learning architectures (EA-LSTM, TiDE, TFT, TSMixer) at three forecast horizons (1, 5, and 10 days). Each architecture is shown in three variants: benchmark models (darkest colours) trained solely on Kyrgyz data, volume-based global transfer learning models (medium colours) pre-trained on 6,690 catchments across Caravan and CAMELS-CH and fine-tuned on Kyrgyz data, and similarity-based global transfer learning models (lightest colours) pre-trained on 1850 hydrologically similar catchments from the same regions and fine-tuned on Kyrgyz basins. The yellow boxplots represent the baseline performance of the dummy model.

(0.14 similarity-based, 0.18 volume-based), and TFT (0.13 similarity-based, 0.11 volume-based).

RSC values decrease with forecast horizon for most architecture-strategy combinations. The decline from a 1-day to a 5-day horizon is generally larger than that from a 5-day to a 10-day horizon. TiDE maintains nearly identical RSC values between 5-day and 10-day horizons for Tajikistan similarity-based selection (0.35 at both). In Kyrgyzstan, TFT shows a slight increase in RSC from 5 to 10 days for both strategies (0.10 to 0.13 for similarity-based, 0.09 to 0.11 for volume-based). All architectures maintain positive median RSC values across all horizons, except for TFT with volume-based selection in Tajikistan, which exhibits a negative median RSC at the 10-day horizon (-0.01).

7 Discussion

7.1 Overview of Research Findings

This section synthesises the principal findings of the thesis by directly addressing the six research questions that guided my research. A detailed examination of these findings and their broader implications follows in the subsequent sections.

RQ1: Does transfer learning improve hydrological forecasting performance?
Yes, transfer learning generally improves streamflow forecasting performance, though with varying magnitudes across architectures. In the regional experiment, EA-LSTM showed improvements at shorter forecast horizons (1-day and 5-day), while TFT exhibited minimal responsiveness across all horizons.

RQ2: Do transfer learning benefits increase at longer forecast horizons?
No, the benefits, when measured by the Remaining Skill Captured metric, do not increase at longer forecast horizons. My analysis reveals that performance gains are most pronounced at short lead times and systematically diminish as the forecast horizon extends to 10 days. This pattern suggests that transfer learning primarily enhances the model's ability to extrapolate the autoregressive streamflow signal, rather than improving its sensitivity to meteorological forcing.

RQ3: Is data volume or hydrological relevance more critical for source domain selection?
Neither strategy is consistently superior. The comparison between selecting a large volume of catchments with low human influence versus a hydrologically similar set of catchments showed that no approach is consistently better. For instance, in the larger-than-RAM experiment for Tajikistan, both the volume-based (6,690 basins) and similarity-based (1,850 basins) strategies achieved an identical median NSE of 0.89 across all models and forecasting horizons. However, the similarity-based approach achieved this with 81% less data.

RQ4: Can transfer learning benefits scale to larger-than-RAM datasets?
Yes, the benefits of transfer learning scale to pre-training datasets that exceed system RAM. The chunking and preprocessing pipeline developed for Experiment 2, Phase 2, successfully enabled training on the entire Caravan dataset. However, my results reveal that data preprocessing can have a significant impact on model performance. Per-basin preprocessing of the meteorological forcing causes catastrophic model failures, which are not observed for the dummy model. Transfer learning reduced these failures by over 90% in Tajikistan (from 12 to 1) and by nearly 50% in Kyrgyzstan (from 33 to 17).

RQ5: How does target domain data availability affect transfer learning benefits?
The benefits of transfer learning are inversely proportional to the amount of data available in the target domain. The improvements were consistently more pronounced in data-limited Tajikistan (15 basins) compared to data-rich Kyrgyzstan (59 basins). In Experiment 2, Phase 1, for example, the median NSE across all models and forecasting horizons improved by 0.04 in Tajikistan, whereas it improved by 0.01 in Kyrgyzstan. This supports the hypothesis that transfer learning provides the greatest relative information gain where local data is scarcest.

RQ6: How do different deep learning architectures respond to transfer learning?
Architectural response depends on model complexity. Simpler MLP-based models, such as TSMixer and TiDE, benefit from transfer learning even with smaller, regional datasets (e.g., pre-training on 59 Kyrgyz basins). EA-LSTM, with its more complex recurrent architecture, exhibits modest improvements from regional transfer learning, which are primarily confined to shorter forecast horizons. In contrast, TFT, the most architecturally sophisticated model with attention mechanisms and multiple processing pathways, requires large-scale global datasets with hundreds or thousands of basins to demonstrate significant improvement. However, all architectures converged to similar performance levels after global transfer learning was applied. This suggests the existence of a performance ceiling for this specific forecasting task, potentially limiting the observable advantages of more complex models.

7.2 The Nature of Transfer Learning Improvements

7.2.1 Primary Finding: Enhancing Temporal Extrapolation, Not Meteorological Sensitivity

The analysis of the Remaining Skill Captured (RSC) metric reveals that the performance gains from transfer learning are primarily driven by enhanced temporal extrapolation of the streamflow signal, not by an improved response to meteorological forcing. This conclusion is supported by a consistent pattern

observed across all experiments: the RSC is highest at short forecast horizons, where autoregressive signals dominate, and systematically decreases as the forecast horizon extends and the model relies more on meteorological inputs. In the regional experiment, for instance, models captured up to 35% of the remaining improvable skill at the 1-day horizon, a figure that dropped to 10–15% at the 10-day horizon.

This pattern may reflect a limitation in my experimental setup rather than an inherent property of transfer learning. I did not provide the models with temporal context through cyclical encoding of day-of-year or month, meaning they cannot directly learn that identical meteorological conditions have different hydrological implications across seasons—April temperatures trigger snowmelt while September temperatures do not. Without this information, the models must infer seasonal context indirectly from the streamflow signal itself, effectively forcing them to rely on autoregressive patterns as their primary source of temporal awareness. Consequently, transfer learning may be improving pattern recognition for seasonal streamflow signatures, rather than enhancing meteorological sensitivity, simply because I constrained the models to learn in this way.

Additionally, the documented limitations of ERA5-Land reanalysis in mountainous regions—particularly its tendency to smooth precipitation extremes and miss orographic effects (Clerc-Schwarzenbach et al., 2024)—may further incentivise models to prioritise the more reliable streamflow signal over potentially noisy meteorological forcing. If the meteorological data inadequately represent true rainfall-runoff relationships, models would learn to downweight these inputs regardless of transfer learning, as the autoregressive pathway provides more consistent predictive value.

This pattern only becomes clear through the RSC metric, which quantifies the fraction of reducible error captured by transfer learning. Mathematically, RSC equals one minus the ratio of new to baseline mean squared errors (see Appendix F for derivation):

$$\text{RSC} = \frac{\text{NSE}_{\text{transfer learning}} - \text{NSE}_{\text{benchmark}}}{1 - \text{NSE}_{\text{benchmark}}} = 1 - \frac{\text{MSE}_{\text{transfer learning}}}{\text{MSE}_{\text{benchmark}}}$$

This formulation addresses the bounded nature of NSE. A model with a low baseline NSE has a large margin for absolute improvement, while a model with a high NSE of 0.9, for example, can only improve by a maximum of 0.1. Observing large absolute gains only in low-performing basins is therefore a mathematical consequence of the bounded metric. The RSC metric normalises for this available room for improvement. For instance, consider an NSE increase from 0.95 to 0.97. This absolute gain of only 0.02 represents 40% capture of the remaining potential. In contrast, a larger absolute gain of 0.10 from a baseline of 0.60 to 0.70 captures only 25% of its much larger potential.

I acknowledge two limitations of this analysis. First, my mechanistic interpretation—that transfer learning improves temporal extrapolation rather than meteorological sensitivity—relies solely on NSE-based RSC patterns. The same analysis could be applied to other bounded metrics, such as KGE or log-NSE, to validate this trend. Second, RSC assumes the theoretical maximum (NSE = 1) represents achievable perfection. In practice, measurement uncertainty, gauge errors, and inherent system stochasticity mean that NSE = 0.95 might represent the effective ceiling for many basins. Beyond this point, *improvements* could merely fit measurement noise rather than capture meaningful hydrological patterns.

Despite these limitations, RSC clarifies the answer to my second research question: the benefits of transfer learning do not increase at longer forecast horizons; rather, its effectiveness in capturing remaining skill diminishes as the autoregressive signal weakens. This distinction is important when comparing my findings to recent studies. Ryd and Nearing (2025) for instance, show that transfer learning benefits flood forecasting in non-autoregressive models. In their setup, performance gains can only come from a better utilisation of meteorological forcing, as there is no streamflow history to exploit. My work implements

autoregressive models; improvements can stem from two sources: better temporal extrapolation or improved meteorological sensitivity. The consistently declining RSC with longer forecast horizons indicates that transfer learning predominantly enhances the former in my experiments.

The operational implications of this finding depend on the specific application. For general water management, such as optimising reservoir operations or planning water allocations, improved temporal extrapolation provides significant value. Better multi-day forecasts support these decisions, and my results demonstrate that transfer learning consistently delivers these improvements. For flood forecasting, however, this mechanism presents a limitation. Accurate flood prediction requires models that respond reliably to extreme precipitation forecasts. A model that excels at extrapolating existing streamflow patterns but fails to adequately capture the rainfall-runoff relationship during intense storms would be of limited use for flood warnings.

7.2.2 Architectural Dependencies

Transfer learning improves performance across all four architectures tested, but the magnitude and conditions for improvement vary. Simple MLP-based architectures—TSMixer and TiDE—respond to transfer learning even with limited pre-training data. In the regional experiment with 59 Kyrgyz basins, TSMixer improves from 0.85 to 0.88 NSE and TiDE from 0.84 to 0.87. The response to transfer learning varies by architectural complexity and dataset scale. EA-LSTM demonstrates modest but consistent improvements at shorter forecast horizons, even with regional pre-training, with NSE increasing from 0.96 to 0.97 at a 1-day horizon and from 0.84 to 0.87 at a 5-day horizon. TFT, however, shows minimal responsiveness to regional transfer learning across all horizons. Both architectures exhibit more substantial improvements when pre-trained on global datasets containing hundreds to thousands of basins.

All architectures converge to similar performance levels. This convergence reflects both the nature of the task and the experimental setup. Streamflow forecasting in mountainous catchments benefits from strong temporal autocorrelation, making accurate prediction achievable without architectural sophistication. Additionally, hyperparameter tuning on the limited target domain data constrains model sizes to 100k-1M parameters across all architectures. This constraint prevents complex models from fully utilising their advanced components.

A consistent pattern emerges when comparing the two target domains. While the benchmark models achieve similar baseline performance in both Tajikistan and Kyrgyzstan, the absolute improvement from transfer learning is consistently smaller for Kyrgyzstan. In the in-memory experiment, the median overall NSE improvement was only 0.01 in Kyrgyzstan, compared to 0.04 in Tajikistan. This difference is likely because the models for Kyrgyzstan are trained on a larger and potentially higher-quality local dataset (59 basins versus 15). This suggests the Kyrgyz benchmark models are already more robust and operate closer to the achievable performance ceiling for this specific task, leaving less remaining error for pre-training on external data to correct.

All models exhibit performance degradation with increasing forecast horizons. Since experiments employ perfect meteorological forecasts, this decline results entirely from the weakening autoregressive signal. At 1-day horizons, median NSE values approach 0.97; by 10-day horizons, they drop to 0.70-0.85. This pattern persists regardless of architecture or transfer learning strategy.

7.2.3 Source Domain Selection: Data Volume vs. Hydrological Relevance

The comparison between volume-based and similarity-based selection reveals no consistently superior approach. However, I observe that similarity-based selection tends to outperform volume-based selection at the 10-day forecast horizon across architectures and target domains. While I trained each model only once—rather than following best practice of multiple training runs to account for random parameter initialisation—the consistency of this pattern across four architectures, two experimental phases, and both data availability scenarios suggests it reflects more than random variation. This observation points toward an efficiency interpretation: selecting hydrologically similar catchments achieves comparable performance with up to 75% less data than the volume-based approach.

Two additional interpretations for the similar performance between strategies are: First, the volume-based approach indiscriminately includes all low-to-medium influence catchments, potentially introducing irrelevant hydrological behaviours that add noise rather than a useful signal. Second, given the relative simplicity of autoregressive streamflow forecasting in mountainous catchments, both approaches may already provide sufficient information for the models to reach their performance ceiling. In this context, my inadvertent exclusion of 91 Swiss basins from the volume-based selection in Experiment 2, Phase 2, likely had a minimal impact—adding these basins to the existing 6,690 would not significantly alter the comparison with the 1,850 similarity-selected basins.

7.3 Methodological Deep Dive: The Role of Experimental Design

7.3.1 A Case Study in Preprocessing: The Larger-than-RAM Experiment

Because the Caravan and CAMELS-CH datasets exceeded available memory, I replaced global Z-scoring with per-basin Z-scoring in Experiment 2, Phase 2. Relative to Phase 1, this change was associated with lower model skill and more catastrophic failures (NSE < 0), especially at longer forecast horizons.

The benchmark models in Phase 2 exhibited degraded performance compared to Phase 1, despite using identical architectures and data. In Tajikistan, median NSE dropped from 0.85 to 0.82, while in Kyrgyzstan, the decline was from 0.85 to 0.81. Beyond these aggregate metrics, the increase in catastrophic failures is noteworthy. Tajikistan experienced 12 such failures in Phase 2 compared to zero in Phase 1, while Kyrgyzstan saw an increase from 3 to 33 failures. These failures predominantly occurred at longer forecast horizons, suggesting that the models struggled when they needed to rely on meteorological forcing rather than autoregressive signals. The persistence model, which repeats the last observed value, experienced no catastrophic failures, indicating that the preprocessing choices specifically impaired the models' ability to extract information from meteorological inputs.

I attribute this performance degradation to four factors, with per-basin scaling of meteorological forcing as the primary cause. In Phase 1, I applied global Z-score normalisation to meteorological forcings, computing statistics across all basins and time steps. This approach preserves the relative differences between basins—a wet catchment remains distinguishably wetter than a dry catchment in the transformed feature space. In Phase 2, I switched to per-catchment normalisation to avoid loading all data into memory simultaneously. This transformation fundamentally altered the data structure: after per-basin normalisation, all basins appear similar in the transformed feature space, as each has been individually centred and scaled.

The second factor involves the Yeo-Johnson power transformation applied to meteorological forcings in Phase 2. This decision stemmed from concerns about catastrophic forgetting when training on sequential data chunks with potentially different distributions. However, this transformation fundamentally

alters how the model perceives hydrological relationships. Precipitation, for example, naturally exhibits strong positive skew with many zero or near-zero values and occasional extreme events—a distribution that carries information about hydrological processes. The Yeo-Johnson transform forces this distribution toward normality, compressing the extreme precipitation events that drive flood responses. This compression becomes particularly problematic when combined with the Mean Squared Error (MSE) loss function. By design, MSE heavily penalises large prediction errors. However, the power transform diminishes the numerical distinction between moderate and extreme events. As a result, the prediction errors for high-magnitude events are suppressed in the transformed space. This mutes the learning signal for these events, preventing the model from being adequately penalised for failing to predict them accurately.

The third factor concerns the Yeo-Johnson transformation applied to streamflow values in Phase 2, replacing the log transformation used in Phase 1. While both transformations compress extreme values, they do so in different ways. The log transformation preserves relative relationships—a doubling of flow always results in the same additive increase in log space. The Yeo-Johnson transform, however, optimises its parameter λ to force the data toward a normal distribution, potentially creating more severe distortions of the streamflow distribution.

The fourth factor involves the different hyperparameter sets used in each phase. Each phase employed hyperparameters tuned specifically for its preprocessing pipeline. While hyperparameter differences alone seem unlikely to cause such systematic degradation across all architectures, they may have amplified the effects of poor preprocessing choices. When hyperparameters are optimised for data that has already lost information through per-basin scaling and power transformations, the resulting model configurations may further limit the model's ability to extract meaningful patterns.

These four factors likely interact synergistically. Per-basin scaling removes inter-basin information, Yeo-Johnson transforms on both meteorological forcing and streamflow distort the remaining signals, and hyperparameters optimised for this compromised data may further degrade performance. The paradox that deep learning models fail catastrophically while the persistence model—which relies purely on autoregressive signals—never fails, suggests a more complex failure mode. Rather than simply defaulting to autoregressive behaviour, the models appear to be attempting to integrate both corrupted meteorological and distorted streamflow signals, producing predictions worse than if they ignored meteorology entirely. This indicates that the preprocessing choices can actively disrupt the learning process.

The decision to apply the Yeo-Johnson transformation arose from concerns about distribution shifts during sequential training on data chunks. However, this concern was unfounded. When randomly sampling 1,500 basins from a population of 6,800, the Law of Large Numbers ensures that each chunk will have similar statistical properties to the whole. The Central Limit Theorem further ensures that sample statistics will converge to their corresponding population parameters. With chunks of 1,500 basins, distribution shifts between chunks would be negligible. The Yeo-Johnson transform was unnecessary and actively harmful, as it added a complex nonlinear transformation that obscured hydrological signals without solving any real problem.

Transfer learning demonstrated robustness. In Tajikistan, models with transfer learning achieved median NSE values of 0.89, matching the performance of Phase 1 despite compromised preprocessing. Transfer learning reduced catastrophic failures from 12 to 1 and 2 cases for volume-based and similarity-based transfer learning, respectively. In Kyrgyzstan, improvements were similarly substantial, with NSE increasing from 0.81 to 0.86 and catastrophic failures dropping from 33 to 17 and 20. This robustness suggests that pre-trained models develop internal representations that are less sensitive to preprocessing choices. Exposure to diverse source domain data may teach models to extract useful signals even from poorly transformed inputs. However, if transfer learning primarily improves autoregressive skill rather

than meteorological sensitivity, its value for operational flood forecasting remains limited. Accurate flood prediction requires models that respond appropriately to extreme precipitation forecasts, not merely those that excel at temporal extrapolation.

Having identified the problems with per-basin preprocessing, several alternative strategies emerge for handling datasets larger than RAM. Instead of computing statistics separately for each basin, global statistics could be computed on a carefully selected subset of basins that fit in memory. Random sampling would select a subset of basins (500-1000), with statistics approximating population parameters for sufficiently large samples. Similarity-based sampling would select basins similar to the target domain using hydrological signatures or static attributes. Stratified sampling would sample across different hydrological regimes, climatic zones, or geographic regions to ensure representation of hydrological diversity. The implementation would involve selecting the representative subset, loading only these basins into memory to compute global statistics, then applying these statistics to transform all data during training while processing basins in chunks. This approach preserves inter-basin information while remaining computationally feasible.

My experience highlights a misconception in academic critiques of deep learning. The traditional concern focuses on interpretability—understanding why models make specific predictions. However, my results demonstrate that the operational challenge differs: understanding how to make models learn effectively. The "black box" nature matters operationally not because model decisions cannot be interpreted, but because the field lacks understanding of how preprocessing choices affect learning dynamics. The Phase 2 results exemplify this: a seemingly reasonable preprocessing pipeline catastrophically degraded performance. Without understanding how models extract information from hydrological data, such failures are difficult to predict or prevent. Operational deployment requires an understanding of how models respond to input transformations. The question is not "why does this model predict high flow?" but rather "how should the data pipeline be structured so the model can learn to predict high flow?"

7.3.2 Other Key Design Choices and Their Implications

I implemented transfer learning by reducing the learning rate by a factor of 25 during fine-tuning while allowing all model weights to update. More sophisticated strategies exist, such as selective layer freezing, which only allows parts of the model to change, progressive unfreezing, which gradually allows deeper layers to change as model training progresses; however, these would have introduced another experimental dimension to the comparison across four architectures, three forecast horizons, and two data availability scenarios. My approach is model-agnostic, ensuring a fair comparison as it does not require deciding which layers are allowed to change for each model.

No unified framework exists for quantifying human influence in catchments. I developed the Human Influence Index, which combines eight static attributes weighted by their assumed importance. Alternative approaches include counting the number of reservoirs or computing the degree of regulation as cumulative reservoir storage relative to mean annual flow (Ouyang et al., 2021). All methods share two limitations. First, validation is limited to the visual inspection of hydrographs and assertions that a basin "looks natural" or "appears regulated." Second, static attributes cannot capture temporal dynamics. A catchment's streamflow might be unregulated for decades before dam construction fundamentally alters its behaviour, yet static metrics treat the entire record uniformly.

The HII classification required selecting thresholds for which no guidelines exist. I chose the 30th and 75th percentiles to separate low, medium, and high influence categories. These thresholds directly determined which of the 16,000+ global catchments were included in the "volume-based" source domain for

transfer learning. Different thresholds would have yielded different source domains.

7.4 Operational Relevance, Limitations, and Future Directions

This work demonstrates that transfer learning offers an immediately actionable alternative to traditional hydrological capacity building. Rather than waiting several years to collect sufficient local data or investing in costly monitoring infrastructure, countries can achieve measurable improvements in forecasts today by utilizing existing regional and global datasets. The regional transfer learning experiment provides quantitative evidence that can directly support policy decisions regarding transboundary data sharing agreements.

Critically, my results reveal that the benefits of transfer learning are inversely proportional to data availability in the target domain—the improvement in Tajikistan (15 basins) was larger than in Kyrgyzstan (59 basins) across all experiments. While this pattern may partially reflect differences in data quality between the two countries, the consistency of this inverse relationship strongly suggests that data-scarce regions are likely to benefit most from transfer learning approaches. This implies that countries with limited financial and technical resources—which typically correlate with smaller monitoring networks—would particularly benefit from this methodology. By providing a pathway to improved water forecasting that bypasses traditional infrastructure requirements, this work offers particular value to regions that might otherwise be left behind in hydrological modernization efforts due to economic or technical constraints.

However, while this thesis demonstrates clear operational potential, the experimental design contains several idealisations that must be distinguished from the requirements of a true operational system. Understanding these limitations is important for contextualizing the results and identifying pathways for future operational implementation.

Scientific research requires a holdout set for unbiased model performance reporting; therefore, in my experiments, I dedicate the most recent 25% of each basin's record exclusively to testing. This convention stands in direct opposition to operational goals. Operationally, the aim is to maximise forecast skill today, so models are trained on essentially all available data (keeping only the validation set to avoid overfitting) rather than permanently dedicating a quarter of the record to testing. The evaluation dilemma is what remains: testing on the latest years improves relevance but withholds exactly the data that best captures current hydrology; testing on earlier years avoids that cost but evaluates the model on a climate regime that may no longer be representative—especially in basins changing rapidly due to climate, infrastructure, or land-use shifts.

I implement a per-basin splitting strategy which divides the data for each basin into a training set (the first 50% of the record), a validation set (the next 25%), and a test set (the final 25%). This approach ensures all basins contribute to model training, thereby maximising the spatial coverage of the dataset. An alternative—splitting by fixed calendar dates across all basins—was not implemented because it might exclude certain catchments from the training set. For instance, a basin whose record falls entirely outside the designated training window would contribute data only to validation or testing, meaning the model would never see it during training.

This methodological choice, however, introduces potential data leakage through spatial correlation. When each basin is split independently by percentage, basins with different temporal coverage contribute different periods to the training and test sets. Consider two neighbouring basins that experience the same regional weather patterns: if Basin A spans 1990-2010 and Basin B spans 2000-2020, then Basin A's test set (2005-2010) overlaps temporally with Basin B's training set (2000-2010). Any extreme event affecting

both basins during 2005-2010 appears in the training data via Basin B, allowing the model to learn its characteristics. When evaluated on Basin A's test set, the model encounters this supposedly "unseen" event—but it has already learned about it indirectly through the spatially correlated basin. This violates the fundamental assumption that test data should be independent from training data. I do not quantify the magnitude of this effect in my thesis. Consequently, the performance estimates reported should be interpreted as potentially optimistic upper bounds.

The way I have set up the models requires complete streamflow observations: they cannot handle missing data. Operational forecasting in Central Asia must contend with gaps due to manual station recording and data transmission delays. I address short gaps through forward-filling imputation up to five consecutive days, but this approach becomes hydrologically implausible for longer periods. Nearing et al. (2021a) demonstrate that adding binary flags to indicate data availability allows models to learn distinct strategies for observed versus imputed values. This approach would enable simple constant-value imputation for extended gaps: missing streamflow values could be replaced with a fixed value (e.g., catchment mean flow) accompanied by a binary flag—0 for observed data, 1 for imputed data (Nearing et al. (2021a); as implemented in Ruparell et al. (2025)). The model learns to process flagged missing data differently from observations, eliminating the need for realistic imputation. An alternative implementation could use the model's previous forecasts as imputation values, still accompanied by binary flags to distinguish them from observations.

The binary flags enable models to distinguish observed from imputed data; this capability may address a limitation revealed by my transfer learning results. Nearing et al. (2021a) established that autoregression is essential for optimal operational performance, while Ryd and Nearing (2025) demonstrated that transfer learning improves flood forecasting in non-autoregressive models. My results indicate that transfer learning improves autoregressive models across various general performance metrics. However, the RSC analysis reveals that these improvements stem primarily from enhanced temporal pattern extrapolation rather than improved meteorological sensitivity. For flood forecasting, this presents a critical limitation: accurate flood prediction requires models that respond to extreme precipitation events, not models that excel at streamflow extrapolation. Therefore, while transfer learning improves autoregressive model performance by conventional metrics, these gains may not translate to improved flood forecasting capability.

To address this limitation, I propose a strategic missing data augmentation approach during training that could force models to develop stronger dependencies on meteorological forcing. By artificially introducing missing streamflow data beyond what occurs operationally, I can effectively "handicap" the autoregressive pathway, forcing the model to extract more information from the meteorological forcing. This augmentation strategy would involve sampling gap characteristics—location, duration, and frequency—from probability distributions, with higher gap density concentrated near the forecast date to mimic operational conditions where recent observations are often unavailable. The binary flag system would treat all missing data identically—whether genuinely missing or artificially removed—teaching the model to rely on meteorological forcing whenever streamflow observations are unavailable. Future work should systematically evaluate whether this forced meteorological dependency enhances model sensitivity to extreme precipitation events and improves flood peak prediction accuracy.

My experiments assume perfect meteorological forecasts, utilising ERA5-Land reanalysis data for the future period. This idealisation diverges from operational reality, where meteorological forecast uncertainty compounds hydrological prediction errors. However, this controlled approach serves a specific purpose: it isolates the hydrological modelling component from meteorological uncertainty, enabling direct attribution of performance differences to transfer learning. Both benchmark and transfer learning models receive identical perfect meteorological inputs, ensuring fair comparison. I hypothesise that the relative benefits persist regardless of forecast quality—if transfer learning yields 10% NSE improvement

with perfect forcing, similar relative gains should occur when both model types face identical uncertain meteorological forecasts. While this hypothesis remains untested in my thesis, it represents a reasonable assumption given that meteorological uncertainty would affect both model types equally. Future work should validate this by evaluating transfer learning benefits using operational meteorological forecasts of varying skill levels, potentially leveraging archived forecast datasets like TIGGE[3] (THORPEX Interactive Grand Global Ensemble) that preserve the uncertainty characteristics of meteorological forecasts.

Beyond these methodological considerations, the evaluation framework itself limits the assessment of operational value. I deliberately employed general metrics—NSE, KGE, RMSE, and MAE—to first establish if transfer learning could broadly improve forecasting. This was a practical necessity: if the models showed no general improvement, application-specific evaluation would be moot. The results confirm that transfer learning does enhance these general metrics, providing a basis for more targeted investigations.

However, operational forecasting serves diverse purposes, from flood warnings that require accurate peak flow timing to hydropower operations that depend on reliable volume forecasts. General metrics, such as NSE and KGE, obscure these distinctions by measuring overall fit rather than performance on critical events. Similarly, RMSE and MAE quantify error magnitude but do not distinguish between underestimation and overestimation, a crucial difference when a missed flood is far more consequential than a false alarm. While my RSC analysis partially addresses this by revealing the source of the improvements, a systematic evaluation of application-specific performance is needed.

Furthermore, operational decisions require the quantification of uncertainty—such as confidence intervals or exceedance probabilities—to assess risk. This study's deterministic approach establishes that transfer learning improves point forecasts but cannot inform how confidently a decision-maker should act on them. Exploring whether transfer learning can also yield more reliable uncertainty estimates remains an open question. Future work could directly address this by enabling quantile regression, for example, by replacing the standard MSE loss function with the pinball loss to optimise for specific quantiles. I hypothesise that transfer learning would improve the accuracy of these quantile predictions and also narrow the uncertainty bands. As Beven (2016) notes, epistemic uncertainties "should be reducible by further experimentation or observation," with an expectation of moving "towards more aleatory residual error in the future." Models exposed to diverse hydrological behaviours across thousands of source domain catchments represent precisely this type of expanded observation, which should yield tighter, more confident predictions—though this remains to be tested.

[3] https://www.ecmwf.int/en/forecasts/dataset/thorpex-interactive-grand-global-ensemble

8 Conclusion

My thesis demonstrates that transfer learning provides consistent, albeit incremental, benefits for operational streamflow forecasting in Central Asian mountainous catchments, with median Nash-Sutcliffe Efficiency improvements ranging from 0.01 to 0.07 across different data availability scenarios. The primary finding reveals that these improvements stem not from enhanced understanding of rainfall-runoff processes, but rather from strengthened temporal extrapolation of the autoregressive streamflow signal—a mechanism evidenced by the systematic decline in Remaining Skill Captured from 35% at 1-day horizons to 10-15% at 10-day horizons. The relatively modest magnitude of these gains, combined with the convergence of all four tested architectures (EA-LSTM, TFT, TSMixer, TiDE) to similar performance levels after global transfer learning, suggests that strong temporal autocorrelation and seasonal patterns in these mountainous basins create a forecasting context where deep learning models can already extract most learnable information from local target domain data alone, thus operating near a performance ceiling. The comparison between volume-based (up to 6,690 catchments) and similarity-based (1,850 catchments) source domain selection strategies revealed no consistently superior approach; however, the latter achieved comparable performance with 81% less data, highlighting the potential for computational efficiency without sacrificing accuracy. While the documented improvements benefit general water management applications, such as reservoir operations and water allocation planning, the reliance on autoregressive patterns rather than meteorological sensitivity presents a significant limitation for flood forecasting. This is because flood forecasting requires robust responsiveness to extreme precipitation events, rather than extrapolating temporal patterns. This work demonstrates that, although transfer learning can successfully scale to larger-than-RAM datasets and reduce catastrophic model failures by up to 90%, careful attention to data preprocessing choices is essential for realizing these benefits in practice. The finding that transfer learning provides the greatest benefits to the most data-limited settings underscores its potential as an equitable solution for global water security, ensuring that financial and technical limitations do not prevent access to improved hydrological forecasting capabilities.

Future research should explore forcing models to develop stronger meteorological dependencies through strategic missing data augmentation, implement uncertainty quantification using quantile regression, and evaluate the benefits of transfer learning under realistic meteorological forecast uncertainty to bridge the gap between these scientific advances and operational deployment in water resources management.

References

Nans Addor, Andrew J Newman, Naoki Mizukami, and Martyn P Clark. The camels data set: catchment attributes and meteorology for large-sample studies. *Hydrology and Earth System Sciences*, 21(10): 5293–5313, 2017.

Nans Addor, Hong X Do, Camila Alvarez-Garreton, Gemma Coxon, Keirnan Fowler, and Pablo A Mendoza. Large-sample hydrology: recent progress, guidelines for new datasets and grand challenges. *Hydrological Sciences Journal*, 65(5):712–725, 2020.

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2623–2631, 2019.

Camila Alvarez-Garreton, Pablo A Mendoza, Juan Pablo Boisier, Nans Addor, Mauricio Galleguillos, Mauricio Zambrano-Bigiarini, Antonio Lara, Cristóbal Puelma, Gonzalo Cortes, Rene Garreaud, et al. The camels-cl dataset: catchment attributes and meteorology for large sample studies–chile dataset. *Hydrology and Earth System Sciences*, 22(11):5817–5846, 2018.

Sam Anderson and Valentina Radić. Evaluation and interpretation of convolutional long short-term memory networks for regional hydrological modelling. *Hydrology and Earth System Sciences*, 26(3): 795–825, 2022.

David Arthur and Sergei Vassilvitskii. k-means++: The advantages of careful seeding. Technical report, Stanford, 2006.

James Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. Algorithms for hyper-parameter optimization. *Advances in neural information processing systems*, 24, 2011.

Thomas Bernauer and Tobias Siegfried. Climate change and international water conflict in central asia. *Journal of Peace Research*, 49(1):227–239, 2012.

Keith Beven. Facets of uncertainty: epistemic uncertainty, non-stationarity, likelihood, hypothesis testing, and communication. *Hydrological Sciences Journal*, 61(9):1652–1665, 2016.

Keith J Beven. Uniqueness of place and process representations in hydrological modelling. *Hydrology and earth system sciences*, 4(2):203–213, 2000.

Si-An Chen, Chun-Liang Li, Nate Yoder, Sercan O Arik, and Tomas Pfister. Tsmixer: An all-mlp architecture for time series forecasting. *arXiv preprint arXiv:2303.06053*, 2023.

Franziska Clerc-Schwarzenbach, Giovanni Selleri, Mattia Neri, Elena Toth, Ilja van Meerveld, and Jan Seibert. Large-sample hydrology–a few camels or a whole caravan? *Hydrology and Earth System Sciences*, 28(17):4219–4237, 2024.

Abhimanyu Das, Weihao Kong, Andrew Leach, Shaan Mathur, Rajat Sen, and Rose Yu. Long-term forecasting with tide: Time-series dense encoder. *arXiv preprint arXiv:2304.08424*, 2023.

Dapeng Feng, Kuai Fang, and Chaopeng Shen. Enhancing streamflow forecast and extracting insights using long-short term memory networks with data integration at continental scales. *Water Resources Research*, 56(9):e2019WR026793, 2020.

Dapeng Feng, Jiangtao Liu, Kathryn Lawson, and Chaopeng Shen. Differentiable, learnable, regionalized process-based models with multiphysical outputs can approach state-of-the-art hydrologic prediction accuracy. *Water Resources Research*, 58(10):e2022WR032404, 2022.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*, volume 1. MIT press Cambridge, 2016.

Alex Graves, Navdeep Jaitly, and Abdel-rahman Mohamed. Hybrid speech recognition with deep bidirectional lstm. In *2013 IEEE workshop on automatic speech recognition and understanding*, pages 273–278. IEEE, 2013.

Miao He, Shanhu Jiang, Liliang Ren, Hao Cui, Shuping Du, Yongwei Zhu, Tianling Qin, Xiaoli Yang, Xiuqin Fang, and Chong-Yu Xu. Exploring the performance and interpretability of hybrid hydrologic model coupling physical mechanisms and deep learning. *Journal of Hydrology*, 649:132440, 2025.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

Marvin Höge, Andreas Scheidegger, Marco Baity-Jesi, Carlo Albert, and Fabrizio Fenicia. Improving hydrologic models for predictions and process understanding using neural odes. *Hydrology and Earth System Sciences*, 26(19):5085–5102, 2022.

Marvin Höge, Martina Kauzlaric, Rosi Siber, Ursula Schönenberger, Pascal Horton, Jan Schwanbeck, Marius Günter Floriancic, Daniel Viviroli, Sibylle Wilhelm, Anna E Sikorska-Senoner, et al. Camels-ch: hydro-meteorological time series and landscape attributes for 331 catchments in hydrologic switzerland. *Earth System Science Data Discussions*, 2023:1–46, 2023.

Markus Hrachowitz, HHG Savenije, Günter Blöschl, Jeffrey J McDonnell, Murugesu Sivapalan, JW Pomeroy, Berit Arheimer, Theresa Blume, MP Clark, Uwe Ehret, et al. A decade of predictions in ungauged basins (pub)—a review. *Hydrological sciences journal*, 58(6):1198–1255, 2013.

Hydrosolutions Ltd. Sapphire – central asia, 2023. URL https://www.hydrosolutions.ch/projects/sapphire-central-asia. Accessed: 2025-04-11.

W Walter Immerzeel, Arthur F Lutz, Marcos Andrade, A Bahl, Hester Biemans, Tobias Bolch, Sam Hyde, S Brumby, BJ Davies, AC Elmore, et al. Importance and vulnerability of the world's water towers. *Nature*, 577(7790):364–369, 2020.

International Association of Hydrological Sciences. About IAHS, 2025. URL https://iahs.info/About-IAHS/about-iahs/. Accessed: 2025-03-27.

Yegane Khoshkalam, Alain N Rousseau, Farshid Rahmani, Chaopeng Shen, and Kian Abbasnezhadi. Applying transfer learning techniques to enhance the accuracy of streamflow prediction produced by long short-term memory networks with data integration. *Journal of Hydrology*, 622:129682, 2023.

Taesung Kim, Jinhee Kim, Yunwon Tae, Cheonbok Park, Jang-Ho Choi, and Jaegul Choo. Reversible instance normalization for accurate time-series forecasting against distribution shift. In *International conference on learning representations*, 2021.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.

Wouter JM Knoben, Ross A Woods, and Jim E Freer. A quantitative hydrological climate classification evaluated with independent streamflow data. *Water Resources Research*, 54(7):5088–5109, 2018.

Frederik Kratzert, Daniel Klotz, Claire Brenner, Karsten Schulz, and Mathew Herrnegger. Rainfall–runoff modelling using long short-term memory (lstm) networks. *Hydrology and Earth System Sciences*, 22 (11):6005–6022, 2018.

Frederik Kratzert, Daniel Klotz, Mathew Herrnegger, Alden K Sampson, Sepp Hochreiter, and Grey S Nearing. Toward improved predictions in ungauged basins: Exploiting the power of machine learning. *Water Resources Research*, 55(12):11344–11354, 2019a.

Frederik Kratzert, Daniel Klotz, Guy Shalev, Günter Klambauer, Sepp Hochreiter, and Grey Nearing. Benchmarking a catchment-aware long short-term memory network (lstm) for large-scale hydrological modeling. *Hydrol. Earth Syst. Sci. Discuss*, 2019:1–32, 2019b.

Frederik Kratzert, Daniel Klotz, Guy Shalev, Günter Klambauer, Sepp Hochreiter, and Grey Nearing. 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, 2019c.

Frederik Kratzert, Grey Nearing, Nans Addor, Tyler Erickson, Martin Gauch, Oren Gilon, Lukas Gudmundsson, Avinatan Hassidim, Daniel Klotz, Sella Nevo, et al. Caravan-a global community dataset for large-sample hydrology. *Scientific Data*, 10(1):61, 2023.

Frederik Kratzert, Martin Gauch, Daniel Klotz, and Grey Nearing. Hess opinions: Never train an lstm on a single basin. *Hydrology and Earth system sciences discussions*, 2024:1–19, 2024.

Bryan Lim, Sercan Ö Arık, Nicolas Loeff, and Tomas Pfister. Temporal fusion transformers for interpretable multi-horizon time series forecasting. *International Journal of Forecasting*, 37(4):1748–1764, 2021.

Yongen Lin, Dagang Wang, Tao Jiang, and Aiqing Kang. Assessing objective functions in streamflow prediction model training based on the naïve method. *Water*, 16(5):777, 2024.

Simon Linke, Bernhard Lehner, Camille Ouellet Dallaire, Joseph Ariwi, Günther Grill, Mira Anand, Penny Beames, Vicente Burchard-Levine, Sally Maxwell, Hana Moidu, et al. Global hydro-environmental sub-basin and river reach characteristics at high spatial resolution. *Scientific data*, 6(1):283, 2019.

Jiangtao Liu, Chaopeng Shen, Fearghal O'Donncha, Yalan Song, Wei Zhi, Hylke E Beck, Tadd Bindas, Nicholas Kraabel, and Kathryn Lawson. From rnns to transformers: benchmarking deep learning architectures for hydrologic prediction. *EGUsphere*, 2025:1–21, 2025.

Kai Ma, Dapeng Feng, Kathryn Lawson, Wen-Ping Tsai, Chuan Liang, Xiaorong Huang, Ashutosh Sharma, and Chaopeng Shen. Transferring hydrologic data across continents–leveraging data-rich regions to improve hydrologic prediction in data-sparse regions. *Water Resources Research*, 57(5): e2020WR028600, 2021.

Beatrice Marti, Andrey Yakovlev, Dirk Nikolaus Karger, Silvan Ragettli, Aidar Zhumabaev, Abdul Wakil Wakil, and Tobias Siegfried. Ca-discharge: Geo-located discharge time series for mountainous rivers in central asia. *Scientific Data*, 10(1):579, 2023.

Met Office. *Cartopy: a cartographic python library with a Matplotlib interface*. Exeter, Devon, 2010 - 2015. URL https://scitools.org.uk/cartopy.

Alberto Montanari, Gordon Young, Hubert HG Savenije, Denis Hughes, Thorsten Wagener, Liliang L Ren, Demetris Koutsoyiannis, Christophe Cudennec, Elena Toth, Stefiani Grimaldi, et al. "panta rhei—everything flows": change in hydrology and society—the iahs scientific decade 2013–2022. *Hydrological sciences journal*, 58(6):1256–1275, 2013.

Joaquín Muñoz-Sabater, Emanuel Dutra, Anna Agustí-Panareda, Clément Albergel, Gabriele Arduini, Gianpaolo Balsamo, Souhail Boussetta, Margarita Choulga, Shaun Harrigan, Hans Hersbach, et al. Era5-land: A state-of-the-art global reanalysis dataset for land applications. *Earth system science data*, 13(9):4349–4383, 2021.

Grey S Nearing, Daniel Klotz, Alden Keefe Sampson, Frederik Kratzert, Martin Gauch, Jonathan M Frame, Guy Shalev, and Sella Nevo. Data assimilation and autoregression for using near-real-time streamflow observations in long short-term memory networks. *Hydrology and earth system sciences discussions*, 2021:1–25, 2021a.

Grey S Nearing, Frederik Kratzert, Alden Keefe Sampson, Craig S Pelissier, Daniel Klotz, Jonathan M Frame, Cristina Prieto, and Hoshin V Gupta. What role does hydrological science play in the age of machine learning? *Water Resources Research*, 57(3):e2020WR028091, 2021b.

Andrew J Newman, Martyn P Clark, Kevin Sampson, Andrew Wood, Lauren E Hay, Andy Bock, Roland J Viger, David Blodgett, Levi Brekke, JR Arnold, et al. Development of a large-sample watershed-scale hydrometeorological data set for the contiguous usa: data set characteristics and assessment of regional variability in hydrologic model performance. *Hydrology and Earth System Sciences*, 19(1):209–223, 2015.

Wenyu Ouyang, Kathryn Lawson, Dapeng Feng, Lei Ye, Chi Zhang, and Chaopeng Shen. Continental-scale streamflow modeling of basins with reservoirs: Towards a coherent deep-learning-based strategy. *Journal of Hydrology*, 599:126455, 2021.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12: 2825–2830, 2011.

François Petitjean, Alain Ketterlin, and Pierre Gançarski. A global averaging method for dynamic time warping, with applications to clustering. *Pattern recognition*, 44(3):678–693, 2011.

Cristina Prieto, Nataliya Le Vine, Dmitri Kavetski, Eduardo García, and Raúl Medina. Flow prediction in ungauged catchments using probabilistic random forests regionalization and new statistical adequacy tests. *Water Resources Research*, 55(5):4364–4392, 2019.

RunPod. RunPod: Cloud GPU Computing Platform, 2025. URL https://www.runpod.io/. Cloud computing platform providing on-demand GPU instances for AI and machine learning workloads.

Karan Ruparell, Robert J Marks, Andy Wood, Kieran MR Hunt, Hannah L Cloke, Christel Prudhomme, Florian Pappenberger, and Matthew Chantry. Hydra-lstm: A semi-shared machine learning architecture for prediction across watersheds. *Artificial Intelligence for the Earth Systems*, 2025.

Emil Ryd and Grey Nearing. Fine flood forecasts: Incorporating local data into global models through fine-tuning. *arXiv preprint arXiv:2504.12559*, 2025.

Hiroaki Sakoe and Seibi Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE transactions on acoustics, speech, and signal processing*, 26(1):43–49, 1978.

Tobias Siegfried, Beatrice Marti, Adrian Kreiner, and Aidar Zhumabaev. *Modeling of Hydrological Systems in Semi-Arid Central Asia*. hydrosolutions GmbH, 2024.q1 edition, 2024. URL https://hydrosolutions.github.io/caham_book/. Accessed: 2025-04-11.

Diego F Silva and Gustavo EAPA Batista. Speeding up all-pairwise dynamic time warping matrix calculation. In *Proceedings of the 2016 SIAM International Conference on Data Mining*, pages 837–845. SIAM, 2016.

Murugesu Sivapalan, K Takeuchi, SW Franks, VK Gupta, H Karambiri, V Lakshmi, X Liang, JJ McDonnell, Eduardo Mario Mendiondo, PE O'connell, et al. Iahs decade on predictions in ungauged basins (pub), 2003–2012: Shaping an exciting future for the hydrological sciences. *Hydrological sciences journal*, 48(6):857–880, 2003.

Swiss Agency for Development and Cooperation (SDC). Sapphire – supporting a regional mechanism for sustainable water resources management in central asia, 2022. URL https://www.eda.admin.ch/deza/de/home/aktivitaeten_projekte/projekte-fokus/projektdatenbank.filterResults.html/content/dezaprojects/SDC/en/2022/7F11001/phase1?oldPagePath=/content/deza/de/home/projekte/projekte.html. Accessed: 2025-04-11.

Chuanqi Tan, Fuchun Sun, Tao Kong, Wenchang Zhang, Chao Yang, and Chunfang Liu. A survey on deep transfer learning. In *Artificial Neural Networks and Machine Learning–ICANN 2018: 27th International Conference on Artificial Neural Networks, Rhodes, Greece, October 4-7, 2018, Proceedings, Part III 27*, pages 270–279. Springer, 2018.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Sanford Weisberg. Yeo-johnson power transformations. *Department of Applied Statistics, University of Minnesota. Retrieved June*, 1:2003, 2001.

Mingyue Yang and Francisco Olivera. Classification of watersheds in the conterminous united states using shape-based time-series clustering and random forests. *Journal of Hydrology*, 620:129409, 2023.

Hanlin Yin, Xiuwei Zhang, Fandu Wang, Yanning Zhang, Runliang Xia, and Jin Jin. Rainfall-runoff modeling using lstm-based multi-state-vector sequence-to-sequence model. *Journal of Hydrology*, 598: 126378, 2021.

Hanlin Yin, Zilong Guo, Xiuwei Zhang, Jiaojiao Chen, and Yanning Zhang. Rr-former: Rainfall-runoff modeling based on transformer. *Journal of Hydrology*, 609:127781, 2022.

A Goodness of Fit Metrics

A.1 Nash-Sutcliffe Efficiency (NSE)

$$\text{NSE} = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

A.2 Kling-Gupta Efficiency (KGE)

$$\text{KGE} = 1 - \sqrt{(r-1)^2 + (\beta-1)^2 + (\alpha-1)^2}$$

where:

- r is the Pearson correlation coefficient between y and $\hat{y}$
- $\beta = \frac{\mu_{\hat{y}}}{\mu_y}$ is the bias ratio
- $\alpha = \frac{\sigma_{\hat{y}}}{\sigma_y}$ is the variability ratio

A.3 Root Mean Square Error (RMSE)

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

A.4 Mean Absolute Error (MAE)

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

A.5 Variable Definitions

- y_i: observed value at time step i
- $\hat{y}_i$: predicted value at time step i
- $\bar{y}$: mean of observed values
- n: total number of observations
- $\mu_y, \mu_{\hat{y}}$: mean of observed and predicted values, respectively
- $\sigma_y, \sigma_{\hat{y}}$: standard deviation of observed and predicted values, respectively

B Human Influence Index Classification

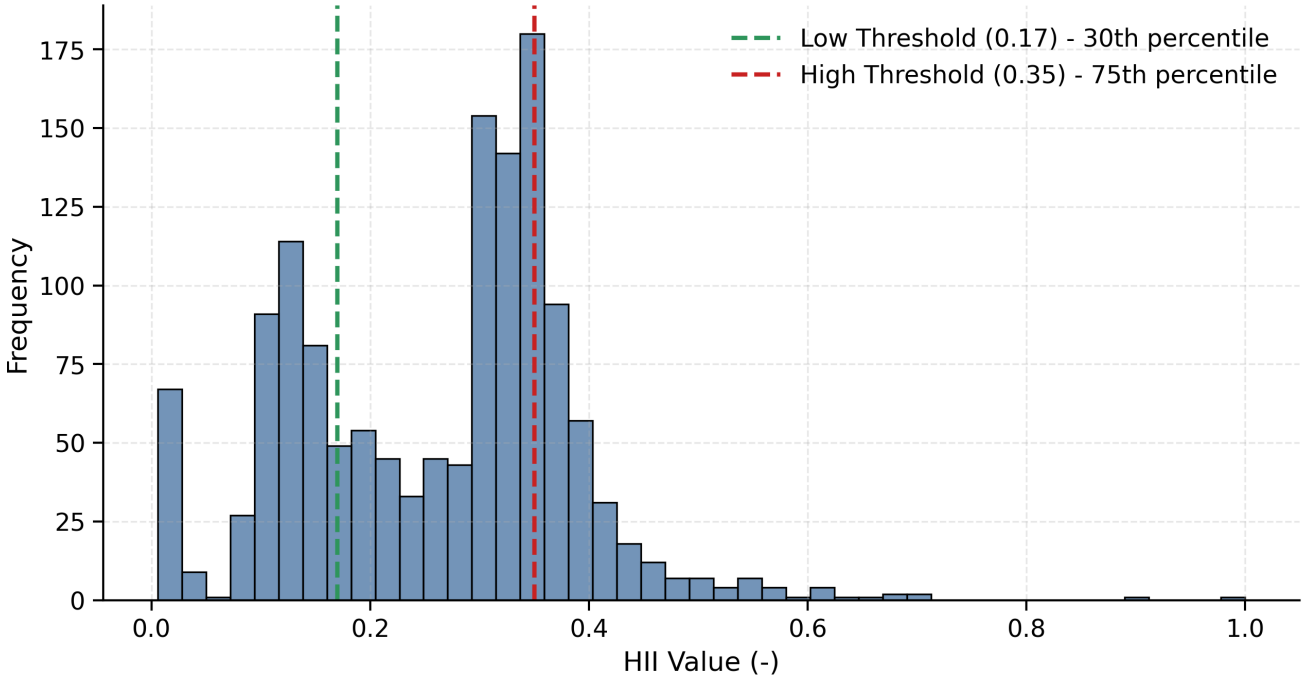

Figure B1: Histogram of the Human Influence Index (HII) values for the catchments in Chile, USA and Switzerland. The dashed vertical lines represent the low threshold at the 30th percentile (HII = 0.17) and the high threshold at the 75th percentile (HII = 0.35).

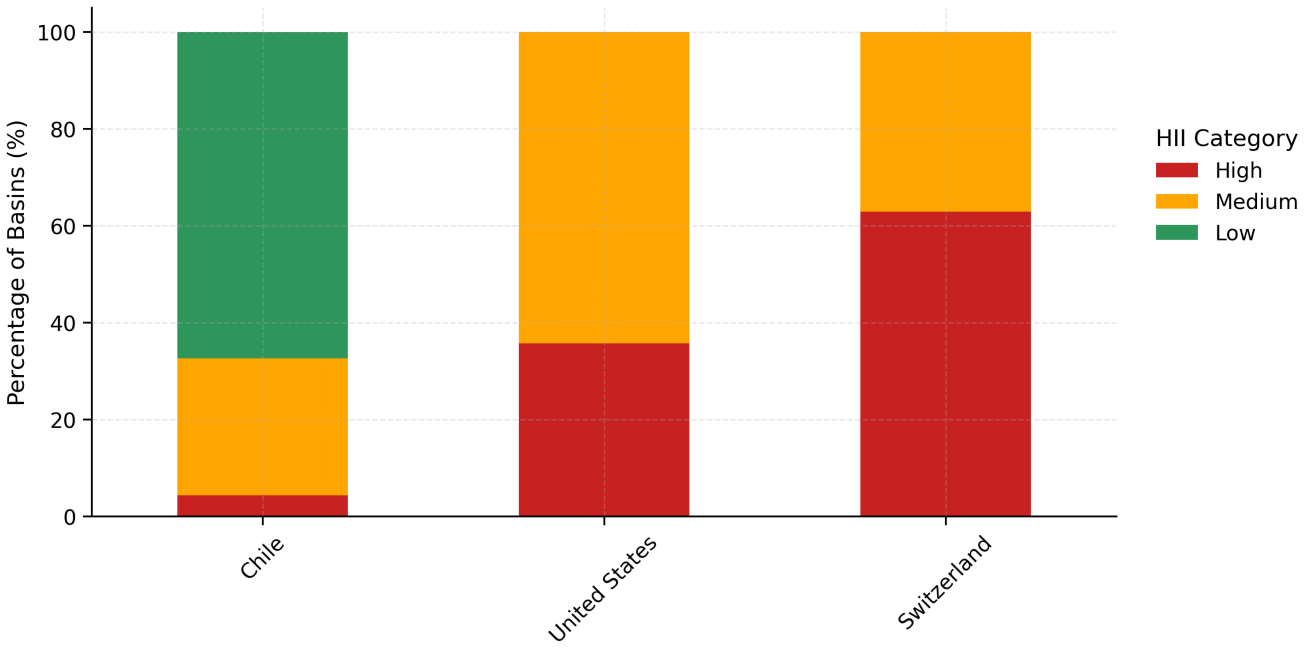

Figure B2: Distribution of Human Influence Index (HII) categories across Chile, USA, and Switzerland. Stacked bars show the relative proportion of Low, Medium, and High influence catchments, normalised to 100% for each country. Countries are ordered by the increasing proportion of high-influence catchments.

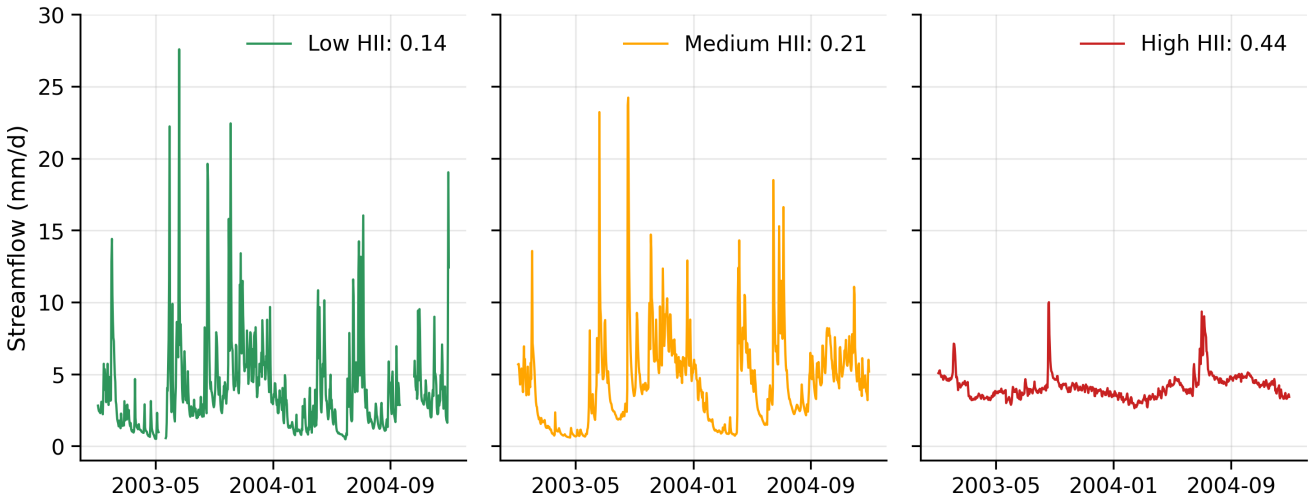

Figure B3: Daily streamflow time series (2003–2004) for three Chilean catchments representing each HII category: Low (HII = 0.14), Medium (HII = 0.21), and High (HII = 0.44). Values shown in mm/day.

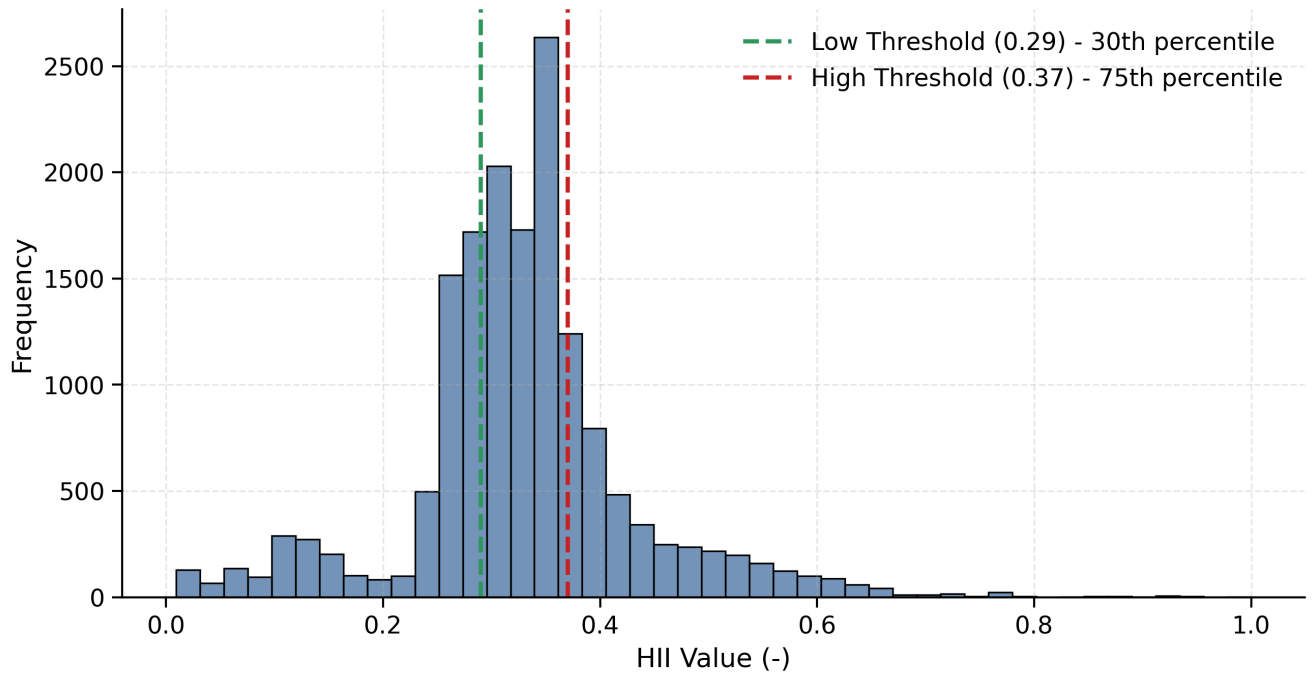

Figure B4: Histogram of the Human Influence Index (HII) values for 16,038 catchments in the Caravan dataset. The dashed vertical lines represent the low threshold at the 30th percentile (HII = 0.29) and the high threshold at the 75th percentile (HII = 0.37).

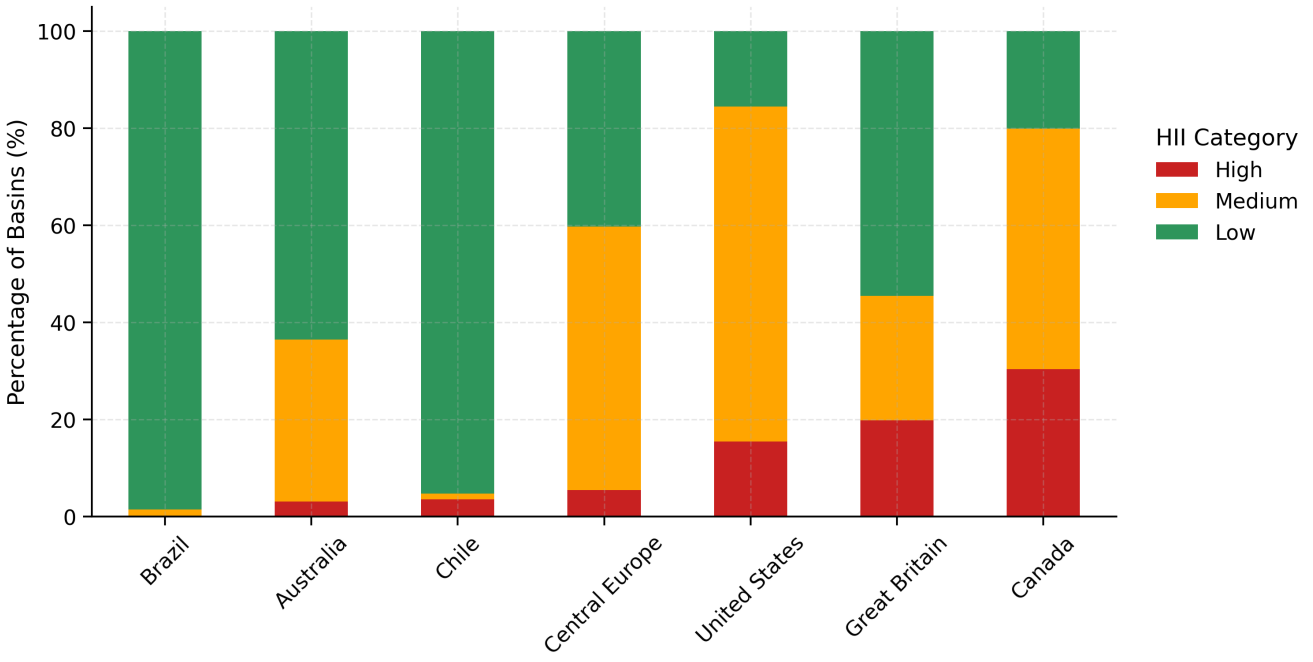

Figure B5: Distribution of Human Influence Index (HII) categories across all regions in the Caravan dataset. Stacked bars show the relative proportion of Low, Medium, and High influence catchments, normalised to 100% for each region. Regions are ordered by increasing proportion of High influence catchments.

C Catchment Similarity

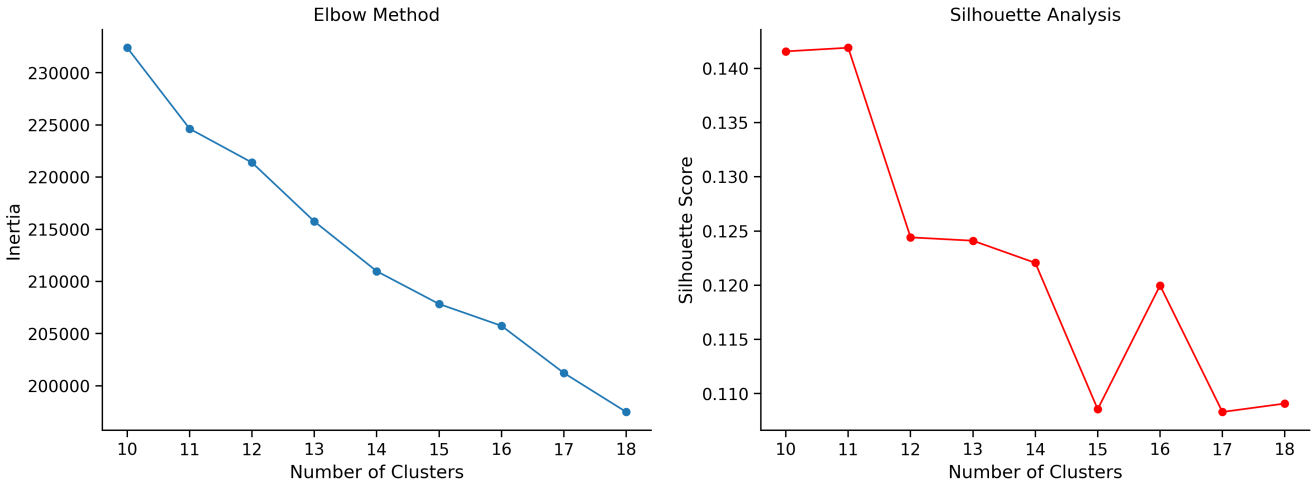

Figure C6: Elbow plot for determining optimal number of clusters in hydrograph-based catchment clustering. The plot shows inertia (within-cluster variance) against cluster counts ranging from 10 to 20, with silhouette scores also displayed.

Table C1: Static catchment attributes used for cluster prediction in the Random Forest model. Source: BasinATLAS Attributes (version 1.0).

Description (HydroATLAS name)	Unit
Catchment area (area)	km^2
Gauge latitude (gauge_lat)	decimal degrees
Gauge longitude (gauge_lon)	decimal degrees
Fraction of precipitation falling as snow (frac_snow)	–
Mean daily precipitation (p_mean)	mm/day
Mean daily potential evaporation (pet_mean_ERA5_LAND)	mm/day
Precipitation seasonality (seasonality_ERA5_LAND)	–
Aridity index (aridity_ERA5_LAND)	–
Terrain slope (slp_dg_sav)	degree (×10)
High precipitation duration (high_prec_dur)	days
High precipitation frequency (high_prec_freq)	–
Low precipitation duration (low_prec_dur)	days
Low precipitation frequency (low_prec_freq)	–
Climate moisture index (cmi_ix_syr)	index value (×10)
Forest cover extent (for_pc_sse)	% cover
Land cover classes (glc_cl_smj)	classes (n=22)
Road density (rdd_mk_sav)	m/km^2

Table C2: Mean climate and physiographic attributes of the 11 clusters. For each attribute the mean and standard deviation are reported.

Cluster	Number of stations	Precipitation (mm/year)	Aridity (-)	Elevation (m.a.s.l)	Snow fraction (-)	Area (km^2)
0	1178	1012.7 ± 364.5	0.8 ± 0.4	595.8 ± 545.1	0.2 ± 0.1	1644.3 ± 8842.8
1	3669	1178.2 ± 225.9	0.8 ± 0.3	399.5 ± 377.5	0.1 ± 0.1	1260.2 ± 5291.9
2	1914	910.5 ± 423.2	0.9 ± 0.5	1803.2 ± 926.4	0.4 ± 0.2	3249.8 ± 14052.7
3	1754	1343.0 ± 549.0	0.6 ± 0.6	416.4 ± 426.4	0.0 ± 0.1	8013.7 ± 149006.2
4	620	977.3 ± 374.8	1.3 ± 0.9	487.9 ± 596.3	0.1 ± 0.1	1074.8 ± 7219.3
5	1480	898.1 ± 395.5	1.5 ± 0.9	515.0 ± 532.9	0.0 ± 0.1	1107.4 ± 5942.4
6	1569	885.7 ± 356.9	1.0 ± 0.5	1350.3 ± 846.7	0.3 ± 0.2	2158.5 ± 10009.4
7	1015	1244.6 ± 568.3	1.0 ± 0.8	1092.6 ± 748.7	0.1 ± 0.2	11962.8 ± 48136.1
8	1169	1151.3 ± 363.7	0.9 ± 0.4	616.6 ± 592.9	0.1 ± 0.2	1405.0 ± 6933.1
9	646	972.3 ± 493.7	1.8 ± 1.5	843.1 ± 1027.3	0.0 ± 0.1	22410.7 ± 110030.4
10	1039	914.0 ± 311.0	1.0 ± 0.5	536.9 ± 440.5	0.2 ± 0.1	1504.7 ± 5149.5

D Hyperparameter Tuning Results

Table D3: Hyperparameter search space for the different models. The hyperparameter names may not be the same as in the original models' publications.

Model	Hyperparameter	Search Space or Fixed Value
Common Hyperparameters		
All Models	Input Length	30-365 days
	Learning Rate	1e-6 to 1e-3 (log)
	Output Length	10 days (Fixed)
	Batch Size	2048 (Fixed)
	Scheduler Factor	0.5 (Fixed)
EA-LSTM		
EA-LSTM	Hidden Size	32-256
	Number of Layers	1-3
	Dropout	0.0-0.5
	Bias	True (Fixed)
	Bidirectional	True (Fixed)
	Bidirectional Fusion	Concat (Fixed)
TFT		
TFT	Hidden Size	32-128 (step of 8)
	Dropout	0.0-0.5
	Attention Dropout	0.0-0.3
	Number of Attention Heads	1-8
	LSTM Layers	1-3
	Encoder Layers	1-3

Continued on next page

Table D3 – continued from previous page

Model	Hyperparameter	Search Space or Fixed Value
	Add Relative Index	True or False
	Context Length Ratio	0.5-1.0
	Quantiles	[0.5] (Fixed)
	TiDE	
TiDE	Hidden Size	32-128
	Dropout	0.0-0.5
	Number of Encoder Layers	1-3
	Number of Decoder Layers	1-3
	Decoder Output Size	8-32
	Temporal Decoder Hidden Size	16-64
	Use Layer Norm	True or False
	Past Feature Projection Size	0 (Fixed)
	Future Forcing Projection Size	0 (Fixed)
	TSMixer	
TSMixer	Hidden Size	32-128
	Dropout	0.0-0.5
	Number of Mixing Layers	2-15
	Static Embedding Size	5-20
	Fusion Method	Add or Concat

Table D4: Optimal hyperparameters for Experiment 1 (Regional Transfer Learning) and Experiment 2, Phase 1 (In-Memory Global Transfer Learning). For Experiment 1, only Tajikistan hyperparameters were used. Separate hyperparameter tuning was conducted for each target domain.

Model	Hyperparameter	Tajikistan	Kyrgyzstan
	EA-LSTM		
EA-LSTM	Learning Rate	8.86×10^{-4}	6.30×10^{-5}
	Input Length	210 days	365 days
	Hidden Size	206	49
	Number of Layers	1	1
	Dropout	0.497	0.048
	TFT		
TFT	Learning Rate	8.42×10^{-4}	9.03×10^{-5}
	Input Length	260 days	310 days
	Hidden Size	48	72
	Dropout	0.089	0.299
	Attention Dropout	0.088	0.217
	Number of Attention Heads	1	3

Continued on next page

Table D4 – continued from previous page

Model	Hyperparameter	Tajikistan	Kyrgyzstan
	LSTM Layers	1	3
	Encoder Layers	2	1
	Add Relative Index	`True`	`False`
	Context Length Ratio	0.781	0.551
	TiDE		
TiDE	Learning Rate	4.19×10^{-4}	9.21×10^{-4}
	Input Length	279 days	361 days
	Hidden Size	67	128
	Dropout	0.188	0.499
	Number of Encoder Layers	3	1
	Number of Decoder Layers	3	3
	Decoder Output Size	20	25
	Temporal Decoder Hidden Size	21	63
	Use Layer Norm	`False`	`False`
	TSMixer		
TSMixer	Learning Rate	4.02×10^{-4}	8.82×10^{-4}
	Input Length	124 days	73 days
	Hidden Size	84	55
	Dropout	0.070	0.022
	Number of Mixing Layers	3	3
	Static Embedding Size	20	6
	Fusion Method	`Add`	`Concat`

Table D5: Optimal hyperparameters for Experiment 2, Phase 2: Larger-than-RAM Global Deep Transfer Learning. Separate hyperparameter tuning was conducted for each target domain. Note that RevIN was disabled (use_rev_in: false) for all models in this experiment.

Model	Hyperparameter	Tajikistan	Kyrgyzstan
	EA-LSTM		
EA-LSTM	Learning Rate	5.00×10^{-5}	8.85×10^{-4}
	Input Length	264 days	302 days
	Hidden Size	191	164
	Number of Layers	1	1
	Dropout	0.426	0.391
	Temporal Fusion Transformer (TFT)		
TFT	Learning Rate	5.62×10^{-4}	5.12×10^{-5}
	Input Length	185 days	331 days
	Hidden Size	40	56
	Dropout	0.001	0.023
	Attention Dropout	0.288	0.054
	Number of Attention Heads	8	4

Continued on next page

Table D5 – continued from previous page

Model	Hyperparameter	Tajikistan	Kyrgyzstan
	LSTM Layers	2	2
	Encoder Layers	3	1
	Add Relative Index	`False`	`False`
	Context Length Ratio	0.631	0.546
	TiDE		
TiDE	Learning Rate	9.69×10^{-4}	7.96×10^{-5}
	Input Length	276 days	217 days
	Hidden Size	99	40
	Dropout	0.296	0.136
	Number of Encoder Layers	2	2
	Number of Decoder Layers	3	1
	Decoder Output Size	22	8
	Temporal Decoder Hidden Size	20	64
	Use Layer Norm	`False`	`False`
	TSMixer		
TSMixer	Learning Rate	2.60×10^{-4}	3.13×10^{-4}
	Input Length	127 days	53 days
	Hidden Size	56	51
	Dropout	0.121	0.030
	Number of Mixing Layers	9	7
	Static Embedding Size	14	9
	Fusion Method	`Add`	`Add`

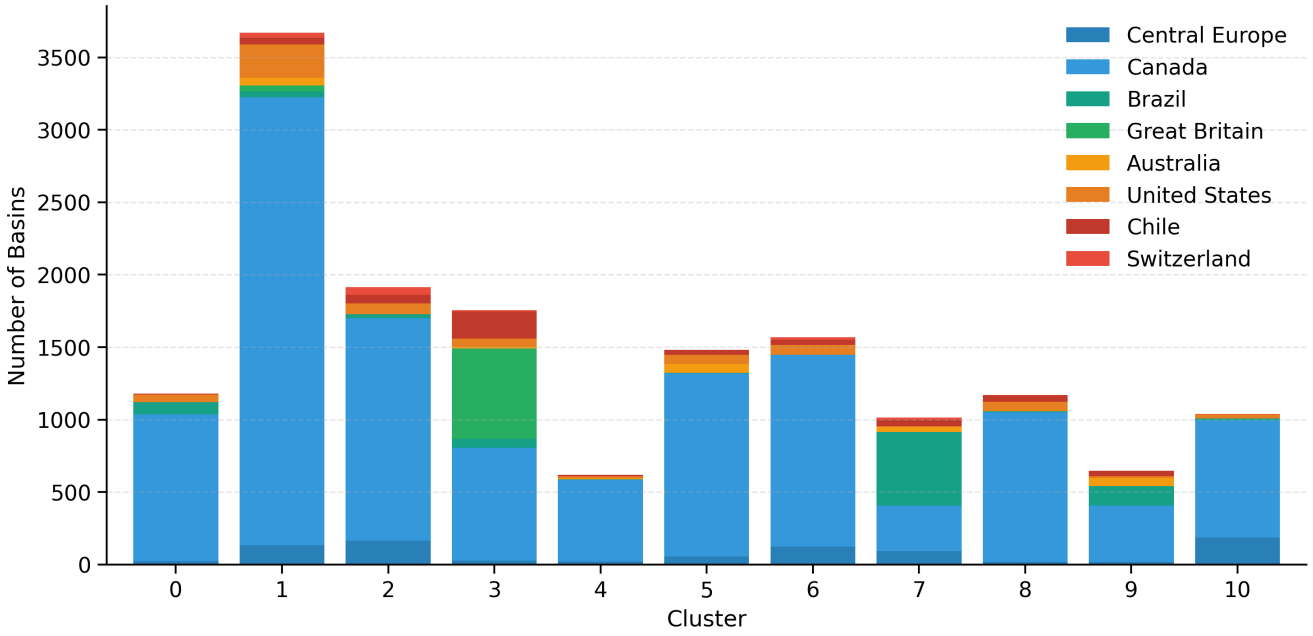

Figure C7: Stacked bar chart showing the distribution of basins by country across all clusters. Each bar represents a cluster, with colored segments indicating the number of basins from each country. Clusters 2 and 7 are identified as most similar to Central Asian catchments.

E Additional Transfer Learning Results

Table E6: Number of basins with catastrophic failures (NSE ¡ 0) across the 1, 5 and 10 days forecast horizons for Tajikistan. Values represent the total count of basin-horizon combinations with negative NSE out of 45 possible combinations (15 basins × 3 horizons).

Architecture	Benchmark	Volume-Based TL	Similarity-Based TL
EA-LSTM	3	0	1
TFT	2	0	0
TiDE	3	0	1
TSMixer	4	1	0
Total	12	1	2

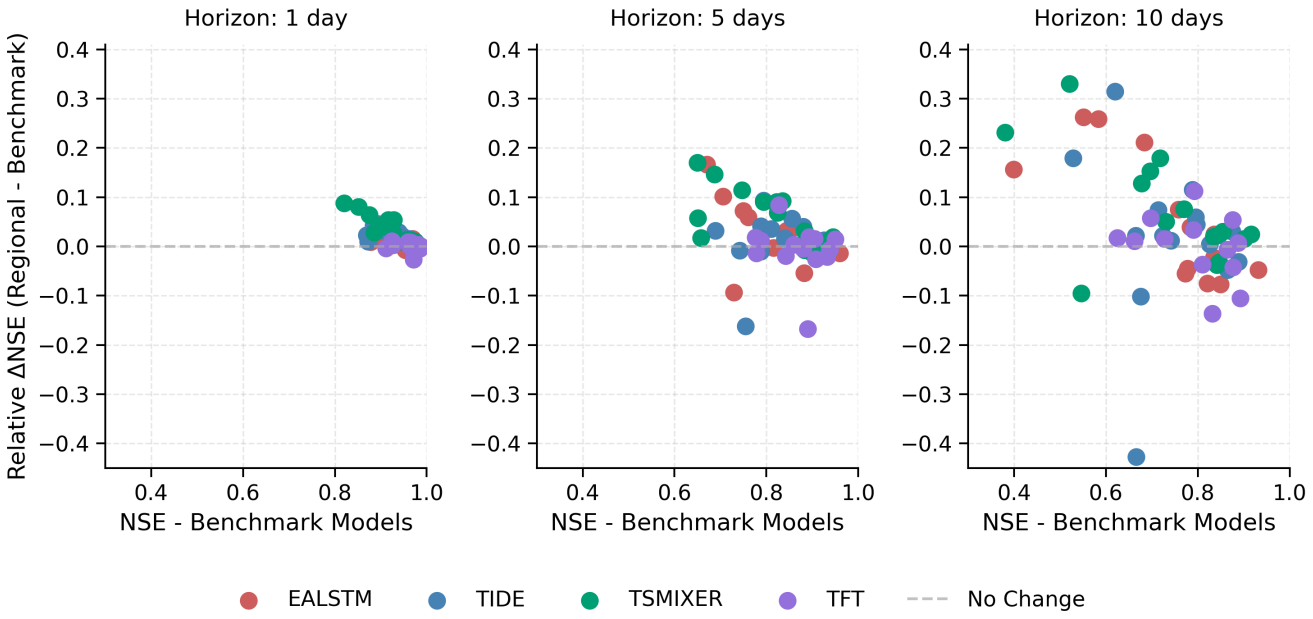

Figure E8: Relative change in NSE from regional transfer learning plotted against benchmark model performance across three forecast horizons (1, 5, and 10 days). Each point represents one basin-architecture combination across the 15 Tajik basins and four deep learning architectures (EA-LSTM, TiDE, TFT, TSMixer). The dashed line represents no change.

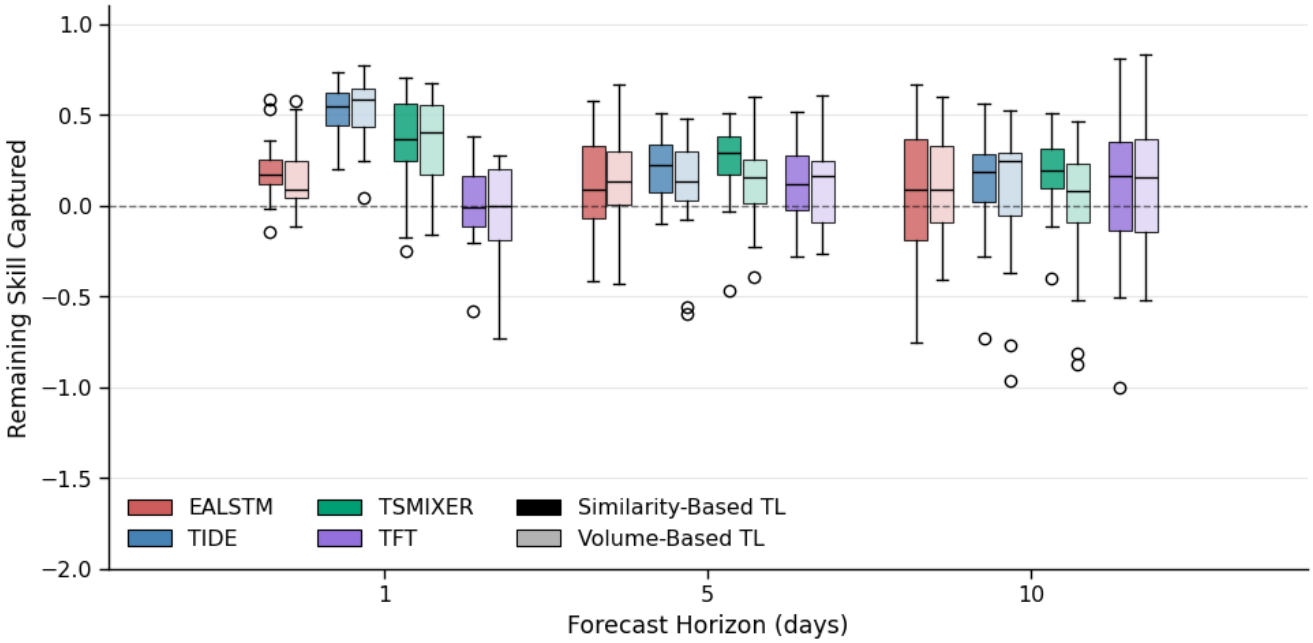

Figure E9: Remaining Skill Captured (RSC) values for four deep learning architectures across three forecast horizons (1, 5, and 10 days) for Tajikistan. RSC quantifies the proportion of potential Nash-Sutcliffe Efficiency improvement achieved through global transfer learning, relative to benchmark models. The figure presents RSC distributions for two global transfer learning strategies: volume-based and similarity-based. Within each architectural group (e.g., all red boxes for EA-LSTM), the darker shade of the boxplot represents the similarity-based transfer learning strategy. In comparison, the lighter shade represents the volume-based transfer learning strategy. Each boxplot represents the distribution of RSC values across the 15 Tajik basins for each architecture, horizon, and strategy combination.

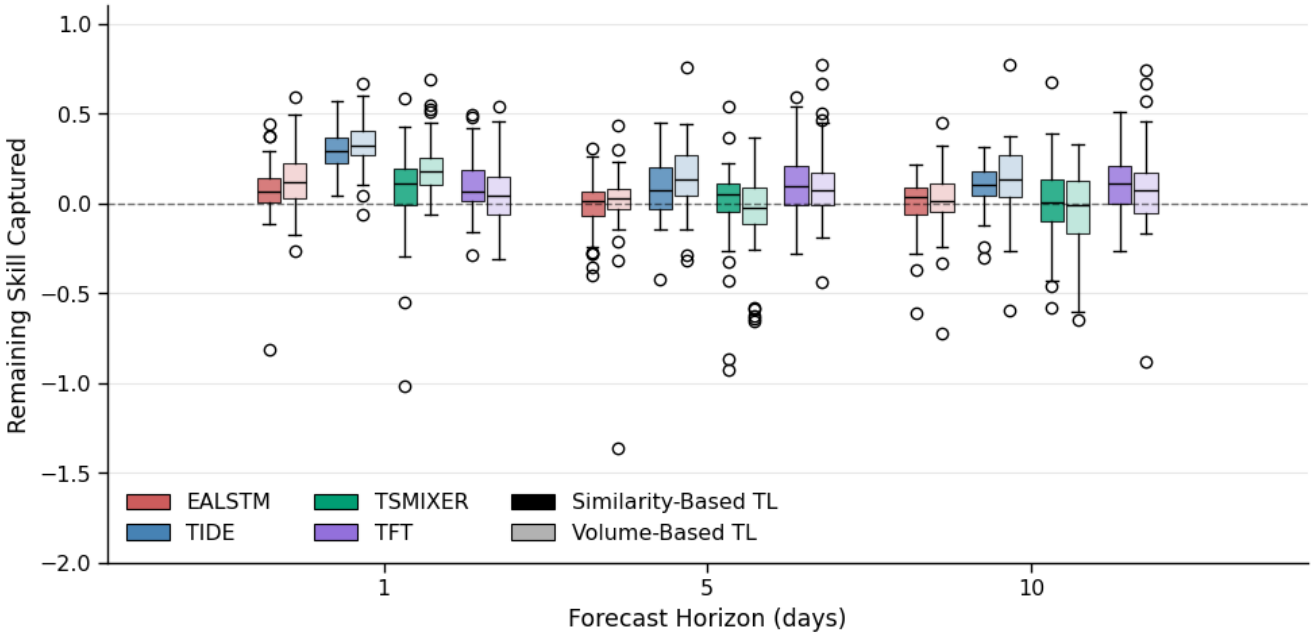

Figure E10: Remaining Skill Captured (RSC) values for four deep learning architectures across three forecast horizons (1, 5, and 10 days) for Kyrgyzstan. RSC quantifies the proportion of potential Nash-Sutcliffe Efficiency improvement achieved through global transfer learning, relative to benchmark models. The figure presents RSC distributions for two global transfer learning strategies: volume-based and similarity-based. Within each architectural group (e.g., all red boxes for EA-LSTM), the darker shade of the boxplot represents the similarity-based transfer learning strategy. In comparison, the lighter shade represents the volume-based transfer learning strategy. Each boxplot represents the distribution of RSC values across the 59 Kyrgyz basins for each architecture, horizon, and strategy combination.

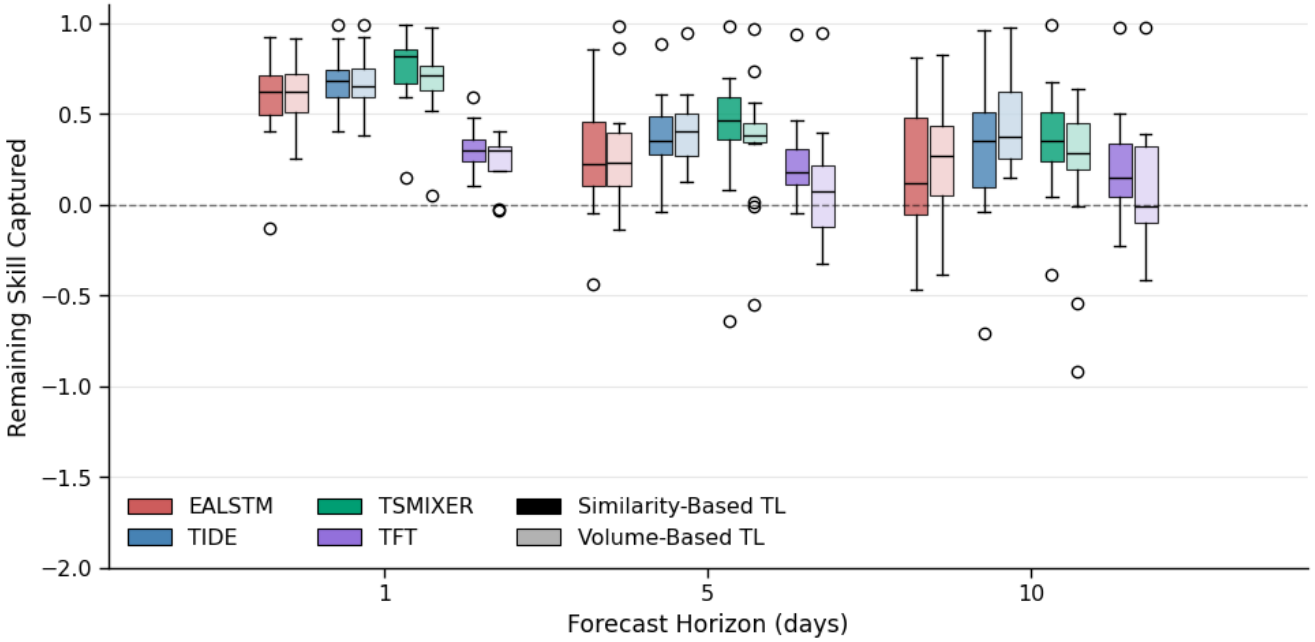

Figure E11: Remaining Skill Captured (RSC) values for four deep learning architectures across three forecast horizons (1, 5, and 10 days) for the Tajikistan Phase 2 experiment. RSC quantifies the proportion of potential Nash-Sutcliffe Efficiency improvement achieved through larger-than-RAM global transfer learning relative to benchmark models. The figure compares two transfer learning strategies: similarity-based (darker shades, pre-trained on 1,850 hydrologically similar catchments) and volume-based (lighter shades, pre-trained on 6,690 catchments). Each boxplot represents the distribution of RSC values across the 15 Tajik basins for each architecture, horizon, and strategy combination.

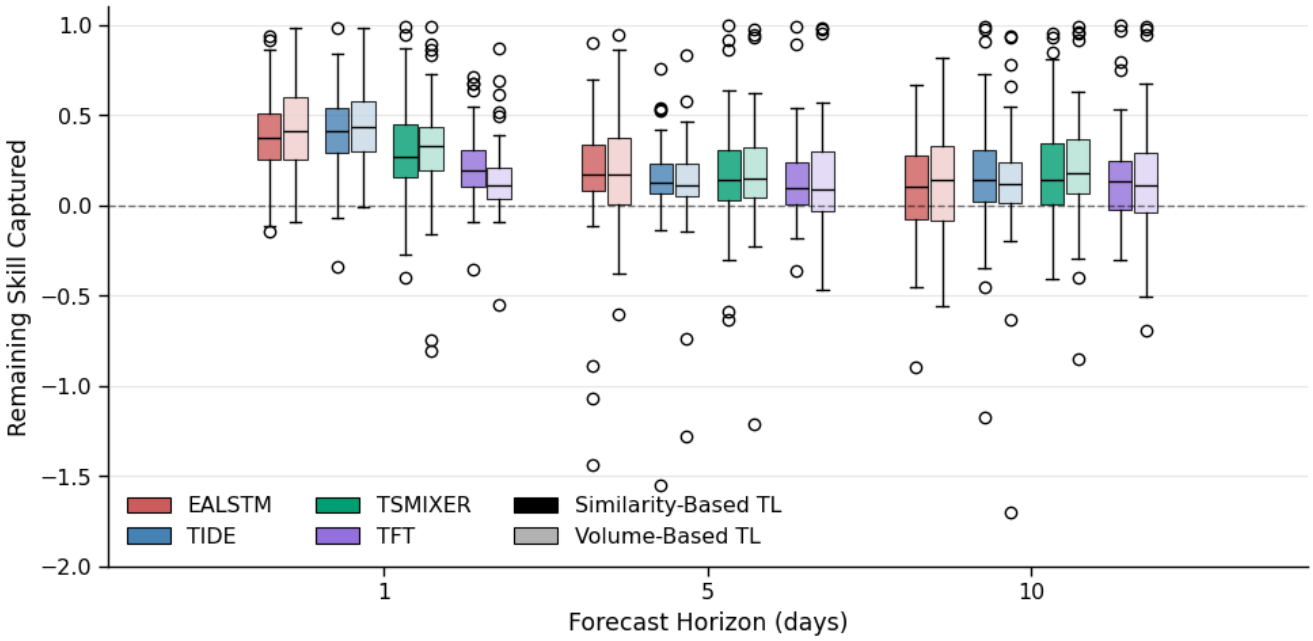

Figure E12: Remaining Skill Captured (RSC) values for four deep learning architectures across three forecast horizons (1, 5, and 10 days) for the Kyrgyzstan Phase 2 experiment. RSC quantifies the proportion of potential Nash-Sutcliffe Efficiency improvement achieved through larger-than-RAM global transfer learning relative to benchmark models. The figure compares two transfer learning strategies: similarity-based (darker shades, pre-trained on 1,850 hydrologically similar catchments) and volume-based (lighter shades, pre-trained on 6,690 catchments). Each boxplot represents the distribution of RSC values across the 59 Kyrgyz basins for each architecture, horizon, and strategy combination.

Table E7: Number of basins with catastrophic failures (NSE ¡ 0) across the 1, 5 and 10 days forecast horizons for Kyrgyzstan. Values represent the total count of basin-horizon combinations with negative NSE out of 177 possible combinations (59 basins × 3 horizons).

Architecture	Benchmark	Volume-Based TL	Similarity-Based TL
EA-LSTM	8	3	5
TFT	8	2	4
TiDE	5	7	4
TSMixer	12	5	7
Total	33	17	20

F Derivation of the RSC to MSE Identity

This derivation shows that the Remaining Skill Captured (RSC) metric is equivalent to one minus the ratio of the Mean Squared Error (MSE) of the new model to the MSE of the baseline model.

The Nash-Sutcliffe Efficiency (NSE) is defined as:

$$\text{NSE} = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

Let the sum of squared residuals be $\text{SS}_{\text{res}} = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2$ and the total sum of squares be $\text{SS}_{\text{tot}} = \sum_{i=1}^{n}(y_i - \bar{y})^2$. The formulas for NSE and MSE can be written as:

$$\text{NSE} = 1 - \frac{\text{SS}_{\text{res}}}{\text{SS}_{\text{tot}}} \quad \text{and} \quad \text{MSE} = \frac{\text{SS}_{\text{res}}}{n}$$

The derivation begins by substituting the definition of NSE into the RSC formula. The term SS_{tot} is constant for both models as it depends only on the observed values.

$$\begin{aligned}
\text{RSC} &= \frac{\text{NSE}_{\text{new}} - \text{NSE}_{\text{base}}}{1 - \text{NSE}_{\text{base}}} \\
&= \frac{\left(1 - \frac{\text{SS}_{\text{res, new}}}{\text{SS}_{\text{tot}}}\right) - \left(1 - \frac{\text{SS}_{\text{res, base}}}{\text{SS}_{\text{tot}}}\right)}{1 - \left(1 - \frac{\text{SS}_{\text{res, base}}}{\text{SS}_{\text{tot}}}\right)} \\
&= \frac{\frac{\text{SS}_{\text{res, base}} - \text{SS}_{\text{res, new}}}{\text{SS}_{\text{tot}}}}{\frac{\text{SS}_{\text{res, base}}}{\text{SS}_{\text{tot}}}} \\
&= \frac{\text{SS}_{\text{res, base}} - \text{SS}_{\text{res, new}}}{\text{SS}_{\text{res, base}}} \\
&= 1 - \frac{\text{SS}_{\text{res, new}}}{\text{SS}_{\text{res, base}}}
\end{aligned}$$

Since $\text{MSE} = \text{SS}_{\text{res}}/n$, the ratio of the MSEs is:

$$\frac{\text{MSE}_{\text{new}}}{\text{MSE}_{\text{base}}} = \frac{\text{SS}_{\text{res, new}}/n}{\text{SS}_{\text{res, base}}/n} = \frac{\text{SS}_{\text{res, new}}}{\text{SS}_{\text{res, base}}}$$

This confirms the identity:

$$\text{RSC} = 1 - \frac{\text{MSE}_{\text{new}}}{\text{MSE}_{\text{base}}}$$

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Declaration of originality

The signed declaration of originality is a component of every written paper or thesis authored during the course of studies. In consultation with the supervisor, one of the following three options must be selected:

- ○ I confirm that I authored the work in question independently and in my own words, i.e. that no one helped me to author it. Suggestions from the supervisor regarding language and content are excepted. I used no generative artificial intelligence technologies[1].
- ○ I confirm that I authored the work in question independently and in my own words, i.e. that no one helped me to author it. Suggestions from the supervisor regarding language and content are excepted. I used and cited generative artificial intelligence technologies[2].
- ◉ I confirm that I authored the work in question independently and in my own words, i.e. that no one helped me to author it. Suggestions from the supervisor regarding language and content are excepted. I used generative artificial intelligence technologies[3]. In consultation with the supervisor, I did not cite them.

Title of paper or thesis:

Advancing Operational Hydrology through Deep Transfer Learning: Multi-Day Streamflow Forecasting in Central Asian Mountains

Authored by:
If the work was compiled in a group, the names of all authors are required.

Last name(s):	**First name(s):**
LAZARO	NICOLAS

With my signature I confirm the following:

- I have adhered to the rules set out in the Citation Guide.
- I have documented all methods, data and processes truthfully and fully.
- I have mentioned all persons who were significant facilitators of the work.

I am aware that the work may be screened electronically for originality.

Place, date	**Signature(s)**
ZURICH, 18.08.2025	

If the work was compiled in a group, the names of all authors are required. Through their signatures they vouch jointly for the entire content of the written work.

[1] E.g. ChatGPT, DALL E 2, Google Bard
[2] E.g. ChatGPT, DALL E 2, Google Bard
[3] E.g. ChatGPT, DALL E 2, Google Bard