Machine Learning Techniques for Time Series Regression in Unmonitored Environmental Systems

A THESIS
SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL
OF THE UNIVERSITY OF MINNESOTA
BY

Jared Daniel Willard

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
Doctor of Philosophy

Prof. Vipin Kumar, Advisor

April, 2023

Acknowledgements

I first would like to acknowledge that the University of Minnesota Twin Cities stands on MiníSóta Makhóčhe. Support for my learning and research has come from this land grant institution illegally occupying the homelands of the Dakhóta Oyóte. The historical and ongoing systemic erasure of indigenous lands, lives and livelihoods by the University of Minnesota, the State of Minnesota, and the United States of America has to be acknowledged, reckoned with, and land re-matriated into good relationships for there to be any hope of sustainable and just futures.

The University of Minnesota has also directly and indirectly benefited from the commodification, incarceration and death of Black people, Indigenous peoples, land, and food in MiníSóta and around the world.

I continue to begin this text with gratitude - for the many relationships, histories, and organizations that have supported, enabled and guided me along this 6 year Ph.D. journey. I would like thank my supervisor, Vipin Kumar, for his guidance and support throughout this study. His passion for research and writing is contagious. To the many members of the Kumar Lab, Kshitij Tayal, Xiaowei Jia, Rahul Ghosh, Laura Connor, Saurabh Agrawal, Shaoming Xu, Praveen Ravirathinam, Somya Sharma, Xiang Li, Kelly Cutler, Ankush Khandelwal, Michael Steinbach, and Arvind Renganathan, I was grateful for the support and company on the long and arduous journey of graduate school. I was also lucky to have many research experiences, collaborators, and mentors outside the university throughout this study. Especially I would like to thank Jordan Read and the US Geological Survey's Water Resources Data Science Branch including Alison Appling, Samantha Oliver, Jacob Zwart, Jeff Sadler, and Simon Topp. Also I'd like to thank Charuleka Varadharajan and others working at or with Lawrence Berkeley

National Lab including Helen Weierbach, Aranildo Lima, Mohammed Ombadi, and Fabio Ciulla. Outside the academy, I'm eternally grateful to have inspiring mentors and teachers in my adolescence that spurred my interest in science and learning. Brian Tillmann, Kalin Laurent, Nathan Strenge, and Ka Ming Tam all contributed to my development as a lifelong student envisioning a world where education and innovation stem from collaboration and compassion rather than competition.

As key as all these people and organizations have been, my family is undoubtedly the fundamental structure supporting everything I've done. My parents amaze me everyday with their selflessness, resiliency, and ambition. I admire and love them and would be lost without them. They have sacrificed so much to give their children a stable and joyous life with many irreplaceable memories. My life partner Sohini inspires me everyday with their creativity and passion, and their lovingly goofy yet fiercely supportive presence makes life a joy. I hope to continue our adventures as our lives move forward through and beyond graduate school.

I leave the remaining space in memory of Justin Willard (1995-2014). Your brilliance, kindness, and artistic nature inspire myself and so many others every day.

Dedication

To my parents, for their endless love and support.

Abstract

This thesis provides a computer science audience with a review of machine learning techniques for modeling time series in unmonitored environmental systems with no available target data that have been published in recent years, and further includes three distinct research efforts applying these methods to real-world water resources prediction scenarios. Additionally, we identify several open questions for time series prediction in unmonitored sites that include incorporating dynamic inputs and site characteristics, mechanistic understanding, and explainable AI techniques in modern machine learning frameworks. This is motivated by the current state of environmental time series modeling seeing a vast increase in applications of various machine learning models, in particular deep learning models built using the growing availability of high performance computing resources. It remains difficult to predict environmental variables for which observations are concentrated in a minority of locations and most locations remain unmonitored, and although many machine learning-based approaches have been developed, there is often a lack of comparison between them. The increased attention to environmental prediction topics such as disaster response, water resources management, and climate change reveal a need to compare these approaches, and understand when and where they should be applied in unmonitored environmental prediction scenarios.

Contents

Acknowledgements **i**

Dedication **iii**

Abstract **iv**

List of Tables **viii**

List of Figures **xi**

1 Introduction **1**

1.1 Overview 1

1.2 Thesis Contributions and Organization 3

2 Overview of Machine Learning Techniques for Unmonitored Prediction **5**

2.1 Introduction 5

2.2 Machine Learning Frameworks for Unmonitored Prediction 6

2.2.1 Broad-scale models using all available entities 7

2.2.2 Broad-scale models using a subgroup of entities 15

2.2.3 Transfer learning 16

2.2.4 Cross cutting theme: knowledge-guided machine learning 21

2.3 Summary and Discussion 34

3 Predicting Water Temperature Dynamics of Unmonitored Lakes with Meta Transfer Learning **37**

3.1 Introduction . . . 37
3.2 Materials and Methods . . . 41
3.2.1 Overview . . . 41
3.2.2 Process-Based Models . . . 42
3.2.3 Process-Guided Deep Learning (PGDL) Models . . . 43
3.2.4 Meta Transfer Learning with Gradient Boosting Regression . . . 43
3.2.5 Data . . . 48
3.2.6 Model Experiments . . . 49
3.3 Results . . . 51
3.4 Discussion . . . 62

4 Entity-aware LSTM estimates of daily surface temperatures for unmonitored lakes 68
4.1 Introduction . . . 68
4.1.1 Associated Dataset Description . . . 69
4.2 Methods . . . 71
4.2.1 Model Descriptions . . . 71
4.2.2 Input: Meteorological conditions and lake-specific characteristics 73
4.2.3 In-situ lake temperature data . . . 75
4.2.4 Oversampling . . . 77
4.2.5 Hyperparameter tuning . . . 77
4.2.6 Error estimation . . . 79
4.2.7 Training EA-LSTM and prediction of 185,549 lakes . . . 80
4.2.8 Technical Validation Methods . . . 81
4.3 Results of Technical Validation . . . 82
4.4 Data Use and Recommendations for Reuse . . . 88
4.5 Comparison with existing datasets . . . 89

5 Transfer learning and broad-scale machine learning modeling of water temperature in unmonitored stream sites 93
5.1 Introduction . . . 93
5.2 Methods . . . 95
5.2.1 Data . . . 95

5.2.2 Model descriptions 97
5.2.3 Experiment description 101
5.3 Results 103
5.3.1 Performance on 580 test stream sites 103
5.3.2 Prediction performance 103
5.3.3 Feature Importances 104
5.4 Discussion 106

6 Open Questions and Future Directions in Unmonitored Prediction 120
6.1 Is more data always better? How do we construct optimal training datasets? 120
6.2 How do we include specificity of place when applying broad-scale models? 122
6.3 How do we address non-stationarity in site characteristics? 123
6.4 Can we use auxiliary data to improve modeling? 125
6.5 How can we leverage process understanding for unmonitored prediction? 126
6.6 How can we leverage model ensembles for unmonitored prediction? . . . 128
6.7 Can we use existing explainable AI techniques to derive domain knowledge? 129
6.8 Can existing process knowledge help to build better explainable AI techniques? 131

7 Conclusion 133

References 134

List of Tables

2.1 Literature Table. Abbreviations as follows, DCBS: direct concatenation broad-scale, TL: transfer learning ANN: artificial neural network (feed forward multilayer perceptron), GNN: graph neural network, LSTM: long short-term memory neural network, MARS: multi-adaptive regression splines, MLR: multilinear regression, GBR: gradient boosting regression, GRU: gated recurrent unit, PDE: partial differential equation, RF: random forest, SVR: support vector regression, TCN: temporal convolution network, XGB: extreme gradient boosting 28

2.1 Literature Table. Abbreviations as follows, DCBS: direct concatenation broad-scale, TL: transfer learning ANN: artificial neural network (feed forward multilayer perceptron), GNN: graph neural network, LSTM: long short-term memory neural network, MARS: multi-adaptive regression splines, MLR: multilinear regression, GBR: gradient boosting regression, GRU: gated recurrent unit, PDE: partial differential equation, RF: random forest, SVR: support vector regression, TCN: temporal convolution network, XGB: extreme gradient boosting 29

2.1 Literature Table. Abbreviations as follows, DCBS: direct concatenation broad-scale, TL: transfer learning ANN: artificial neural network (feed forward multilayer perceptron), GNN: graph neural network, LSTM: long short-term memory neural network, MARS: multi-adaptive regression splines, MLR: multilinear regression, GBR: gradient boosting regression, GRU: gated recurrent unit, PDE: partial differential equation, RF: random forest, SVR: support vector regression, TCN: temporal convolution network, XGB: extreme gradient boosting 30

2.1 Literature Table. Abbreviations as follows, DCBS: direct concatenation broad-scale, TL: transfer learning ANN: artificial neural network (feed forward multilayer perceptron), GNN: graph neural network, LSTM: long short-term memory neural network, MARS: multi-adaptive regression splines, MLR: multilinear regression, GBR: gradient boosting regression, GRU: gated recurrent unit, PDE: partial differential equation, RF: random forest, SVR: support vector regression, TCN: temporal convolution network, XGB: extreme gradient boosting 31
2.1 Literature Table. Abbreviations as follows, DCBS: direct concatenation broad-scale, TL: transfer learning ANN: artificial neural network (feed forward multilayer perceptron), GNN: graph neural network, LSTM: long short-term memory neural network, MARS: multi-adaptive regression splines, MLR: multilinear regression, GBR: gradient boosting regression, GRU: gated recurrent unit, PDE: partial differential equation, RF: random forest, SVR: support vector regression, TCN: temporal convolution network, XGB: extreme gradient boosting 32
2.1 Literature Table. Abbreviations as follows, DCBS: direct concatenation broad-scale, TL: transfer learning ANN: artificial neural network (feed forward multilayer perceptron), GNN: graph neural network, LSTM: long short-term memory neural network, MARS: multi-adaptive regression splines, MLR: multilinear regression, GBR: gradient boosting regression, GRU: gated recurrent unit, PDE: partial differential equation, RF: random forest, SVR: support vector regression, TCN: temporal convolution network, XGB: extreme gradient boosting 33
3.1 *Results of PB-MTL and PGDL-MTL Applied to Test Lakes.* 51
3.2 *Median Actual RMSE of PGDL Source Models of Different Metamodel-Predicted Ranks.* 53
3.3 *Selected Features for PB-MTL and PGDL-MTL and Importances* 54
3.4 *Data for Figure 3.7, Performance of PGDL Trained on Various Amounts of Target Lake Temperature Data Profiles* 61
3.5 *Method Comparison Across Broad-Scale Modeling of 1882 Lakes in the Midwestern United States* 65

4.1 Performance comparison of the three modelling approaches across the five test folds in cross-validation. Here, ERA5* is the bias-corrected version of ERA5 (an offset of +3.31°C was applied to the ERA5 data), and LM is only tested on data from June to September. From left to right, Median lake-specific RMSE and overall RMSE assess overall performance, then median RMSE is shown for lakes within different size ranges, and lastly median bias of all observations in different temperature ranges is shown (all values are in ℃). Bias for bias-corrected ERA5* is not shown because observations were used in the bias correction itself, and bias in the lowest temperature range is not shown for LM due to lack of data. Numbers in parentheses represent the number of lakes (lake size) and observations (temperature group) in each data partition with the exception of the LM observations, which are lower due to their restriction to the summer months, and the ERA5 comparisons, which have 2,974 fewer observations from 70 coastal lakes that are not resolved in the dataset. 83
5.1 Overall RMSE statistics across the 580 testing stream sites 103
5.2 RMSE statistics per USGS-defined hydrological region (`https://water.usgs.gov/GIS/regions.html`). Values are the mean RMSE(°C) across sites in a region, with RMSE standard deviation in parentheses. Lowest mean values per region are shown in bold. 104
A.1 Feature Name Descriptions. Any feature appended by "*t-xx*" where *xx* is an integer is the time lagged value by *xx* days. 111

List of Figures

2.1 Example of an long short-term memory (LSTM) network model with directly concatenated site characteristics and dynamic inputs 9
2.2 Example of a combination static feature encoder neural network with a long short-term memory (LSTM) network model 10
2.3 Conceptual example of transductive and inductive graph learning. In both left and right panels, $\mathscr{F}$ is a model learned during training. Blue and red nodes represent entities with data for use in training and test entities without any data respectively. In transductive graph learning, the model has access to nodes and edges associated with test entities during training, but no new nodes can be introduced during testing. In inductive graph learning, the model is trained on an initial graph without any knowledge of the test entities, but the model can generalize to any new nodes during testing. 13
2.4 Process diagram of the Meta Transfer Learning framework. Models are first built from data-rich source domains. The metamodel is trained using characteristics extracted from the source domains to predict the performance metrics from transferring models between source domains. Then given a target system or domain, the metamodel is able to output a prediction of how well each of the source models will perform on the target system. Adapted from Willard et al. [1] 17

3.1 Meta-learning general framework. The meta learning model (metamodel) is trained to predict source model performance (root mean square error; RMSE) based on lake domain characteristics (meta features). The performances and characteristics from all source models applied to all other source lakes are used for metamodel training. This trained model is used to predict source model performance and inform source model selection for a *new* target lake. 45

3.2 Map of all lakes used in experiments. 145 source lakes are shown in red, 305 initial target lakes are shown in blue, and the additional 1882 expanded target lakes are shown in yelllow. 48

3.3 Comparison of the performance of the three MTL approaches relative to PB0 on 305 lakes. a-c) RMSE of PB0 relative to the three transfer models, where the dotted line shows the 1:1 relationship. d-f) The difference between RMSE of the transfer and PB0 models, where the black dotted line shows the zero or no change line, and the solid colored lines show the linear regression fit of the change in RMSE as a function of PB0 RMSE. g-i) The distribution of RMSE from PB0 and transfer models, where the vertical gray and colored lines are the median PB0 and transfer RMSE, respectively. PB-MTL (a,d,g) and PGDL-MTL (b,e,h) are the transfer of process-based and PGDL models respectively, and PGDL-MTL9 (c,f,i) is an averaged ensemble prediction of the top 9 PGDL models. 52

3.4 Deep-water predictions for three example lakes to illustrate the application of PGDL-MTL and PGDL-MTL9. Lakes were selected to represent successful and unsuccessful PGDL-MTL results for stratified lakes (rows 1 and 2, respectively) and the easier case of a mixed lake (row 3). Steiger, Pat, and Tait Lakes have 2,573, 469, and 3,865 total temperature observations, respectively. Panels a-c: Time series predictions at two depths in 2012 for each target lake from the top-ranked PGDL source (Source 1), 9th-ranked source (Source 9), and a lower-ranked source (Source 99), with observed values (points) for comparison. Panels d-f: metamodel selections of source lakes for each lake, arranged by three features that dominated the MTL predictions: maximum depth (y axis), surface area (x axis), and predicted stratification (darker = stratified). Panels g-i: Metamodel-predicted RMSEs versus actual RMSEs (for all depths and years) for the three example lakes. 55

3.5 Top-selected source models compared to lesser-selected sources. In a), the seven process-based (PB) models chosen as a top source for ten or more target lakes by the meta transfer learning (MTL) model are shown in red, with grey lines connected to the paired target lake location; b) is the same as a) but for process-guided deep learning source models. In c), properties of lakes in the upper quartile of commonly chosen PB source models (white fill boxplot) are compared to the lowest quartile (hashed fill boxplot; based on MTL rank). Red dots represent the location of the seven source lakes featured in a). d) is the same as c), but for process-guided deep learning source models. 57

3.6 Plot showing the distribution of actual ranks of the metamodel-predicted top source models, for metamodels built on either PB sources (gray fill) or PGDL sources (white fill). Leftmost pair of bars: actual ranks for top-predicted models for each of the 305 target lakes. Other bars: best, median, and worst of the top-9-predicted sources. 59

3.7 Median RMSE for PGDL trained with differing numbers of temperature profiles, with error bars representing upper and lower quartiles of the median RMSE across the 5 randomized selections of observations for each target lake. Colored horizontal lines represent the median RMSE with a band showing the range from lower to upper quartile for PGDL-MTL and PGDL-MTL9 60
3.8 Comparison of model performance of PGDL-MTL (a,c,e) and PGDL-MTL9 (b,d,f) RMSE relative to PB0 across 1882 test lakes. a-b) RMSE of PB0 relative to the two transfer models, where the dotted line shows the 1:1 relationship. c-d) The difference between RMSE of the transfer and PB0 models, where the black dotted line shows the zero or no change line, and the solid colored lines show the linear regression fit of the change in RMSE as a function of PB0 RMSE. e-f) The distribution of RMSE from PB0 and transfer models, where the vertical gray and colored lines are the median PB0 and transfer RMSE, respectively. 65
4.1 Overview of the data and modelling flow used to create the daily surface temperature predictions. The entity-aware long short-term memory neural network (EA-LSTM), a deep learning approach designed for time series and other sequential data, is built using the seven input drivers shown below as well as observed surface temperatures. EA-LSTM outputs are compared against the ERA5 reanalysis-simulated epilimnetic lake temperature outputs [2] and a linear model (LM) described in Bachmann et al. [3]. Each data component shown is available as part of the associated data release [4]; https://doi.org/10.5066/P9CEMS0M). Inset map displays summer predictions for a single date and the spatial division used to break up the largest files (prediction and weather data) into three NetCDF files. 72
4.2 Geographic and temporal coverage of in-situ surface temperature data. Panel (a) shows geographic coverage of the 12,227 observed lakes across single degree latitude and longitude cells in the conterminous United States. Panel (b) shows observations by season and by year between 1980 and 2020. 76

4.3 Nested cross-validation process. Performance is aggregated over the 5-fold outer loop where each instance of training folds also contains an inner 4-fold loop for hyperparameter tuning on validation data. Hyperparameters are selected to minimize error across validation folds. 79
4.4 Root mean square error (RMSE) for predicted compared to observed water temperatures within a single degree latitude and longitude cell for each of the three methods is shown in panels (a), (d), and (g). Only cells with at least 100 observations are shown. Panels (c), (f), and (i) show year-specific RMSE per modelling method. Panels (d) and (f) show the bias-corrected ERA5 errors (ERA5* in Table 1). The distributions of all 303,579 observations along with a 1:1 line are shown in panel (b) for EA-LSTM, panel (e) for ERA5, and (h) showing the same for LM but with only summer months included (n=187,774 observations). An additional 1:1 dotted line is shown in panel (e) with a y-intercept of -3.31 to represent the ERA5* bias-correction. 86
4.5 Bias of predicted compared to observed water temperatures for all three approaches. Panels (a), (d), and (g) show median bias per year ranging from 1980 to 2020. Panels (b), (e), and (h) show bias per 2℃ temperature bins ranging from 0-36℃. Day of year median bias is shown in panels (c), (f), and (i) with bins covering three days and positive and negative bias visualized as pointing outward and inward, respectively. Dashed rings denote biases at different radii of the plot and lines separate seasons (January 1st is the top of these plots). The dotted line in panels (d) and (e) represents the -3.31℃. shift for bias corrected ERA5 predictions (ERA5* in Table 1). 87
5.1 Per-year RMSE values for each method 105
5.2 Spatial distribution of RMSE values for the continental-scale LSTM model (*LSTM_conus*) model and the meta transfer learning with pre-training (MTL-PT) framework over 580 testing stream locations 109
A.1 Observations per year for the 580 test sites 110
A.2 Selected Features and Importances for the *MLR_conus* model 112
A.3 Selected Features and Importances for the *XGB_conus* model 113

A.4 Selected Features and Importances for the *LSTM_conus* model 114
A.5 Selected Features and Importances for the *LSTM_regional* model 115
A.6 Feature Importances for the *MTL* model 116
A.7 Feature Importances for the *MTL_PT* model 117
A.8 Meta-feature Importances for the *MTL* metamodel. Features appended by "_diff" represent the difference calculated between the source and target system as target value minus source value. 118
A.9 Meta-feature Importances for the *MTL_PT* metamodel. Features appended by "_diff" represent the difference calculated between the source and target system as target value minus source value. 119

Chapter 1

Introduction

1.1 Overview

Environmental data often does not exist at the appropriate spatial resolution or coverage for decision-making or characterizing change. Although advances in sensor networks have resulted in a data deluge [5, 6], the amount of observations available will continue to be inadequate for the foreseeable future, notably for environmental variables where observations are concentrated in a minority of locations and most locations remain unmonitored. Since observing key variables at scale is prohibitively costly [7], models that can efficiently use the existing data and transfer information to unmonitored systems are critical to closing our information gaps. For instance, the problem of streamflow prediction in ungauged basins, also known as "PUBs", has been a longstanding challenge in hydrology due to its importance for the design of drainage infrastructure, flood defenses, and energy production especially under changing climate regimes and increasing human impacts on water resources [8, 9, 10, 11].

Machine learning (ML) models and their alternatives — process-based, empirical, and statistical models — take a variety of forms and have been used to predict key ecosystem variables such as soil moisture [12], hydrological flow [13], stream temperature [14], and lake temperature [4], which otherwise would be unavailable at the spatial and temporal scales needed for environmental decision-making [15]. ML models in particular have continually outperformed the traditional process-based models across hydrology and water resources in terms of both predictive performance at a broad scale

and computational efficiency [16, 17, 6, 18]. In particular, deep learning architectures like long short-term memory (LSTM) networks have been popularized due to their ability to model memory effects and cumulative status within ecosystems (e.g., snowpack depth; [19]). LSTMs have shown to outperform both state-of-the-art process-based models and also classical ML models in applications like lake temperature [17, 20], stream temperature [14], and groundwater dynamics [21] among many others.

However, these models have primarily focused on predictions in well-monitored locations, where a model is trained on a time period within one site and then predictions are made for new time periods at the same site. We refer to this as the *monitored* prediction scenario (also called the *gauged* scenario in streamflow modeling). While temporal extrapolation like this is important, extrapolating to *unmonitored* sites is even more crucial because in many cases the vast majority of sites remain unmonitored for many environmental variables. For example, Rahmani et al. [22] note that the U.S. Geological Survey's (USGS) National Water Information System [23] contains 5000 gauging stations, of which only 820 stations have stream temperature measurements for >10% of the days between 2004 and 2016. Similarly, Willard et al. [4] found that of the over 185,000 lakes at least 4 hectares in area in the conterminous United States, just over 12,000 had at least one lake surface temperature measurement. Of those lakes, less than 1% have 100 or more days of temperature observations and less than 5% have 10 or more days [24].

The unmonitored prediction scenario also becomes increasingly globally important as the brunt of the impact of climate change on water systems disproportionately falls on low and middle income countries with significantly less monitoring data and who are severely underrepresented in current scientific discourse, analysis, and policy [25, 26, 27]. Understanding how to leverage state-of-the-art ML alongside the wealth of observational data from high income countries to predict in unmonitored sites can lend insights into both how to train and select models to transfer to new regions and how new monitoring paradigms can be set up for optimal efficiency to refine existing models and datasets.

Traditionally water resources modeling in unmonitored sites has relied on the regionalization of process-based models. Regionalization techniques relate the parameter values of a model calibrated to the data of a monitored site to the inherent characteristics of the unmonitored site [28, 29, 30]. However, large uncertainty and mixed success

have prevented process-based model regionalization from being widely employed in hydrological analysis and design [31, 32, 33]. A major issue preventing this is the fact that there are usually strong interactions between process-based model parameters (e.g. between soil porosity and soil depth in rainfall-runoff models), such that any joint probability distribution over model parameters will be complex and multi-modal making calibration and regionalization difficult [34, 35]. This is closely related to the problem of equifinality [36], where different model structures or parameter values are equally capable of reproducing a similar hydrological outcome. On the other hand, ML models do regionalization implicitly without the dependence on expert knowledge, pre-defined hydrological models, and also often without any hydrological knowledge at all. Since ML models have significantly more flexibility in how parameters and connections between parameters are optimized, unlike process-based models where each parameter represents a specific system component or property, issues relevant to equifinality become largely irrelevant [37].

In recent years, numerous ML approaches have been explored for environmental variable time series prediction in unmonitored sites that span a variety of methods and applications in hydrology and water resources engineering. Though most of these approaches were developed in hydrology due to the wealth of streamflow data compared to other variables like river and lake water quality, these efforts are expanding as data collection and modeling continue to advance. However, often these approaches are not compared in detail with each other or sufficiently benchmarked making it challenging for researchers to know which to use for a given prediction task.

1.2 Thesis Contributions and Organization

This thesis provides a computer science audience with a review of machine learning techniques for modeling time series in unmonitored environmental systems with no available target data that have been published in recent, and further includes three distinct research efforts applying these methods to real-world water resources prediction scenarios. Additionally, we identify several open questions for time series prediction in unmonitored sites that include incorporating dynamic inputs and site characteristics, mechanistic understanding, and explainable AI techniques in modern machine learning

frameworks. The following are the main contributions of this work and the organization of the remainder of this thesis.

- Chapter 2 provides a comprehensive and systematic review of ML techniques for umonitored prediction and demonstrates the progress that has been made across different environmental applications using them.

- Chapter 3 presents a novel transfer learning framework for predicting in unmonitored sites. The proposed approach uses meta-learning to select either process-guided ML models or calibrated process-based models to predict in a given location given its characteristics, and is demonstrated for the task of daily at-depth temperature prediction in the Midwestern United States.

- Chapter 4 presents a dataset constructed using a continental-scale entity-aware LSTM model for daily lake surface temperature prediction from 1980-2020 for 185,549 lakes in the United States, and compares the approach with a state-of-the-art process-based model used for global reanalysis and an existing empirical model.

- Chapter 5 compares different approaches of ML modeling of unmonitored sites for the task of continental-scale stream temperature prediction at daily scale.

- Chapter 6 enumerates open questions that span the various gaps and opportunities for advancing research in this promising direction that have been revealed through the review of techniques in Chapter 2 and the case studies in Chapters 3-5.

Chapter 2

Overview of Machine Learning Techniques for Unmonitored Prediction

2.1 Introduction

Numerous ML approaches have been explored for environmental variable time series prediction in unmonitored sites in recent years that span a variety of methods and applications in hydrology and water resources engineering. Though most of these approaches were developed in hydrology due to the wealth of streamflow data compared to other variables like river and lake water quality, these efforts are expanding as data collection and modeling continue to advance. However, often these approaches are not compared in detail with each other or sufficiently benchmarked such that researchers know what to use in a given prediction task. In this paper, we provide a comprehensive and systematic review of these techniques and demonstrate the progress that has been made across different environmental applications using them. We also enumerate the gaps and opportunities that exist for advancing research in this promising direction.

We organize this chapter as follows. Section 2.2.1 first describes different broad-scale ML frameworks built on all available training sites that can be used to generalize to unmonitored locations. Then, Section 2.2.2 describes models built on a subset of

all available sites based on metrics like similarity to the target site(s). Section 2.2.3 covers transfer learning scenarios using individual models, and Section 2.2.4 covers a cross-cutting concepts in knowledge-guided machine learning. The chapter concludes with Section 2.3 with a summary and discussion of the surveyed works.

Scope: We note that the works surveyed in this chapter are limited to the scenario of predicting in a location lacking *any* observations of the target environmental variable, though there are some recent innovations for significant advancement in ML modeling within the field for the case of a *few* data points being available (e.g. Ghosh et al. [38] for streamflow, Chen et al. [39] and Jia et al. [20] for stream temperature). We also do not cover the many efforts made in recent years progressing the traditional and ML-based methods of regionalizing process-based hydrological models. Guo et al. [11] provide a recent extensive survey on this topic. Lastly, we also exclude remote sensing applications to either estimate key water quality parameters and other environmental variables or calibrate either process-based or ML models. Although it is a promising direction for modeling previously unmonitored inland water bodies, remote sensing faces many limitations including atmospheric effects, measurement frequency, and insufficient resolution for smaller water bodies like rivers all of which present challenges to increasing the scale and robustness of remote sensing applications [40]. This has been covered to a great extent in a number of surveys [41, 42, 43, 40].

2.2 Machine Learning Frameworks for Unmonitored Prediction

In this section we enumerate different methodologies for unmonitored prediction across different applications in water resources time series modeling. Generally, the process of developing ML models for unmonitored prediction first involves modeling a set of entities (e.g. stream gauge sites, lakes) with monitoring data of the target variable (e.g. discharge, water quality). Then, the knowledge, data, or models developed on those systems are used to predict the target variable on entities with no monitoring data available. Across these different methods, the assumption is made that the input data used for prediction consists of both dynamic physical drivers (e.g. daily meteorology)

and site-specific characteristics of each entity (e.g. geomorphology, climatology, pedology, land use). These characteristics across the literature are also called attributes, traits, or properties.

This type of model is known as an entity-aware model [44, 45], which attempts to incorporate inherent characteristics of different entities to improve prediction performance. However, varied methodologies have been developed which differ both in how these characteristics are used to improve performance and also how entities are selected and used for modeling. Section 2.2.1 covers the different methods of constructing single broad-scale models using all available sites, Section 2.2.2 looks at how broad-scale models are built from subgroups of sites deemed relevant or similar to the target unmonitored sites, and 2.2.3 covers various transfer learning of specific models (e.g. single site) that have been applied. Lastly, 2.2.4 covers a cross-cutting theme of integrating ML with domain knowledge and process-based models for unmonitored prediction.

2.2.1 Broad-scale models using all available entities

Historically, the most successful process-based models have been calibrated to specific locations, which is fundamentally different from a broad-scale or regionalized model built on a large number of sites that must differentiate between dynamic behaviors in different sites (e.g., ecology, geology, pedology, topography, geometry) [46, 47]. The objective of broad-scale modeling is to learn and encode these differences such that differences in site characteristics translate into appropriately heterogeneous hydrologic behavior. Usually the choice is made to include all possible sites or entities in building a singular broad-scale model, but many studies differ in how the ML framework leverages the site characteristics. The following subsections enumerate different ways of incorporating site characteristics into broad-scale models that use all available entities, covering direct concatenation of site characteristics and dynamic features, encoding of characteristics using ML, and the use of graph neural networks to encode dependencies between sites.

Direct concatenation broad-scale model

When aggregating data across many sites for a entity-aware broad-scale model, it is common to append site characteristics directly with the input forcing data directly before feeding it to the ML model. This is a simple approach that does not require novel

ML architecture, and is therefore very accessible for researchers. Though recurrent neural network-based approaches like LSTM commonly seen in environmental time series prediction are not built to incorporate static inputs, many applications in a variety of disciplines have found success in repeating static values over each timestep through this concatenation process [48, 49, 50], and we see similar results in water resources prediction. In a landmark result for streamflow modeling, Kratzert et al. [44] show an LSTM with directly concatenated site characteristics and dynamic inputs built on 531 geographically diverse catchments within the Catchment Attributes and Meteorology for Large-sample Studies (CAMELS) dataset was able to predict more accurately on unseen data on the same 531 test sites than state-of-the-art process-based models calibrated to each basin individually[1]. Given the success of the model, that study was expanded to the scenario of predicting in unmonitored stream sites [51], where they found the accuracy of the broad-scale LSTM with concatenated features in ungauged basins was comparable to calibrated process-based models in gauged basins. Arsenault et al. [52] and Jiang et al. [53] further show a similar broad-scale LSTM can outperform the state-of-the-art regionalization of process-based models for unmonitored prediction in the United States, and similar results are seen in Russian [54], Brazilian [55], and Korean [56] watersheds. Similar results have also been seen in the prediction of other environmental variables like continental-scale snow pack dynamics [57], hydrological baseflow [58], dissolved oxygen in streams [59], and lake surface temperature [4].

The previously mentioned approaches primarily cover averaged daily value prediction, but accurate predictions of extremes (e.g. very high flow events or droughts) remains an outstanding and challenging problem in complex spatiotemporal systems [60]. In particular in unmonitored systems, prediction of extremes is vital for the design, operation, public safety, and maintenance of water resources systems [61]. This a longstanding fundamental challenge in catchment hydrology [9] where typically the approach has been to subdivide the study area into fixed, contiguous regions which are used to regionalize floods or low flows from process-based models for all catchments in a given area. As recent ML and statistical methods are shown to outperform process-based models for the prediction of extremes [62, 63], opportunities exist to apply

[1]In this paper we use the term "entity-aware" in the context of a general way of modeling a large number of entities with inherent characteristics with ML, as opposed to the "entity-aware long short-term memory" architecture in Kratzert et al. [44].

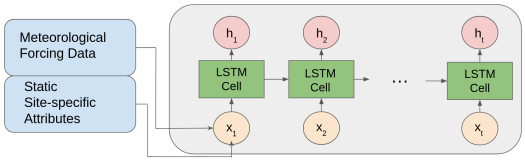


Figure 2.1: Example of an long short-term memory (LSTM) network model with directly concatenated site characteristics and dynamic inputs

broad-scale entity-aware methods in the same way as daily averaged predictions. Initial studies using broad-scale models with concatenated inputs for peak flood prediction show that these methods can also be used to predict extremes. Rasheed et al. [64] build a peak flow prediction model that combines a "detector" LSTM that determines if the meteorological conditions pose a flood risk, with an entity-aware ML model for peak flow prediction to be applied if there is a risk. They show that building a model only on peak flows and combining it with a detector model improves performance over the broad-scale LSTM model trained to predict all varieties of flow (e.g. Kratzert et al. [51]).

Based on these results, we see site characteristics often contain sufficient information to differentiate between site-specific dynamic behaviors for a variety of prediction tasks. This challenges a longstanding hydrological perspective that transferring models and knowledge from one basin to another requires that they must be functionally similar [29, 46], since these broad-scale models are built on a large number of heterogeneous sites.

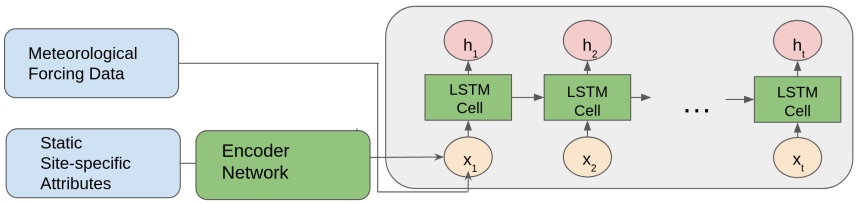
Concatenation of encoded site characteristics for broad-scale models

Though recurrent neural network models like the LSTM have experienced success with direct concatenation of static and dynamic features, other methods have been developed that encode static features to improve accuracy or increase efficiency. One way is to use two separate neural networks; the first learns a representation of the static characteristics using an encoding neural network (e.g. autoencoder), and the second takes that encoded representation at each time-step along with dynamic time-series inputs to predict the target using a time series ML framework (e.g. LSTM) [65, 50, 66, 48]. The idea is to extract the information from characteristics that accounts for data heterogeneity across

multiple entities. This extraction process is independent from the LSTM or similar time series model handing the dynamic input, and therefore can be flexible in how the two components are connected. Examples to improve efficiency include, (1) static information may not be needed at every time step and be applied only at the time step of interest [48], or (2) the encoding network can be used to reduce the dimension of static features prior to connecting with the ML framework doing the dynamic prediction in order. In terms of performance, works from multiple disciplines have found this approach improves accuracy over the previously described direct concatenation approach [48, 66, 49].

The use of an additional encoding network has been seen in hydrological and water resources applications. Tayal et al. [66] demonstrate this in lake temperature prediction using an invertible neural network in the encoding step, showing slight improvement over the static and dynamic concatenation approach. It has also been shown in streamflow prediction that the encoder network can be used either on the site characteristics [53] or also on partially available soft data like soil moisture or flow duration curves [67].

Figure 2.2: Example of a combination static feature encoder neural network with a long short-term memory (LSTM) network model



Broad-scale graph neural networks

The majority of works reviewed in this chapter treat entities as systems that exist independently from each other (e.g. different lakes, different stream networks). However, many environmental and geospatial modeling applications exhibit strong dependencies and coherence between systems [6]. These dependencies can be real, interactive physical connections, or a coherence in dynamics due to certain similarities regardless of whether the entities interact. For example, water temperature in streams is affected by

a combination of natural and human-involved processes including meteorology, interactions between connected stream segments within stream networks, and the process of water management and timed release reservoirs. Similar watersheds, basins, or lakes may also exhibit dependencies and coherence based on characteristics or climatic factors [68, 69, 70, 71]. Popular methods like the previously described broad-scale models using direct concatenation of inputs (Section 2.2.1) offer no intuitive way to encode interdependencies between entities (e.g. in connected stream network) and often ignore these effects. Researchers are beginning to explore different ways to encode these dependencies explicitly by using graph neural networks (GNNs) for broad-scale modeling of many entities. The use of GNNs can allow the modeling of complex relationships and interdependencies between entities, something traditional feed-forward or recurrent neural networks cannot do [72]. GNNs have seen a surge in popularity in recent years for many scientific applications and several extensive surveys of GNNs are available in the literature [73, 74, 75, 72]. Hydrological processes naturally have both spatial and temporal components, and in the same way that the LSTM architecture exploits temporal patterns and dependencies, GNNs attempt to exploit the spatial connections, causative relations, or dependencies between similar entities. Recent work has attempted to encode stream network structure within GNNs to capture spatial and hydrological dependencies for applications like drainage pattern recognition [76], groundwater level prediction [77], rainfall-runoff or streamflow prediction [78, 79, 80, 81, 82, 83], lake temperature prediction [84], and stream temperature prediction [85, 86, 39].

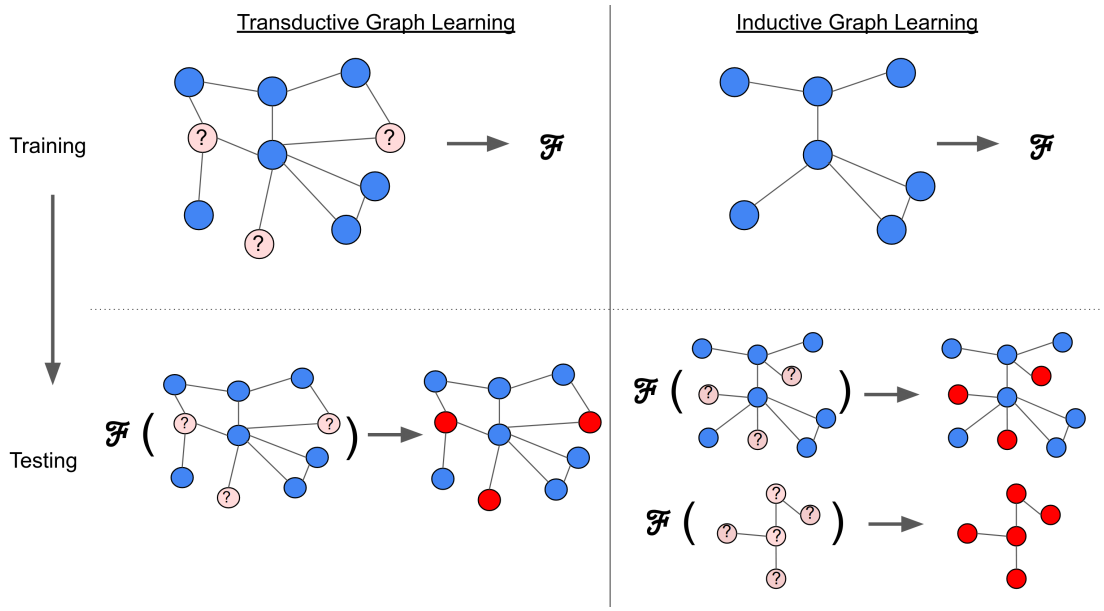
In hydrology, there are two intuitive methods for construction of the graph itself. The first is geared towards non-interacting entities, building the graph in the form of pair-wise similarity between entities, whether that be between site characteristics [81], spatial locations [87, 88] (e.g. latitude/longitude), or both [89]. The second type is geared more toward physically interacting entities, for example the upstream and downstream connections between different stream segments in a river network [20] or connections between reservoirs with timed water releases to downstream segments [90]. Relying only on the characteristics or location for graph construction in the non-interacting case more easily allows for broad-scale modeling because it can model spatially disconnected entities, however it introduces no *new* information (e.g. physical connectivity) beyond what the previously described direct concatenation-based methods use since the static

characteristics would be the same. However, performance could still improve and interpretations of encodings within a graph framework could yield new scientific discovery since pairwise encodings between entities can be directly extracted. Graphs built using real physical connections between entities (e.g. stream segments in a stream graph), on the other hand, allow for the capability to learn how information is routed through the graph and how different entities physically interact with each other. So far, this has only been seen on stream modeling using stream network graphs [20, 79, 91].

There are two different classes of GNN models, transductive and inductive, that differ in how the graph is incorporated in the learning process. Depending on how the graphs are constructed, one of these is more natural than the other. A conceptual depiction of both is shown in Figure 2.3. The key aspect of transductive GNNs is that both training and testing entities must be present in the graph during training. A prerequisite for this approach is that the test data (e.g. input features in unmonitored sites) is available during model training, and one key aspect is that the model would need to be completely re-trained upon the introduction of new test data. Even if the training data is unchanged prior to re-training, introducing new test nodes in the graph can affect how information is diffused to each training node during optimization [92]. This type of approach is generally preferred for river network modeling given the often unchanging spatial topology of the sub-basin structure which is known *a priori* [20, 80, 93]. Graph connections from the test nodes to the training nodes in a transductive setting can be used either in the training or prediction phase, or both [94]. Inductive GNNs on the other hand, are built using only training entities and allow for new entity nodes to be integrated during testing. For applications that continuously need to predict on new test data, inductive approaches are much more preferred. New entity nodes are able to be incorporated because inductive frameworks also learn an information aggregator that transfers the necessary information from similar or nearby nodes to predict at nodes unseen during training. As shown in Figure 2.3, inductive graph learning can either be done on nodes that connect with training set nodes in the graph or those that are disconnected. Inductive GNNs can be understood as in the same class as more standard supervised ML models like LSTM or feed-forward neural networks, where they are able to continuously predict on new test data without the need for re-training.

Though most of these works using GNNs in water resources have not focused on the

Figure 2.3: Conceptual example of transductive and inductive graph learning. In both left and right panels, $\mathscr{F}$ is a model learned during training. Blue and red nodes represent entities with data for use in training and test entities without any data respectively. In transductive graph learning, the model has access to nodes and edges associated with test entities during training, but no new nodes can be introduced during testing. In inductive graph learning, the model is trained on an initial graph without any knowledge of the test entities, but the model can generalize to any new nodes during testing.



prediction in unmonitored sites, a few notably do. Sun et al. [81] apply different types of spatiotemporal GNNs to the problem of unmonitored streamflow prediction including three transductive GNN methods, two variants of the ChebNet-LSTM [95] and then the Graph Convolutional Network LSTM (GCN-LSTM) [96]. They compare this with an inductive GNN in the GraphWaveNet [97]. In all cases, the graph is initially constructed as an adjacency matrix containing the pairwise Euclidean distance between stream sites using the sites' characteristics. Importantly, all three of these models simplify to direct concatenation-based models described in Section 2.2.1 if the graph convolution-based components are removed (See Figure S2 in Sun et al. [81] for a visualization). For ChebNet-LSTM and GCN-LSTM, these would simplify to a traditional LSTM, and for GraphWaveNet, it would simplify to a gated temporal convolution network

(TCN). They found that the for the transductive case, both ChebNet-LSTMs and GCN-LSTM performed worse in terms of median performance across basins than the standard LSTM and GraphWaveNet was the only one that performed better. GraphWaveNet, the only GNN also capable of doing inductive learning, also performed better in the inductive case than standard LSTM. Jia et al. [20] take a different spatiotemporal GNN approach for stream temperature where they construct their graph by using stream reach lengths and upstream and downstream connections to construct a weighted adjacency matrix. They found their GNN pre-trained on simulation data outperformed both a non-pre-trained GNN and a baseline LSTM model. Based on these results we see than encoding both broad characteristics-based dependencies as well as physical interaction and connections-based dependencies in streams for GNNs has the potential for improved performance over existing standard deep learning models like the feed-forward artificial neural network (ANN) or LSTM.

Further research in this area could explore different ways of constructing the adjacency matrix based on the application and available data. An example of a domain-informed method for graph construction can be seen in Bao et al. [86] for stream temperature prediction in monitored sites, where they leverage partial differential equations of underlying heat transfer processes to estimate the graph structure dynamically. This graph structure is combined with temporal recurrent layers to improve prediction performance beyond existing process-based and ML approaches. Dynamic temporal graph structures like this are common in other disciplines like social media analysis and recommender systems, but have not been widely used in the geosciences [98]. For time series water resources modeling, it may be important to structure for dynamic dependencies across temporal scales. Another avenue that could be explored is the issue of imbalances in training data and how this affects GNN-based models. For many prediction scenarios in water resources, data is concentrated in certain regions or systems and many more systems are sparsely monitored or unmonitored, and this could lead to very densely-connected graph regions alongside very-sparse or disconnected graph regions. Various regularization techniques have been proposed to deal with imbalanced training data in other applications that have not been attempted in geosciences [99, 100]. Incorporating graphs into ML architecture design also has the added bonus of making the previously black-box algorithm more interpretable, a desirable but typically missing

component of ML models used in environmental modeling. Interpretable ML techniques could be used to discover patterns in the calculated distances and similarities within the graph that are relevant to domain scientists. This is explored further in Chapter 6.

2.2.2 Broad-scale models using a subgroup of entities

In the previous subsection, models were built with all available data to create a broad-scale model. However, using the entirety of available data is not always optimal. Researchers may consider selecting only a subset of entities for training for a variety of reasons including (1) the entire dataset may be imbalanced such that performance diminishes on minority system types [101], or (2) some types of entities may be noisy, contain erroneous or outlier data, or have varying amount of input data. Traditionally in geoscientific disciplines like hydrology, stratifying a large domain of entities into multiple homogeneous subgroups or regions that are "similar" is common practice, and this is based on evidence in process-based modeling that grouping heterogeneous sites for regionalization can negatively affect performance extrapolating to unmonitored sites [102, 103]. Therefore, it remains an open question whether using all the available data is the optimal approach for building training datasets for prediction in unmonitored sites. Copious research has been done investigating various homogeneity criteria trying to find the best way to group sites for these regionalization attempts [104, 105], and many recent approaches also leverage ML for clustering sites (e.g. using k-means [106, 107]) prior to parameter regionalization [108, 46].

Many studies continue the practice of using subgroups of sites when building broad-scale models using ML. For example, Araza et al. [109] demonstrate that a principal components analysis-based clustering of 21 watersheds in the Luzon region of the Philippines outperforms an entity-aware broad-scale model built on all sites together for daily streamflow prediction. Chen et al. [110] cluster weather stations by mean climatic characteristics when building LSTM and temporal convolution network models for predicting evapotranspiration in out-of-sample sites claiming models performed better on similar climatic conditions. Additionally for stream water level prediction in unmonitored sites, Corns et al. [111] group sites based on the distance to upstream and downstream gauges. The water levels from the upstream and downstream gauges are also used as input variables. The reasoning behind not grouping everything together

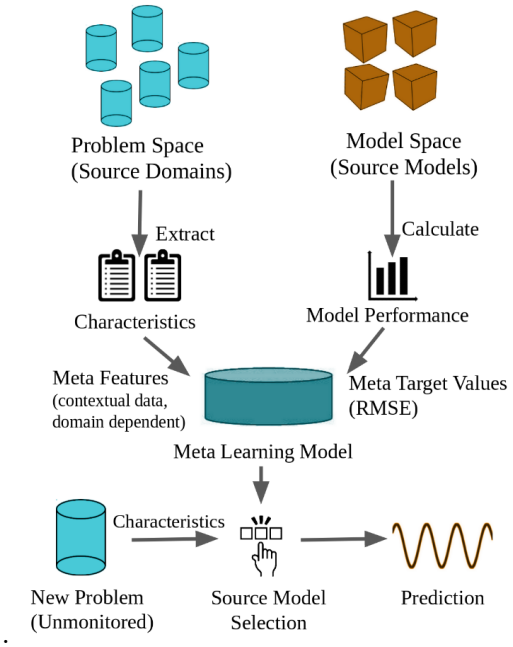
was that they did not want predictions to be made using input data from upstream and downstream gauges regardless of proximity. Also, the peak flood prediction model described in Section 2.2.1 divides the models and data across the 18 hydrological regions in the conterminous US as defined by USGS [23]. However, it is important to note that it remains to be seen how selecting a subgroup of entities as opposed to using all available data fairs in different prediction applications because much of this work does not compare the performances of both these cases.

However, when viewed through the lens of modern data-driven modeling and results from the previous section, evidence suggests deep learning methods in particular may benefit from pooling large amounts of heterogeneous training data. Fang et al. [112] demonstrate this effect of "data synergy" on both streamflow and soil moisture modeling in gauged basins showing that deep learning models perform better when fed a diverse training dataset spanning multiple regions as opposed to homogeneous dataset on a single region even when the homogeneous data is more relevant to the testing dataset and the training datasets are the same size.

2.2.3 Transfer learning

Transfer learning is a powerful technique for applying knowledge learned from one problem domain to another, typically to compensate for missing, nonexistent, or unrepresentative data in the new problem domain. The idea is to transfer knowledge from an auxiliary task, i.e., the source system, where adequate data is available, to a new but related task, i.e., the target system, often where data is scarce or absent [113, 114]. Situations where transfer learning may be more desirable than broad-scale modeling approaches listed in previous sections include when (1) a set of highly tuned and reliable source models may already be available, (2) local source models are more feasible computationally or more accurate than broad-scale models when applied to unmonitored systems, or (3) broad-scale models may need to be transferred and fine tuned to a given region or system type more similar to an unmonitored system. In the context of geoscientific modeling, transfer learning for ML is analogous to calibrating process-based models in well-monitored systems and transferring the calibrated parameters to models for unmonitored systems, which has shown success in hydrological applications [115, 116]. Deep learning is particularly amenable to transfer learning because it can

Figure 2.4: Process diagram of the Meta Transfer Learning framework. Models are first built from data-rich source domains. The metamodel is trained using characteristics extracted from the source domains to predict the performance metrics from transferring models between source domains. Then given a target system or domain, the metamodel is able to output a prediction of how well each of the source models will perform on the target system. Adapted from Willard et al. [1]



make use of massive datasets from related problems and alleviate data paucity issues common in applying data-hungry deep neural networks to environmental applications [16, 117]. Transfer learning using deep learning has shown recent success in water applications such as flood prediction [118, 119], soil moisture [120], and lake and estuary water quality [121, 1].

Transfer learning can also be a capable tool for prediction in unmonitored sites [122]. However, in most transfer learning applications, it is assumed that least some data is available in the target system for fine-tuning a model [114, 123]. The specific case of transferring to a system or task without any training data is also known as "zero-shot learning" [124], where only the inputs or a high level description may be available for the testing domain that does not contain any target variable values. This is a significantly more challenging problem because taking a pre-trained model from a data-rich

source system and fine-tuning it on the target system is not possible, and instead other contextual data about the source and target systems must be used. For the case of environmental variable prediction in unmonitored sites, we often only have the dynamic forcing data and the characteristics of the target system available. The following subsections cover different ways researchers have addressed the zero-shot transfer learning problem for water resources prediction.

Choosing which model to transfer

A central challenge in zero-shot transfer learning is determining which model to transfer from a related known task or how to build a transferable model. Previous work on streamflow prediction has based this purely on expert knowledge. For example, Singh et al. [125] operates under the assumption that the model must be trained on other basins in the same climatic zone and at least some of the source basin's geographical area must have similar meteorological conditions to the target basin. Other work has naively transferred models from data-rich regions to data-poor regions without any analysis of the similarity between the source and target regions. Le et al. [126] transfer ML streamflow models built on North America (987 catchments), South America (813 catchments), and Western Europe (457 catchments); to data-poor South Africa and Central Asian regions. They transfer these models as-is and do not take into account any of the sparse data in the data-poor region or the similarity between regions and find that the local models trained on minimal data outperform the models from data-rich regions. Attempts have also been made to use simple expert-created distance-based metrics (e.g. Burn et al. [127]) using the site characteristic values [128]. However, it is reasonable to think that a data-driven way to inform model building based off both the entity's characteristics and past modeling experiences would be possible.

The idea of building or selecting a model based off past modeling experience is a type of *meta-learning* [129, 130]. One meta-learning strategy for model selection is to build a metamodel to learn from both the model parameters of known tasks (with ground truth observations) and the correlation of known tasks to zero-shot tasks [131]. For example, in lake temperature modeling, Willard et al. [1] use meta-learning for a model selection framework where a metamodel learns to predict the error of transferring a model built on a data-rich source lake to an unmonitored target lake. A diagram of the

approach is shown in Figure 2.4. They use variety of contextual data is used to make this prediction, including (1) characteristics of the lake (e.g. maximum depth, surface area, clarity etc., (2) meteorological statistics (e.g. average and standard deviation of air temperature, wind speed, humidity etc., (3) simulation statistics from an uncalibrated process-based model applied to both the source and target (e.g. differences in simulated lake stratification frequency), and (4) general observation statistics (e.g. number of training data points available on the source, average lake depth of measured temperature, etc). They show significantly improved performance predicting temperatures in 305 target lakes treated as unmonitored in the Upper Midwestern United States relative to the uncalibrated process-based General Lake Model [132], the previous state-of-the-art for broad-scale lake thermodynamic modeling. This was expanded to a streamflow application in Ghosh et al. [133] with numerous methodological adaptations. First, instead of using the site characteristics as-is they use a sequence autoencoder to create embeddings for all the stream locations by combining input time series data and simulated data generated by a process-based model. This adaptation alleviated a known issue in the dataset that the site characteristics were commonly incomplete and inaccurate. They also use a clustering loss function term in the sequence autoencoder to guide the model transfer, where source systems are selected based on available source systems within a given cluster of sites as opposed to building an ensemble with a set number source sites. They show on streams within the Delaware River Basin that this outperforms the aforementioned simpler meta transfer learning frameworks on sites based on [1].

Transferring and localizing regional models

Another transfer learning strategy in geoscientific modeling that can also be based on pre-training is to localize a larger-scale or more data-rich regional or global model to a specific location or subregion. This variant of transfer learning has seen success in deep learning models for applications like soil spectroscopy [134, 135] and snow cover prediction [136, 137]. However, these strategies have seen mixed success in hydrological applications. Wang et al. [57] show that localizing an LSTM predicting continental-scale snowpack dynamics to individual regions across the United States had insignificant benefit over the continental-scale LSTM. Xiong et al. [138] show a similar result for

the prediction of stream nitrogen export, where the individual models for the 7 distinct regions across the conterminous United States transferred to each other never could outperform the continental-scale model using all the data. Also, Lotsberg et al. [139] show that streamflow models trained on CAMELS-US (United States) transfer to CAMELS-GB (Great Britain) about as well as a model trained on the combined data from US and GB, and models trained on CAMELS-GB transfer to CAMELS-US about as well as a model using the combined data. They also show that the addition of site characteristics is not beneficial in transfer learning tasks, but acknowledge this could be due to the way data is normalized prior to training. Based on these results, it is possible that the entity-aware model using all available data is already learning to differentiate between different regions or types of sites on its own and fine-tuning to more similar sites based on expert knowledge may be less useful. However, this remains to be seen in most hydrological and water resources prediction tasks.

Unsupervised Domain Adaptation

Domain adaptation methods are a subset of transfer learning algorithms that attempt to answer the question, *how can a model both learn from a source domain and learn to generalize to a target domain*? Often domain adaptation seeks to minimize the risk of making errors on the target data, not necessarily the source data as in traditional supervised learning. Unsupervised domain adaptation (UDA) in particular focuses on the case of the target domain being void of target data. Similar to the types of graph neural networks mentioned in Section 2.2.1, review papers have divided transfer learning algorithms into the categories, (1) *inductive* transfer learning where the source and target tasks are different and at least some labeled data from the target task is required to induce a model, (2) *transductive* transfer learning where the source and target tasks are the same but from different feature space domains and zero labeled data is available from the target domain, and (3) *unsupervised* transfer learning where no labeled data is available in both the source and target domains [140, 141]. UDA specifically lies in the transductive transfer learning scenario, and usually involves using the input data from the target or testing task during the training process, in addition to the source data. Researchers can employ UDA methods when attempting to account for differences in the source and target tasks and datasets. However, UDA methods can differ in terms

of what they are accounting for. Commonly UDA methods attempt to account for the difference in input feature distribution shifts between the source and task, but other methods attempt to account to the difference in label distribution or conditional distributions. This differs from previous approaches we have mentioned like the broad-scale models which generally ignore the input data from the testing sites, meta transfer learning which uses test data inputs during model selection but not during training, and localizing regional models which uses available data from regions containing the test sites but not any data from the test sites themselves. UDA has seen success in many disciplines including computer vision [142, 143], robotics [144, 145], natural language processing [146], and fault diagnostics [147] but applications of UDA in hydrology are limited. In the only current example, Zhou et al. [148] introduce a UDA framework for unmonitored flood forecasting that involves a two-stage adversarial learning approach. The model is first pre-trained on a large sample source dataset, then they perform adversarial domain adaptation using an encoder to map the source and target inputs to the same feature space and learn the difference between the source and target datasets. They show this method is effective in flood forecasting across the Tunxi and Changhua flood datasets spanning Eastern China and Taiwan. Currently UDA that accounts for a shift in label distribution (real or synthetic) has not been attempted in hydrological prediction, and future research on UDA in hydrology will need to consider whether to account for either input or label distribution shift between entities and systems.

2.2.4 Cross cutting theme: knowledge-guided machine learning

There is a growing consensus that solutions to complex nonlinear environmental and engineering problems will require novel methodologies that are able to integrate traditional process-based modeling approaches with state-of-the-art ML techniques, known as *Knowledge-guided machine learning* (KGML) [149] (also known as *Physics-guided machine learning* or *Physics-informed machine learning* [150, 151, 152]). These techniques have been demonstrated to improve prediction in many applications including lake temperature [153, 17], streamflow [154, 155, 156], groundwater contamination [157], and

water cycle dynamics [158] among others. Willard et al. [150] divide KGML methodologies into four classes; (i) physics-guided[2] loss function, (ii) physics-guided initialization, (iii) physics-guided design of architecture, and (iv) hybrid physics-ML modeling. Many of these methods are helpful in the case of unmonitored prediction, since known physics or existing models can exist in the absence of observed target data. Note that KGML is a cross-cutting theme, as its principles can be integrated into either of the previously described broad-scale modeling and transfer learning. The benefits we see from KGML as a class of standalone techniques can also help address resource efficiency issues in building both broad-scale entire-aware models and also source models in transfer learning while maintaining high predictive performance, training data efficiency, and interpretability relative to traditional ML approaches [150].

The field of KGML is rapidly advancing, and given the numerous applications we see in unmonitored prediction we include the following discussion on the different ways of harnessing KGML techniques in a given physical problem that has traditionally been simulated using process-based models. The following three subsections are divided based on how KGML techniques are used to either replace, augment, or recreate an existing process-based model. Section 6.5 further expands on this discussion by addressing the role of KGML in the future of unmonitored prediction and open questions that exist.

Guiding ML with domain knowledge: KGML loss functions, architecture, and initialization

Traditional mechanistic or process-based models for simulating environmental variables oftentimes provide an incomplete representation of the target variable due to simplified or missing physics. Though a key benefit of pure ML is the flexibility to literally fit any dataset as well as not being beholden to the causal structure that process-based models are, its inability to make use of process-based knowledge can lead to negative effects like sample inefficiency, inability to generalize to out-of-sample scenarios, and physically inconsistent solutions. When building an ML model as a replacement for a process-based model, there are three primary methods that researchers should consider to to guide the ML model with domain knowledge for improved predictive performance;

[2]In this paper, we use the term "knowledge-guided" as opposed to "physics-guided" but they are used interchangeably in the literature.

KGML loss function terms, architecture, and initialization.

KGML loss function terms can constrain model outputs such that they conform to existing physical laws or governing equations. Steering ML predictions towards physically consistent outputs has numerous benefits. For unmonitored prediction, the major benefit of informed loss function terms is that often the computation requires no observation data. Therefore, optimizing for that term allows for the inclusion of unlabeled data in training, which is often the only data available. Other benefits include that the regularization by physical constraints can reduce the possible search space of parameters, and there is also possibility to learn with less labeled data while also ensuring the consistency with physical laws during optimization. KGML loss function terms have also shown that models following desired physical properties are more likely to be generalizable to out-of-sample scenarios [17], and thus become acceptable for use by domain scientists and stakeholders in water resources applications. Loss function terms corresponding to physical constraints are applicable across many different types of ML frameworks and objectives, however most of these applications have been in the monitored prediction scenario (e.g. lake temperature [153, 17, 159], lake phosphorous [160], subsurface flow [161]). In this survey, we find only one work using informed loss function terms within a meta transfer learning framework for lake temperature modeling [1] incorporating conservation of energy relating the ingoing and outgoing thermal fluxes into the lake.

Another direction is to use domain knowledge to directly alter a neural network's architecture to implicitly encode physical consistency or other desired physical properties. However, KGML-driven architecture optimizing for physical consistency is usually understood as a hard constraint since the consistency is hardcoded into the model, whereas KGML loss function terms are a soft constraint that can depend on optimization and weights within the loss function. Other benefits from KGML loss function terms are also experienced by KGML-driven model architecture, including reducing the search space and allowing for better out-of-sample generalizability. KGML-driven model architectures have shown success in hydrology, however it has been limited to the monitored prediction scenario. Examples include Jiang et al. [53] where they show a rainfall-runoff process model can be embedded as a special recurrent neural layers in a deep learning architecture, Daw et al. [162] where they show a physical intermediate neural network

node as part of a monotonicity-preserving structure in the LSTM architecture for lake temperature, and more examples in the Willard et al. [150] KGML survey. However, there is nothing preventing these approaches from being applied in the unmonitored scenario.

Lastly, if process-based model output is already available, for instance the National Water Model streamflow outputs [163], FLake model lake surface temperature outputs within ERA5 [2], or PRMS-SNTemp simulated stream temperature [164], this data can be used to help initialize an ML model using pre-training which is known as KGML initialization. This is arguably the most accessible KGML method since there is no alteration to the ML itself. By pre-training, the ML model can learn to emulate the process-based model prior to seeing training data in order to accelerate or improve the primary training. Numerous studies in water resources perform KGML-based model initialization by making use of process-based model output to inform ML model building, either to create site-specific embeddings used for similarity calculation in meta transfer learning [133], as a pre-training stage for source models in meta transfer learning [1], or as a pre-training stage for entity-aware broad-scale models [165, 166].

Augmenting process models with ML using hybrid process-ML models

In many cases certain aspects of process-based models may be sufficient but researchers seek to use ML in conjunction with an operating process-based to address key issues. Examples include where (1) process-based model outputs or intermediate variables are useful inputs to the ML model, (2) a process-based model may model certain intermediate variables better than others that could utilize the benefits of ML, or (3) optimal performance involves choosing between process-based models and ML models, based on prediction circumstance in real time. Using both the ML model and a process-based model *simultaneously* is known as a hybrid process-ML model and is the most commonly used KGML technique for unmonitored prediction. In the Willard et al. [150] survey of KGML methods, they define hybrid models as either process and ML models working together for a prediction task, or a subcomponent of a process-based model being replaced by an ML model. This type of KGML method is also very accessible for domain scientists since it requires no alterations to existing ML frameworks. In this chapter, we do not cover the large body of work of ML predictions of process-based

model parameters since these methods have been outpaced by ML for predictive performance and tend to extrapolate to new locations poorly [167], but summaries can be found in Reichstein et al. [6] or Xu et al. [168].

The most common form of hybrid process-ML models in hydrological and water resources engineering is known as residual modeling. In residual modeling, a data-driven model is trained to predict a corrective term to the biased output of a process-based or mechanistic model. This concept goes by other names such as error-correction modeling, model post-processing, error prediction, compensation prediction, and others. Correcting these residual errors and biases has been shown to improve the skill and reliability streamflow forecasting [169, 170], water level prediction [171], and groundwater prediction [172]. When applying residual modeling to unmonitored prediction, the bias correcting ML model must be trained on either a large number of sites or sites similar to the target site. Hales et al. [173] demonstrate a framework to build a residual model for stream discharge prediction with the GEOGloWS ECMWF Streamflow Model that selects similar sites based on the dynamic time warping and euclidean distance time series similarity metrics. For unmonitored sites, they substitute simulated data instead of the observed data and show a substantial reduction in model bias in ungauged subbasins.

A slight alteration to the residual model is a hybrid process-ML model that takes an ML model and adds the output of a process-based model as an additional input. This adds a degree of flexibility to the modeling process compared to the standard residual model as the residual error is not modeled explicitly and multiple process-based model outputs can be used at once. Karpatne et al. [159] showed that adding the simulated output of a process-based model as one input to an ML model along with input drivers used to drive the physics-based model for lake temperature modeling can improve predictions, and a similar result was seen in Yang et al. [174] augmenting a global hydrological models-based flood simulation model for flood prediction. This has been applied to unmonitored prediction recently as well, with Noori et al. [166] using the output of SWAT (Soil & Water Assessment Tool [175]) as an input to a feed-forward neural network for predicting monthly nutrient load prediction in unmonitored watersheds. They find that the hybrid process-ML model has greater prediction skill in unmonitored sites than the SWAT model calibrated at each individual site.

Another simple way to combine process-based models with ML models is through

multi-model ensemble approaches that combines the predictions of two or more types models. Ensembles can both provide more robust prediction and allow quantification and reduction of uncertainty. Multiple studies in hydrology have shown that using two or more process-based models with different structures improves performance and reduce prediction uncertainty in ungauged basins [176, 177]. Razavi et al. [178] show an ensemble of both ML models and process-based models for streamflow prediction, which further reduced prediction uncertainty and outperformed individual models. However this study is limited to building a model for an ungauged stream site using only the three most similar and closely located watersheds, as opposed to more comprehensive and inclusive datasets like CAMELS.

Comparisons between different types of hybrid models are not commonly seen, as most studies tend to stick to one method. However, different types should be considered based on the context of the task. For example, if multiple process-based models are available then multi-model ensemble or using multiple process-based outputs as inputs to an ML model can be considered. Or, if part of the physical process is well-known and modeled compared to more uncertain components, researchers can consider replacing only part of the process-based model with an ML model component. In one study highlighting different hybrid models, Frame et al. [179] compare three approaches, (1) LSTM residual models correcting the National Water Model (NWM), (2) a hybrid process-ML model using an LSTM that takes the output of the NWM as an additional input, and (3) a broad-scale entity-aware LSTM like we have described in Section 2.2.1. They find that in the unmonitored scenario, the third approach performed the best which leads to the conclusion that the output from the NWM actually impairs the model and prevents it from learning generalizable hydrological relationships. Additional research is required to address when hybrid modeling is beneficial for unmonitored prediction, since there are often numerous process-based models and different ways to hybridize modeling for a given environmental variable.

Building differentiable and learnable process-based models

Models like the broad-scale entity-aware LSTM have revolutionized environmental variable time series prediction accuracy. However, they often lack interpretability and

clarity about physical processes that can be used to answer more specific scientific questions about internal processes and causation. Numerous efforts have been made to address this issue, building models that have equal or greater accuracy but with increased interpretability, transparency, and explainability using the principles of differentiable process-based (DPB) modeling [180, 181, 182]. The main idea of DPB models is to keep an existing geoscientific model's structure but replace the entirety of its components with differentiable units (e.g. ML). From an ML point of view, it can be viewed as a domain-informed structural prior resulting in a modular neural network with physically meaningful components. This differs from the previously described hybrid process-ML methods that include non-differentiable process-based models or components. One recent example in hydrological flow prediction is shown in Feng et al. [181], though similar models have been used in other applications like earth system models [183] and molecular dynamics [184]. The DPB model proposed by Feng et al. [181] starts with a simple backbone hydrological model (Hydrologiska Byråns Vattenbalansavdelning model [185]), replaces parts of the model with neural networks, and couples it with a differentiable parameter learning framework (see Figure 1 in Feng et al. [181] for a visualization). Specifically, the process model structure is implemented as a custom neural network architecture that connects units in a way that encodes the key domain process descriptions, and an additional neural network is appended to the aforementioned process-based neural network model to learn the physical parameters. The key concept is that the entire framework is differentiable from end to end, and the authors further show that the model has nearly identical performance in gauged flow prediction to the record-holding entity-aware LSTM while exhibiting interpretable physical processes and adherence to physical laws like conservation of mass. A simpler implementation is seen in Khandelwal et al. [180], also for streamflow, where intermediate RNN models are used to predict important process model intermediate variables (e.g. snowpack, evapotranspiration) prior to the final output layer. In both of these implementations, we see a major advantage of the DPB model is the ability to output an entire suite of environmental variables in addition to the target streamflow variable, including baseflow, evapotranspiration, water storage, and soil moisture. The DPB approach has been further demonstrated on unmonitored prediction of hydrological flow in Feng et al. [186], showing better performance than the entity-aware LSTM for mean flow and high flow predictions but slightly

worse for low flow. The results of DPB models in both unmonitored and monitored scenarios challenge the notion that process-based model structure rigidness is undesirable as opposed to the highly flexible nature of neural network, and that maybe elements of both can be beneficial when the performance is near-identical in these specific case studies.

Table 2.1: Literature Table. Abbreviations as follows, DCBS: direct concatenation broad-scale, TL: transfer learning ANN: artificial neural network (feed forward multilayer perceptron), GNN: graph neural network, LSTM: long short-term memory neural network, MARS: multi-adaptive regression splines, MLR: multilinear regression, GBR: gradient boosting regression, GRU: gated recurrent unit, PDE: partial differential equation, RF: random forest, SVR: support vector regression, TCN: temporal convolution network, XGB: extreme gradient boosting

Work by	Variable Predicted	ML Framework Demonstrated	Type of Models Compared	Region Covered
Araza et al. [109]	Streamflow (daily)	DCBS, subgroup of entities broad-scale model	RF	21 watersheds in Luzon, Philippines
Arsenault et al. [52]	Streamflow (daily)	DCBS	LSTM, 3 different process-based models (HSAMI, HMETS, GR4J)	148 catchments in Northeast North America
Ayzel et al. [54]	Streamflow (daily)	DCBS	LSTM, process-based models (GR4J)	200 catchments in Northwest Russia
Bao et al. [86]	Streamflow (daily)	KGML (PDE-driven graph network)	ANN, RNN, recurrent graph network (2 types), PDE-driven graph network	42 river segments in Delaware River Basin

Table 2.1: Literature Table. Abbreviations as follows, DCBS: direct concatenation broad-scale, TL: transfer learning ANN: artificial neural network (feed forward multilayer perceptron), GNN: graph neural network, LSTM: long short-term memory neural network, MARS: multi-adaptive regression splines, MLR: multilinear regression, GBR: gradient boosting regression, GRU: gated recurrent unit, PDE: partial differential equation, RF: random forest, SVR: support vector regression, TCN: temporal convolution network, XGB: extreme gradient boosting

Work by	Variable Predicted	ML Framework Demonstrated	Type of Models Compared	Region Covered
Chen et al. [110]	Evapotranspiration (daily)	DCBS	LSTM, Temporal Convolution Network, ANN, RF, SVR, 7 different empirical models	16 weather stations in Northeast plain of China
Choi et al. [56]	Streamflow (daily)	DCBS	LSTM with different sets of inputs	13 catchments in South Korea
Corns et al. [111]	Stream water level (daily)	DCBS	LSTM ensembles	20 catchments in Missouri
Frame et al. [179]	Streamflow (daily)	DCBS, hybrid process-ML model	LSTM, NWM reanalysis, LSTM+NWM hybrid	531 catchments in US (CAMELS)
Feng et al. [67]	Streamflow (daily)	DCBS	LSTM with different sets of encoded inputs	671 catchments in US (CAMELS)
Ghosh et al. [133]	Streamflow (daily)	TL (meta transfer learning)	LSTM, sequence autoencoder	191 river segments in Delaware River Basin

Table 2.1: Literature Table. Abbreviations as follows, DCBS: direct concatenation broad-scale, TL: transfer learning ANN: artificial neural network (feed forward multilayer perceptron), GNN: graph neural network, LSTM: long short-term memory neural network, MARS: multi-adaptive regression splines, MLR: multilinear regression, GBR: gradient boosting regression, GRU: gated recurrent unit, PDE: partial differential equation, RF: random forest, SVR: support vector regression, TCN: temporal convolution network, XGB: extreme gradient boosting

Work by	Variable Predicted	ML Framework Demonstrated	Type of Models Compared	Region Covered
Jiang et al. [53]	Streamflow (daily)	DCBS	ANN, LSTM, KGML (custom network architecture)	450 basins in US (CAMELS)
Nogueira et al. 2022 [55]	Streamflow (monthly)	DCBS	LSTM, ANN, SMAP conceptual model	25 catchments in Brazil
Kalin et al. [187]	8 river water quality variables (daily)	DCBS	ANN with varying inputs	18 monitoring locations in west Georgia, USA
Koch et al. [165]	Streamflow (daily)	DCBS	LSTM, DK process model	301 basins in Denmark
Kratzert et al. [51]	Streamflow (daily)	DCBS	LSTM, SAC-SMA process model, NWM reanalysis	531 basins in USA (CAMELS)
Lee et al. [188]	Maximum Streamflow (annual)	DCBS, Hybrid process-ML model	ANN, RF, RNN, SVR	64 catchments in South Korea
Muhebwa et al. [189]	Streamflow (daily)	DCBS subgroup of entities broad-scale model	LSTM (not directly compared)	5 classes of catchments in Canada

Table 2.1: Literature Table. Abbreviations as follows, DCBS: direct concatenation broad-scale, TL: transfer learning ANN: artificial neural network (feed forward multilayer perceptron), GNN: graph neural network, LSTM: long short-term memory neural network, MARS: multi-adaptive regression splines, MLR: multilinear regression, GBR: gradient boosting regression, GRU: gated recurrent unit, PDE: partial differential equation, RF: random forest, SVR: support vector regression, TCN: temporal convolution network, XGB: extreme gradient boosting

Work by	Variable Predicted	ML Framework Demonstrated	Type of Models Compared	Region Covered
Noori et al. [166]	3 water quality nutrient loads (monthly)	DCBS, hybrid process-ML model	ANN, SWAT process model, Hybrid SWAT+ANN	29 monitoring locations in Georgia, USA
Ouyang et al. [190]	Streamflow (daily)	DCBS, subgroup of entities model	LSTM	3557 basins in USA
Potdar et al. [191]	Maximum streamflow (annual)	DCBS	XGB	3490 stream gauges in USA
Rahmani et al. [22]	Stream temperature (daily)	DCBS	LSTM	455 basins in USA
Rasheed et al. [64]	Flood peaks (¿90% quantile streamflow) (daily)	DCBS	LSTM, RF, gradient boosting	670 catchments in USA (CAMELS)
Razavi et al. [178]	Streamflow (daily)	Hybrid Process-ML, subgroup of entities broad-scale model	ANN, 2 process-based models (MAC-HBV and SAC-SMA)	90 watersheds in Ontario, Canada
Singh et al. [125]	Streamflow (daily)	TL	SVR, XGB, SWAT process model	6 catchments in India

Table 2.1: Literature Table. Abbreviations as follows, DCBS: direct concatenation broad-scale, TL: transfer learning ANN: artificial neural network (feed forward multilayer perceptron), GNN: graph neural network, LSTM: long short-term memory neural network, MARS: multi-adaptive regression splines, MLR: multilinear regression, GBR: gradient boosting regression, GRU: gated recurrent unit, PDE: partial differential equation, RF: random forest, SVR: support vector regression, TCN: temporal convolution network, XGB: extreme gradient boosting

Work by	Variable Predicted	ML Framework Demonstrated	Type of Models Compared	Region Covered
Sun et al. [81]	Streamflow (daily)	DCBS, broad-scale graph ML model	3 GNN architectures, LSTM	530 basins in USA (CAMELS)
Tayal et al. [66]	Lake temperature at depth (daily)	DCBS, broad-scale with encoding of site characteristics	LSTM with varied encoder networks	450 lakes in Midwest USA
Vaheddoost et al. [128]	Streamflow (daily)	TL, hydrid process-ML	RF, MARS, DAR process model	10 gauging stations on the Coruh River in Türkiye
Wang et al. [57]	Snow water equivalent (daily)	DCBS, TL	LSTM, SN17 process model	30,000 4km resolution pixels across USA
Weierbach et al. [14]	Stream temperature (monthly)	DCBS	XGB, MLR, SVR	93 monitoring stations in Mid-Atlantic and Pacific Northwest USA
White et al. [192]	Stream temperature (monthly)	DCBS	RF, MLR, BCM process model	69 basins in California, USA

Table 2.1: Literature Table. Abbreviations as follows, DCBS: direct concatenation broad-scale, TL: transfer learning ANN: artificial neural network (feed forward multilayer perceptron), GNN: graph neural network, LSTM: long short-term memory neural network, MARS: multi-adaptive regression splines, MLR: multilinear regression, GBR: gradient boosting regression, GRU: gated recurrent unit, PDE: partial differential equation, RF: random forest, SVR: support vector regression, TCN: temporal convolution network, XGB: extreme gradient boosting

Work by	Variable Predicted	ML Framework Demonstrated	Type of Models Compared	Region Covered
Willard et al. [1]	Lake temperature at depth (daily)	TL (meta TL), KGML (informed loss, simulation pre-train)	LSTM, GLM process model	450 lakes in Midwest USA
Willard et al. [4]	Lake surface temperature (daily)	DCBS	LSTM, ERA5 reanalysis, linear model	185,549 lakes in USA
Xiong et al. [138]	Riverine nitrogen export (daily)	DCBS, TL	LSTM	7 watersheds across the world
Yin et al. [193]	Streamflow (daily)	DCBS	LSTM with attribute-weighting module and multi-head-attention module	531 basins in USA (CAMELS)
Zhi et al. [59]	Riverine dissolved oxygen (daily)	DCBS	LSTM	236 watersheds in USA (CAMELS)
Zhou et al. [148]	Flood forecasting (6 hour scale)	DCBS, TL	Unsupervised Domain Adaptation with LSTM, TCN, and GRU	2 watersheds in China and Taiwan

2.3 Summary and Discussion

We see that many variations of the three classes of ML methodologies discussed in Section 2.2 are available for unmonitored prediction, a list of which is shown in Table 2.1. Clearly, entity-aware broad-scale modeling through direct concatenation of features remains the dominant approach, though it remains to be seen how these different methods stack up against each other when predicting different environmental variables since most of current studies are on streamflow prediction. More broadly, the evidence so far suggests that combining data from heterogeneous regions when available should be strongly considered. This recommendation is supported by the results from Frame et al. [179] discussed in Section 2.2.4 that suggest that using a broad-scale entity-aware ML model combining data from all regions is preferable to two different hybrid process-ML frameworks that harness a well-known process-based model in the NWM, and the results from Fang et al. [112] discussed in Section 2.2.2 that suggest deep learning models perform better when fed a diverse training dataset spanning multiple regions as opposed to homogeneous dataset on a single region even when the homogeneous data is more relevant to the testing dataset and the training datasets are the same size. This can likely be attributed to the known vulnerability property of ML models, where most ML models perform better when fed data from a diverse or slightly perturbed dataset (e.g. from adversarial perturbations) where they are able to learn the distinctions in underlying processes (see [194] for an example in hydrology).

We also see that generalization of ML models to unmonitored sites requires the availability of site characteristics. In streamflow prediction, these include soil porosity, catchment elevation, land use, slope, etc. These physical descriptors are universally used in both regionalized process-based or data-driven models, and ideally they account for process complexities and regional differences between sites. The entity-aware models listed in this study tend to exhibit performance increases when such characteristics are included. For example, [64] find site characteristics like soil porosity, forest fraction, and potential evapotranspiration all exhibit significant importance for flood peak prediction, and [58] find that the combined catchment characteristics make up 20 percent of the total feature importances for a continental-scale baseflow prediction model. Many methods in this survey use site characteristics in different ways, and an open question remains

of how to best add site characteristics to an ML model in a given task.

Throughout this review, we see several ways to incorporate site characteristics into ML model architecture and frameworks. The primary method we see to utilize site characteristics is in an entity-aware model using concatenated input features as seen in Section 2.2.1, presumably based on the landmark results from [44] and [51] in streamflow modeling. This is the simplest and easiest way as it requires no modification to model training or architecture. However, it has also been demonstrated that using a graph neural network approach using these site characteristics to determine similarity between sites can slightly outperform the concatenated input approach [81]. Site characteristics have also been used to build and predict with a metamodel the performance of different local models to be transferred to an unmonitored site [1, 133]. Other works mentioned in section 2.2.1 demonstrate the effectiveness of learning ML-based encodings of site characteristics as opposed to using them as-is [66, 133]. However, these approaches have not been tested against the concatenated input entity-aware approach commonly seen in other works.

It is also clear that the LSTM model remains by far the most prevalent neural network architecture for water resources time series prediction due to its natural ability to model sequences, its memory structure, and its ability to capture cumulative system status. We see that 26 of the 33 reviewed studies in Table 2.1 use LSTM. This aligns with existing knowledge and studies that have consistently found that LSTM are better suited for environmental time series prediction than traditional architectures without explicit cell memory [195, 196]. Even though we see the traditional ANN sometimes perform nearly as well or better [110, 55], the LSTM has the advantage of not having to tune the number of delayed input drivers which is a critical hyperparameter, due to its recurrent structure already incorporating many previous timesteps. We see other neural network architectures suitable for temporal data like transformers [197] and temporal convolution networks (TCN) [198] are not seen in unmonitored water resources applications nearly as much as other disciplines doing sequential modeling such as natural language processing and bioinformatic sequence analysis where these methods have largely replaced LSTM as the cutting edge high performing models. This is likely due to their recent development compared to LSTM and also possibly due to their lack of inclusion in major deep learning software packages like Pytorch and Keras. However, early adoption of these

methods in hydrology have been seen in prediction in monitored sites (e.g. TCN for runoff forecasting [199, 200], transformer for runoff prediction [201]).

We see that prediction at daily scale receives the most research focus, although a few studies choose to predict at a monthly, annual, or hourly scale based off factors like desired output resolution, data availability computational efficiency, or available computational power. For instance, monthly predictions may be desirable over daily due to the ability to use more interpretable, computationally efficient, and easy-to-implement classical ML models [14]. Increased computational efficiency can also enable running a large number (e.g. millions) of model trainings or evaluations for parameter sensitivity or uncertainty analysis.

Spatially, the majority of studies cover the United States at 23 out of 33 studies. 12 of these span the entire conterminous United States, while 11 are specific regions. The remaining studies are specific to certain countries and span Asia (8 studies), South America (1 study), Europe (1 study), other North America (2), and one study covers multiple continents. The strong focus on the United States can be due to its large land area with rivers alongside the economic capability to have advanced monitoring stations where data are freely available for study worldwide.

We also see the prevalence of the CAMELS dataset being used in streamflow studies; it is used in 8 out of the 33 studies in Table 2.1. CAMELS serves as transformative continental-scale benchmark dataset for data-driven catchment science with its combined high quality streamflow measurements spanning 671 catchments, climate forcing data, and catchment characteristics like land cover and topography. However, we note that it is limited to "unimpaired" catchments that are not influenced by human management via dams. In addition to dam managed catchments, catchments close to and within urban areas excluded from CAMELS are more likely to be impacted by roadways or other infrastructure. There are over 800,000 dammed reservoirs affecting rivers around the world, including over 90,000 in the United States [202, 203]. The effect of dammed reservoirs on downstream temperature is also complicated by variable human-managed depth releases and changing demands for water and energy that affect decision making [204]. These limitations may hamper the ability of current models to extrapolate to real-world scenarios where many catchments of high economic and societal value are either strongly human-impacted or data-sparse.

Chapter 3

Predicting Water Temperature Dynamics of Unmonitored Lakes with Meta Transfer Learning

3.1 Introduction

Environmental data often does not exist at the appropriate resolution or extent for decision making or characterizing change. Models can be used to fill gaps in key ecosystem variables, such as extreme precipitation rates [205], soil moisture [206], hydrological flow [207], and lake temperature [208], which otherwise would be unavailable at the spatial and temporal scales needed for ecological decision-making [209]. Although sensor data is increasingly prevalent, it will always be incomplete, especially for variables where observations are concentrated in a small subset of locations and the majority of locations remain unmonitored. Since observing key variables like these at scale is prohibitively costly [210], models that can efficiently use existing data and transfer information to unmonitored systems are critical to closing our information gaps.

There are many modeling approaches for predicting complex environmental phenomena, and model choice can be viewed as a trade-off among prediction accuracy, data needs, and generalizability to new systems. Process-based models are a popular modeling choice for water resources tasks like the prediction of stream temperature

[211], hydrological variables [212, 213], and lake temperature [214, 215, 216]. Process-based models encode our understanding of relevant physical processes into numerical formulations. These relationships are often developed from decades of theory, observation, and experimentation, resulting in sufficient understanding of processes and their interactions to support defining them with code for a simulation model [217, 218]. However, these models provide an approximation of reality and often require time-intensive parameter calibration to compensate for incomplete inclusion or resolution of processes. More recently, the rapid growth of sensor data [5, 219] along with advances in computation have led to development and increased use of powerful data-driven environmental models. Ensemble tree methods like gradient boosting and random forests, in addition to more advanced methods like deep learning [220], have been effectively used for geoscientific applications [221] and water resources [222, 223, 224]. A major reason for this success is that ML models, given sufficient data, can discern patterns and structure in problems where complexity prohibits explicit programming of a system's exact physical nature. Given this ability to automatically extract complex relationships from data, ML models (e.g., deep learning) appear promising for scientific problems with physical processes that are not fully understood by researchers, but for which data of adequate quality and quantity is available. Given enough data, data-driven models can increase prediction accuracy relative to existing process-based methods due to lack of *a priori* constraints and the expressive power of modern data-driven models, though they can lack interpretability and generalizability, and they often fail to leverage domain knowledge. Coupling deep learning in particular with process-based models is an emerging paradigm for modeling earth systems, enabling the discovery of patterns that are not only generalizable but also consistent with existing scientific knowledge [225, 223, 221]. For example, in [226, 227], typically data-hungry long short term memory deep learning models [228] are augmented with process-based knowledge to predict lake temperature dynamics more accurately than both the process-based model and the standard deep learning model. This class of method has been called "process-guided deep learning" (PGDL) and is an accelerating field of study [229, 230]. Previous works modeling lake temperature at a broad scale have focused on calibrating parameters with available data, when data are unavailable, using recommended default values based on field and laboratory studies [231, 216]. These approaches have since been outperformed by PGDL

models in cases of both high and low data availability [227]. However, in the case of no available temperature measurements to train or calibrate a model, no effort has yet been made to transfer PGDL models from well-monitored systems for prediction.

Lakes are an exemplar for the disparity in observations across systems, where >80% of in-situ water quality observations come from 20% of monitored lakes [232], and the majority of lakes have no in-situ monitoring data. In this chapter, we designate "monitored" vs "unmonitored" status of lakes based on the presence of in-situ data, and consider remote sensing integration in the discussion section. How can we leverage the information in a small population of lakes to make predictions in the much larger population of sparsely monitored to completely unmonitored systems? First, temporal synchrony in characteristics across ecosystems suggests that information or models from a highly monitored system could be transferred to a less- or un-monitored system. Examples include synchrony between stream temperature and streamflow, between organic matter concentrations across different lakes [233, 234], or coherence in lake temperature patterns [235]. Synchrony can emerge for a variety of reasons, including but not limited to shared underlying physical processes, weather conditions, or landscape context; patterns in synchrony across ecosystems therefore exhibit strong relationships to other physical variables. For instance, lake morphometric factors like maximum depth and surface area have a direct relationship to the stratification dynamics of lakes [236, 237] and correlate with temporal coherence between lakes [69, 68]. Water clarity can also affect the responses of below-surface phenomena to solar radiation across different systems [238, 239]. Differences in coherence strength can also be attributed to different dominant external drivers [240]. Fortunately, many of these physical characteristics like shape, depth, and water clarity are more widely available than other measures of water quality. Further, these characteristics mediate the relationship between external drivers and within-lake responses (e.g., through sedimentation rates and head storage), such that information gained about dynamics in one lake could be transferred to other similar lakes, regardless of whether they exhibit temporal synchrony. Determining the generalizability of the relationship between physical characteristics and water quality dynamics across different ecosystems could allow the strategic transfer of site-specific models from well-monitored systems to predict temporal patterns in unmonitored systems.

Currently, methods to extend accurate site-specific models to broad scale predictions

are rare or nonexistent. In hydrology, extending site-specific parameterizations has been achieved through regionalization and catchment classification [241, 242, 243]. For example, [243] focus on transfer functions connecting geophysical attributes to process model parameters. However, these approaches are not widely regarded as successful, with noted drawbacks of (1) uncertainty in geophysical attributes, which translates to large uncertainty in parameter estimates, and (2) often-weak relationships between these attributes and parameters, perhaps because many of those parameters lack direct physical meaning [244]. Water resources research has yet to establish a robust way to bridge scales for prediction accuracy for key ecosystem variables.

Transfer learning is a powerful technique for applying knowledge learned from one problem domain to another, typically to compensate for missing or nonexistent data in the new problem domain. The idea is to transfer knowledge from an auxiliary task, i.e., the source task, where sufficient labeled data is available, to a new but related task, i.e., the target task, often when data is scarce or inadequate [245, 114]. Transfer learning using deep neural networks has shown recent success in ecological applications such as plant classification models [246], air quality prediction [247], and grassland fire risk assessment [248]. Transfer learning for deep neural networks is analogous to calibrating process-based models in well-monitored systems and transferring the calibrated parameters to models for unmonitored systems, which has shown success in hydrological applications [115, 116]. The task of deciding what model or parameters to transfer can be posed as a problem to be solved by meta-learning, or learning from previous learning experiences, which is another active area of machine learning research [249, 250]. In this chapter, we focus on the meta-learning task of systematically learning how to map candidate source models (models trained on well-monitored lakes) to a particular task (prediction in unmonitored lakes) [251]. For clarity, we define *base-learning* models as the traditional machine learning models or process-based models that learn or are calibrated for specific tasks (e.g., prediction in a specific lake) as opposed to the *meta-learning* model's goal of learning from a multitude of experiences transferring source models to target tasks. In the transfer learning context, which we call *Meta Transfer Learning*, the meta-learning predicts which base models to transfer based on performance metrics for past transfer learning experiences and meta-features relating to the

transferability of base-learning models [252]. We demonstrate this method by transferring a suite of source lake temperature models to a number of artificially unmonitored target lakes, where temperature observations were only used for final evaluation. The metamodel was used to determine which source models would transfer well to the target lake and which lake attributes can best indicate the transfer performance.

Here, we demonstrate a meta transfer learning framework to predict lake temperature at depth. Our objectives are to (1) Demonstrate the use of a metamodel to rank both process-based models and process-guided deep learning models from well-monitored lakes (source lakes) in terms of their expected ability to predict lake temperature for a different, unmonitored lake (target lake); (2) Evaluate the MTL approach against existing process-based modeling approaches; and (3) Investigate the extent to which MTL can outperform the existing state-of-the-art process-guided deep learning models for the target lake itself in situations of limited observation data.

3.2 Materials and Methods

3.2.1 Overview

Here, we describe a method for model selection of trained source models from data-rich systems to predict lake water temperature in target systems with no data. The general idea of the MTL framework is visualized in Figure 3.1 and summarized as follows,

1. Build and train two source models, a calibrated PB model and PGDL model, for each of the 145 well-monitored lakes.

2. For each source lake, use all 144 other source models of the same type (PB or PGDL) to predict daily temperatures and evaluate prediction accuracy.

3. Train the meta-learning model to predict the 145*144 collected model performance values from (2) based on the lake characteristics that we hypothesized could be important for selecting good transfer models.

4. Given an artificially unmonitored target lake, where data is only used for final evaluation, and its meta-features, use the meta-learning model to predict model

performance of each source model. Use the source model[s] with the lowest predicted error to model the target.

Sections 3.2.2 and 3.2.3 summarize the two types of source models, PB and PGDL models, respectively. Then, we describe the meta-learning model and how it is trained and used to identify lake features that predict successful transfers between source and target lakes 3.2.4. Lastly, Section 3.2.5 describes the data used in Section 3.2.6, which contains descriptions of the experiments.

3.2.2 Process-Based Models

As in previous studies of deep learning applied to lake temperature prediction [227, 226], we chose the General Lake Model (GLM) 2.2 [215] to represent process-based modeling, due to its proven ability to simulate thermal hydrodynamics in lakes along with its open-source code availability (`https://github.com/AquaticEcoDynamics/GLM`). GLM can also be used to predict temperature at broad scales using widely-available lake characteristics (depth, surface area, clarity) to parameterize the model even when observations are not available [216]. We acknowledge that GLM may not be the ideal process-based model in all calibrated and uncalibrated modeling scenarios, but consider the comparison of different process-based models for broad scale prediction to be out-of-scope of this chapter.

Given that the MTL framework can use any similar hydrodynamic process-based model, we will further refer to the calibrated GLM using all the available observation data as "PB" and the parameterized but uncalibrated version of GLM as "PB0". To calibrate GLM for the PB models, we selected three parameters for calibration based on their known importance to model fits and their relative uncertainty: solar radiation scaling factor, momentum exchange coefficient, and hypolimnetic mixing efficiency. We used the `optim()` function in R [253] to modify these parameter values to minimize the RMSE of GLM temperature predictions relative to the available observations. See the Supplemental Information (text S3) in [227] for details.

3.2.3 Process-Guided Deep Learning (PGDL) Models

We used a recently-developed PGDL model for lake temperature prediction, [226, 227], where process knowledge was combined with a Long Short Term Memory (LSTM) network via (1) a loss function term to encourage physical consistency and (2) pre-training using process-based model simulation data. LSTM networks are part of a class of deep learning architectures built for sequential and time series modeling called recurrent neural networks [228]. These are particularly suited for predicting lake temperature dynamics given the often persistence of the response and the time lag between the input drivers and water temperature changes that can be represented in the memory properties of LSTM [226, 227]. Here, the simulation data used for pre-training are the output of the parameterized but uncalibrated version of the PB model (PB0) described in Section 3.2.2. The components of the PGDL model are described in more detail in Supplemental Information (Text S1). The input features for the model are the meteorological factors that contribute to incoming and outgoing heat fluxes and the depth (distance from surface) of the target prediction [254, 255, 256]. This includes short-wave and long-wave radiation (in W/m^2), air temperature (in), relative humidity (0-100%), wind speed (in m/s), rain (in m/day), and snow (in m/day). The meteorological features are identical to the drivers used in the GLM simulations except that they are each normalized to a mean of 0 and standard deviation of 1 based on a calculated global mean for each driver across all lakes, a recommended step for training neural networks to address differences in the scales across input variables [257].

3.2.4 Meta Transfer Learning with Gradient Boosting Regression

Our MTL framework aims to predict the accuracy of each source model on an unmonitored target lake. Here, two metamodels were built, one for predicting the performance of source PB models on target lakes (PB-MTL), and one for predicting the performance of source PGDL models on the same target lakes (PGDL-MTL). As shown in Figure 3.1, each meta-learning model takes in lake-level features that may contain information about the transferability from a source to a target. We call these predictors *meta-features*; meta-features included differences in physical attributes between the source and target lake, measures of data quality in the source lake, and features of the source

and target that were derived from PB0 such as the likelihood of stratification. The response variable was the prediction accuracy (measured as root mean squared error, RMSEs) of transferring the source model (either PGDL or PB) to the target lake, where lower RMSEs represent a successful transfer between lakes. If i is the index for the source lake, and j the index of the target lake, the meta-features for each unique source-target pair can be written as $X_{\mathrm{i}\to\mathrm{j}}$, and target RMSE values as $RMSE_{\mathrm{i}\to\mathrm{j}}$. The function $\mathcal{F}$ we are attempting to approximate can then be written as

$$\mathcal{F}(X_{\mathrm{i}\to\mathrm{j}}) = RMSE_{\mathrm{i}\to\mathrm{j}} \tag{3.1}$$

The training dataset for each metamodeling scenario then contains all $(n) * (n-1)$ possible source-target pairs as follows:

$$\{(X_{i\to j}, RMSE_{i\to j}) | i \neq j\} \tag{3.2}$$

The following subsections describe details of this MTL approach, including the method of gradient boosting regression used for the metamodel, how meta-features were selected, and how gradient boosting hyperparameters were tuned.

Gradient Boosting Regression

Due to its predictive power, ease of implementation, and ability to illustrate the relationships between predictors and the response, we chose gradient boosting regression to predict the RMSE of source-target pairs from meta-features. In short, gradient boosting creates an ensemble of estimator models. It starts by fitting an initial regression tree model to the data. Regression decision trees are generated such that each decision node in the tree contains a test on the input variable's value, and the tree terminates with nodes that contain the predicted output variable values (RMSE in this case). Then, it builds a second model that prioritizes accurately predicting the cases where the first model performs poorly, a process known as boosting. The ensemble of these two models can be expected to perform better than the first model due to this new prioritization. Estimators are then continuously added until a set amount is reached. Gradient boosting in particular generalizes boosting by optimizing with a differentiable loss function, which in the case of regression is usually mean squared error (MSE). Further method

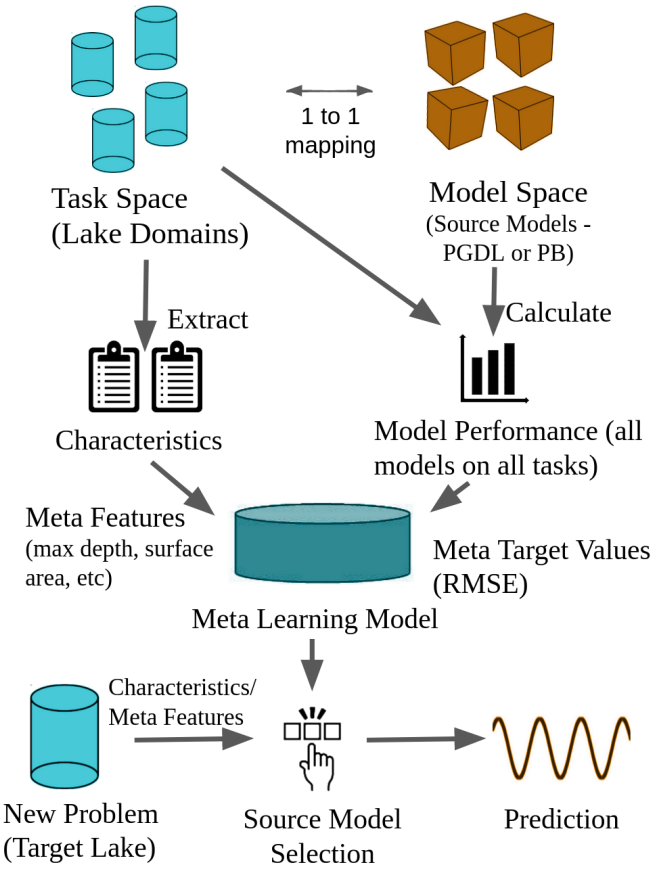


Figure 3.1: Meta-learning general framework. The meta learning model (metamodel) is trained to predict source model performance (root mean square error; RMSE) based on lake domain characteristics (meta features). The performances and characteristics from all source models applied to all other source lakes are used for metamodel training. This trained model is used to predict source model performance and inform source model selection for a *new* target lake.

details can be found in [258].

Identification of Meta-Features

We started with a collection of candidate meta-features that we hypothesized could predict the performance of source models in predicting temperature in different target lakes. As in Equations 3.1 and 3.2, each set of meta-features ($X_{i \to j}$) is unique to a source-target lake pair. An exhaustive list of 96 possible meta-features is listed in Supplemental Information Table S1, which are divided into four categories: lake attributes,

PB0 simulation statistics, general observation statistics, and meteorological statistics. The last two categories are commonly-used meta-features that either (1) use statistics relating to the quality and quantity of observations of the source lake, or (2) compare differences in the data distributions of input features between source and target domains [259]. We expand on this with the two additional categories, lake attributes and a PB0 simulation statistic. All differences are calculated as the source value minus the target value.

1. **Lake Attributes**: These features contain information about maximum depth, surface area, and other lake properties that are not directly used in model training since they are not features or observations, but may mediate or contain useful information about the physical response of lakes to meteorological drivers. They are calculated as the difference between the source and the target lake values.

2. **PB0 Simulation Statistic**: This feature describes an important property of the PB0 temperature predictions, the percentage of dates on which each lake is stratifies. We used PB0 predictions as a surrogate for in-situ temperature observations, which are not available for target lakes. The PB0 model translates driver data into temperature predictions via process understanding, and it can therefore give insight into similarities across lakes such as the likelihood the lake is stratified or how the lake responds to wind events. This statistic was already available as part of the pre-training process for PGDL models in this chapter study, and is also calculated as a difference between the source and the target lake.

3. **General Observation Statistics**: These features contain information about temperature measurements that only pertains to the source lake. Ideally they would contain information about the quality of the source data. For example, a very poorly monitored source lake without adequate data to train a model could indicate poor transfer performance. Example statistics include total observations, number of observations per season, mean depth where temperature was measured, and mean temperature measured.

4. **Meteorological Statistics**: These features contain differences between the source and the target lake in both annual and seasonal averages and standard deviations

of the 7 meteorological drivers used as inputs to the source models. Examples include differences in mean air temperature, solar radiation, and relative humidity.

Then, to narrow down the number of meta-features, we performed recursive feature elimination with cross validation (RFECV) [260]. Recursive feature elimination is a feature selection method that fits a model and iteratively removes the weakest features until an ideal set that produces the lowest cross-validation error is reached. To do this we used two Scikit-learn python modules [261]. For building the base model we used `GradientBoostingRegressor` with default parameters and 3000 estimators, and for performing feature selection we used the `RFECV` library with 24-fold cross validation and mean squared error loss. We also used the importance of each meta-feature to interpret how the transfers were selected. Here, feature importance was calculated by the `GradientBoostingRegressor` as a measure of how each feature affected mean squared error across nodes in the decision trees, weighted by how often those nodes are reached [261].

Hyperparameter Tuning

For both PB-MTL and PGDL-MTL, we tuned two gradient boosting hyperparameters that are known to affect performance: the number of boosted decision trees and the learning rate (impact of each tree on final outcome). The remaining parameters were left at their default values for the *GradientBoostingRegressor* class in scikit-learn version 0.22.1. We construct a nested 24-fold cross validation (CV) to estimate the generalization ability of the model given certain hyperparameter values. This CV works by performing 24 iterations of removing 1/24th of samples from the dataset for validation and taking the average mean squared error as an estimate of model performance for a given set of hyperparameters. CV is done for every set of candidate hyperparameter values in an exhaustive search of two candidate learning rates {0.05, 0.10} and intervals of 100 decision tree estimators from 1000 to 6000. The ideal hyperparameters were found to be learning rates equal to 0.05 for both PB-MTL and PGDL-MTL, and number of decision trees equal to 4500 for PB-MTL and 4900 for PGDL-MTL.

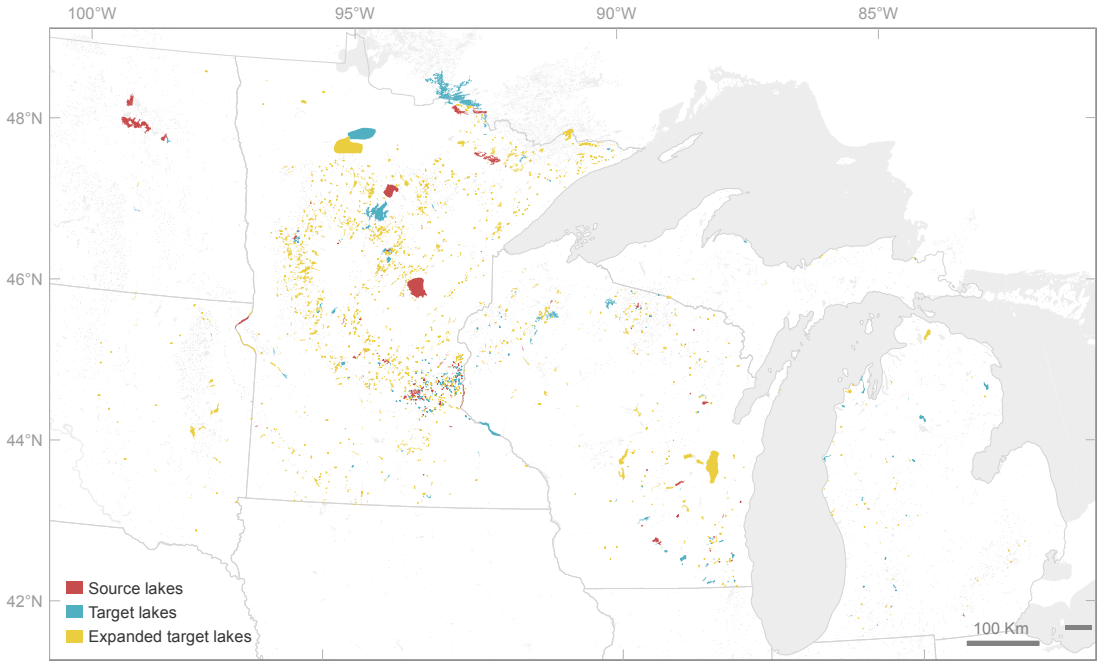


Figure 3.2: Map of all lakes used in experiments. 145 source lakes are shown in red, 305 initial target lakes are shown in blue, and the additional 1882 expanded target lakes are shown in yelllow.

3.2.5 Data

All the data used in this work is available through a data release on the U.S. Geological Survey's ScienceBase platform [262]. All study lakes are located in the Midwestern United States, and details about the selected lakes are included in the data release. Briefly, 450 lakes met our data density criterion of at least 50 unique observation dates where there are at least one measurement for every two meters of depth or at least 5 total observations. From these lakes, in-situ lake temperature measurements between 1980 and 2019 were used to train and test all our models. To build the metamodel 145 of these lakes were used, and the rest are considered "artificially unmonitored", where data is only used for final evaluation. An additional 1882 lakes with fewer observations were used as targets in an expansion exercise described in the Discussion (Figure 3.2). Meteorological data used as the input drivers for our models were gathered from the North American Lake Data Assimilation System (NLDAS-2) [263]. As in [216, 227], these gridded data were transformed into process-model ready input (see [215]). These

inputs were then normalized for use in the machine learning models as mentioned in Section 3.2.3. Lake attributes used as meta-features in the MTL algorithm were acquired in the same manner as in previous modeling studies in the region [231, 227]. A more detailed description of the sources and processing of these attribute data can be found in [216].

3.2.6 Model Experiments

We designed two experiments that use the previously described metamodel built using meta-features and past model transfer RMSE. For both experiments, we used 145 of the 450 well-monitored lakes as detailed in Section 3.2.5 as source lakes, and we kept the remaining 305 lakes as target lakes for which the metamodel was used to select one of the 145 source lakes. Source lakes were selected to be representatively distributed across maximum depth values and log-scale surface area values (see Supplemental Information Figure S3). In all experiments the metamodel training data consisted of RMSEs from applying each of the 145 source lake models on all other source lakes, leading to 144*145 meta-learning data points (20880 total). Then, after the metamodel is trained, for each source-target pair we constructed the meta-features as described in Section 3.2.4. From these meta-features, both metamodels (PB-MTL, PGDL-MTL) were then used to predict the expected RMSE of each of the 145 source models when transferred to the target lake.

Experiment 1: Predicting Temperatures in "Unmonitored" Lakes

Experiment 1 evaluates the performance of the meta transfer learning models in a real-world scenario: predicting water temperature at multiple depths in unmonitored lakes. Given the predictions of source model performances from both metamodels, the PB or PGDL model with the lowest predicted source-to-target RMSE was singled out for use on each target lake. We compared the top-predicted transfers for each of the 305 test lakes against its PB0 simulation. We assumed that our metamodel would not be able to select the true best source lake in all instances. We therefore also evaluated an ensemble method that combined several of the top predicted models. Ensembles of multiple individual models that perform well can almost always improve over their average prediction error[264, 265]. This was proven by [266], who showed that increasing

ensemble diversity (the extent to which single models disagree), given constant average error of individual ensemble members, reduces the overall ensemble error. This reduction often occurs because some ensemble member predictions are biased positively and some negatively, leading to bias cancellation in the ensemble prediction. We used a simple ensemble that takes an unweighted average of the predictions from each selected PGDL source model for each date and depth. Lakes selected for ensembling were the top "n" source models predicted to have the lowest RMSE on the test lake. The optimal value of n was estimated using a 29-fold cross validation. In each cross validation fold, 5/145 source lakes were designated as validation lakes and a metamodel with the same hyperparameters as described in Section 3.2.4 was trained on the remaining 140 lakes. Then, n source lakes were selected for each validation lake. Estimated ensemble error was then the mean ensemble errors across all folds. This was repeated for values of n ranging from 2 to 10, where 9 was found to be the optimum, but values differed minimally between 5 and 10. We call this 9 source ensemble approach PGDL-MTL9. Lastly, we examined the metamodels themselves. In addition to evaluating the performance of the predicted best source-target transfer, we looked at how well the metamodel predicted the RMSE of *every* source-to-target combination and how well it was able to rank the source models. For the former we calculate a median across all target lakes of the metamodels' predictions for RMSEs and the actual source-to-target RMSEs. For the ranking evaluation, we used the Spearman rank correlation coefficient shown as r_s. We also looked at distributions of the actual ranks (for the RMSEs of sources actually applied to targets) of those models that were identified by the metamodel as the top or top 9.

Experiment 2: Comparing PGDL-MTL with PGDL for Sparsely Monitored Systems

Experiment 2 examines the extent to which PGDL-MTL is an improvement over PGDL in systems that have some observations but are not sufficiently monitored to train any traditional deep learning model effectively. In this experiment, we define "sparsely monitored" as between 1 and 50 sampling dates. Deep learning models are generally data-hungry, but PGDL models pre-trained on PB0 output have shown to achieve high accuracy with only a few observations [226, 227]. Thus, both PGDL and PGDL-MTL

Table 3.1: *Results of PB-MTL and PGDL-MTL Applied to Test Lakes.*

Method	Median RMSE (° C)	Lower quartile RMSE	Upper quartile RMSE	Median meta RMSE	Median r_s
PB0	2.52	2.07	3.12	–	–
PB-MTL	2.42	2.04	2.95	0.853	0.653
PGDL-MTL	2.16	1.74	2.81	0.871	0.663

Note. The first three columns are the quartile distributions of RMSE of the best predicted source lake for each test lake. The fourth column is the median RMSE between the metamodel-predicted RMSEs and the observed RMSEs. The fifth column is the median Spearman rank correlation coefficient between the metamodel-predicted RMSEs and the actual RMSEs.

have the potential to alleviate the difficulty in calibrating process-based models for sparsely monitored lakes, where overfitting can be a problem. However, PGDL-MTL has the potential to harness more data and thereby outperform PGDL. The situation of few available observations is also far more common than the well-monitored case of the lakes chosen in this work [267]. To that end, we artificially sparsified the data available in the 305 test lakes to train PGDL models on low amounts of data. Then, these low-data PGDL models were compared to the PG-MTL and PGDL-MTL results of Experiment 1. We used this comparison to estimate the data threshold where PGDL tends to outperform PGDL-MTL. Artificial sparsity was induced by building five PGDL models for each suitable lake for twelve different amounts of sampling dates (1, 2, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50) used for training. The test period was set as the first third of temperature observations in time for each lake, leaving the training period as the last two thirds of temperature observations. For each sampling date treatment we used all lakes that had at least that number of sampling dates during the training period. Of the 305 possible lakes, 221 had 50 or more observations, 270 had 40 or more observations, and all 305 had 30 or more observations. For the five models within each lake and data availability category, variability was introduced by randomly selecting dates for the training data.

3.3 Results

PB- and PGDL-MTL model accuracy on 305 test lakes

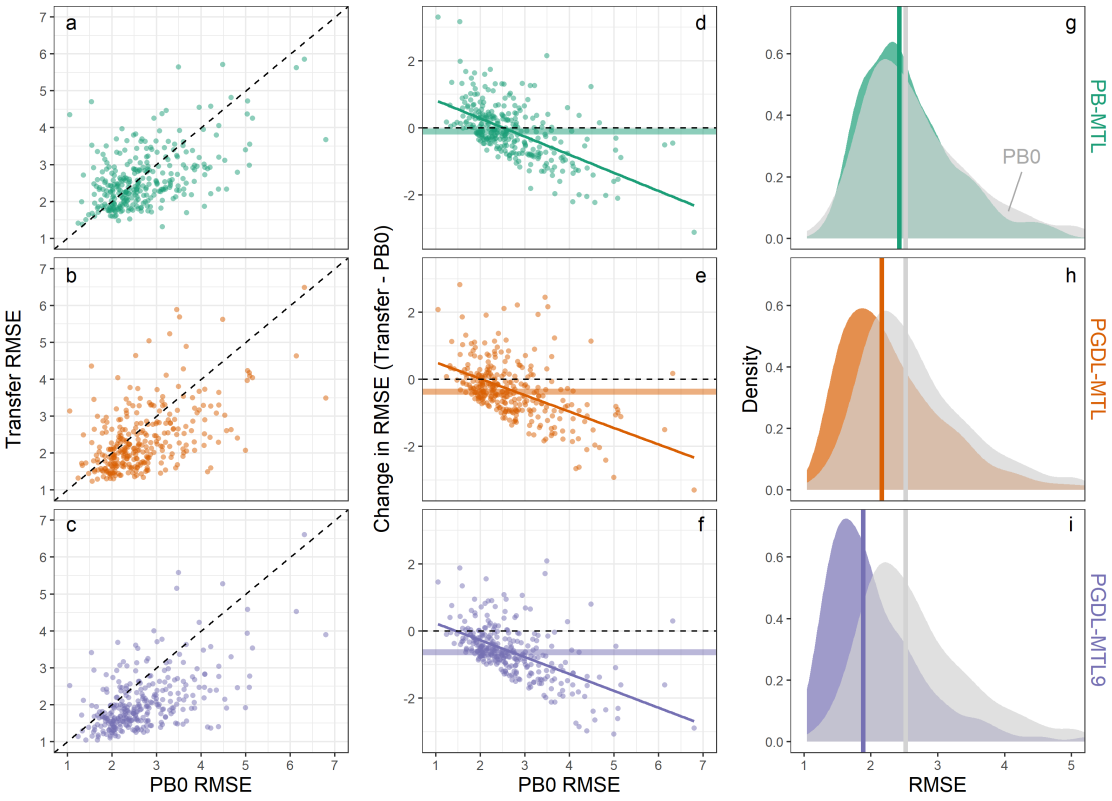


Figure 3.3: Comparison of the performance of the three MTL approaches relative to PB0 on 305 lakes. a-c) RMSE of PB0 relative to the three transfer models, where the dotted line shows the 1:1 relationship. d-f) The difference between RMSE of the transfer and PB0 models, where the black dotted line shows the zero or no change line, and the solid colored lines show the linear regression fit of the change in RMSE as a function of PB0 RMSE. g-i) The distribution of RMSE from PB0 and transfer models, where the vertical gray and colored lines are the median PB0 and transfer RMSE, respectively. PB-MTL (a,d,g) and PGDL-MTL (b,e,h) are the transfer of process-based and PGDL models respectively, and PGDL-MTL9 (c,f,i) is an averaged ensemble prediction of the top 9 PGDL models.

Table 3.2: *Median Actual RMSE of PGDL Source Models of Different Metamodel-Predicted Ranks.*

Source system(s)	Median RMSE (°C)	Lower quartile RMSE	Upper quartile RMSE
Rank 1 Source	2.16	1.73	2.80
Rank 2 Source	2.21	1.79	2.77
Rank 3 Source	2.15	1.75	2.82
Rank 4 Source	2.20	1.85	2.86
Rank 5 Source	2.20	1.78	2.83
Rank 6 Source	2.25	1.85	2.86
Rank 7 Source	2.23	1.84	2.86
Rank 8 Source	2.24	1.84	2.90
Rank 9 Source	2.21	1.83	2.90
9 Source Ensemble	1.88	1.56	2.41

In Experiment 1, PGDL-MTL and PB-MTL predictions of water temperature in the 305 test lakes were typically more accurate than predictions from the uncalibrated process-based model (PB0; Table 3.1 and Figure 3.3). The median RMSE across the test lakes was 2.42°C for PB-MTL and 2.16°C for PGDL-MTL, versus 2.52°C for PB0. PB-MTL outperformed PB0 for 203/305 of the lakes and PGDL-MTL outperformed PB0 for 226/305 of the lakes, and the amount of improvement the transfer provided generally increased with PB0 error (Figure 3.3). Predictions of deeper water temperatures from the transferred models had higher RMSEs in general as compared to the lake-specific accuracy of all depths, with the highest median RMSE from PB0 models, followed by PB-MTL, and with PGDL-MTL having the lowest median deep-water RMSE (2.59°C, 2.56°C, and 2.36°C, respectively; RMSEs calculated based on predicted versus observed temperatures at or below 75% of the maximum depth of the lake; 18 of 305 lakes had no observations at these depths).

Model ensemble performance

Additionally, the ensemble PGDL-MTL9 model provided still better performance than PGDL-MTL. We can see in Table 3.2 that the RMSE of the combined averaged prediction of the source models tended to be lower than most of the source models individually. The ensemble model PGDL-MTL9 had a median RMSE of 1.88 °C, which is an

Table 3.3: *Selected Features for PB-MTL and PGDL-MTL and Importances*

Meta-feature	**MTL importance**	
	PB	PGDL
Max Depth Difference	0.39	0.26
Max Depth Percent Difference	0.11	0.19
GLM Stratification Percent Difference	0.18	0.066
Surface Area Difference	0.065	0.087
Surface Area Percent Difference	0.037	0.087
Mean Source Observation Temperature	0.037	0.072
Number of Source Temperature Observations	0.028	0.072
Square Root Surface Area Percent Difference	—	0.085
Lathrop Stratification Difference	0.020	0.034
Autumn Relative Humidity Difference	—	0.048
Source Observation Temp and Target Air Temp Difference	0.033	—
Mean Autumn Wind Speed Difference	0.027	—
GLM Stratification Absolute Difference	0.024	—
Kurtosis Source Observation Temperature	0.023	—
Mean Autumn Shortwave Difference	0.020	—
Skew Source Observation Temperature	0.017	—

improvement over the single-source PGDL-MTL, which had a 2.16 °C median RMSE. When compared to PB0 in Figure 3.3, PGDL-MTL9 outperforms PB0 for 260/305 of the test lakes. Table 3.2 also shows the distribution of RMSE values per source systems at given ranks, between 1 and 9, as predicted by the metamodel. Comparing the individual source model RMSEs across the top 9 ranks, we see ranges of only 0.09 °C in median RMSE, 0.12 °C in lower quartile RMSE, and 0.13 in upper quartile RMSE.

Meta-features and importances

The top selected meta-features were related to maximum depth in both PB-MTL and PGDL-MTL, with combined importances of 50% and 45%, respectively. Surface area, observation count, source lake observed temperature, and stratification indicators were selected as meta-features in both PB-MTL and PGDL-MTL but were of lesser importance (Table 3.3).

Example time series prediction

The metamodels typically chose source models that were good, but not optimal, matches to the target lake (Figure 3.4). In a stratified lake with high PGDL-MTL

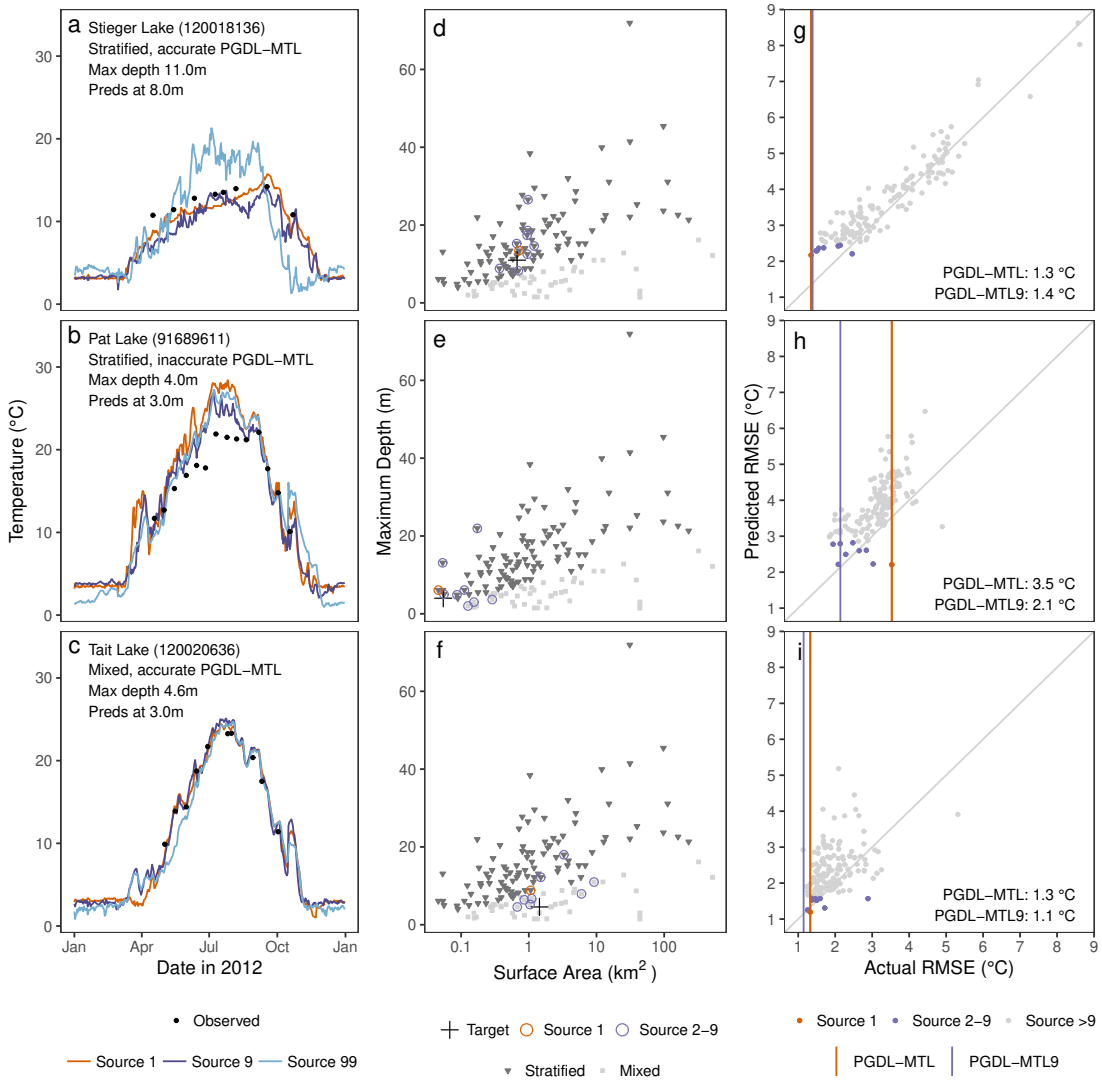


Figure 3.4: Deep-water predictions for three example lakes to illustrate the application of PGDL-MTL and PGDL-MTL9. Lakes were selected to represent successful and unsuccessful PGDL-MTL results for stratified lakes (rows 1 and 2, respectively) and the easier case of a mixed lake (row 3). Steiger, Pat, and Tait Lakes have 2,573, 469, and 3,865 total temperature observations, respectively. Panels a-c: Time series predictions at two depths in 2012 for each target lake from the top-ranked PGDL source (Source 1), 9th-ranked source (Source 9), and a lower-ranked source (Source 99), with observed values (points) for comparison. Panels d-f: metamodel selections of source lakes for each lake, arranged by three features that dominated the MTL predictions: maximum depth (y axis), surface area (x axis), and predicted stratification (darker = stratified). Panels g-i: Metamodel-predicted RMSEs versus actual RMSEs (for all depths and years) for the three example lakes.

accuracy (RMSE = 1.3 °C), top-ranked source models all came from stratified source lakes (Figure 3.4d) and captured the summer stratification dynamics (Figure 3.4a). In a stratified lake with low PGDL-MTL accuracy (RMSE = 3.5 °C), top-ranked source models came from a mix of stratified and unstratified source lakes (Figure 3.4e) and had similar predictions to a low-ranked source model (Figure 3.4b). In our 305-lake test set, mixed lakes (n=121) had lower mean RMSEs (mean=2.01, SD=0.52 °C) than stratified lakes (n=184; mean=2.62, SD=0.96 °C). Our mixed example lake illustrates that all candidate source lakes had lower RMSEs (Figure 3.4i) and similar predictions (Figure 3.4c) such that even though the metamodel selected a combination of mixed and stratified source lakes, the resulting RMSEs could still be quite low (PGDL-MTL: 1.3 °C, PGDL-MTL9: 1.1 °C). Consistent with the meta-feature importances in Table 3.3, the selected source lakes tended to be similar to the target lake with respect to not just stratification but also maximum lake depth and surface area (Figure 3.4d-f). Ensembling with PGDL-MTL9 yielded similar accuracy to PGDL-MTL for the two example lakes with high PGDL-MTL accuracy (Figure 3.4g,i) and substantially improved accuracy in the example lake where the PGDL-MTL model failed to capture the observed stratification dynamics (Figure 3.4h).

Features of best and worst source lakes

There were large differences in the frequency at which source models were chosen by the MTL to represent target lakes, and several factors emerged that suggested differences exist between commonly and rarely selected source lakes. A small fraction of source models were used to predict almost one third of target lake water temperatures, and eleven lakes were selected as the top PGDL or PB source for ten or more target lakes. Seven top PGDL source models were used for 100 target lakes and seven PB models for 95 of 305 target lakes, and three lakes were in this top category for both PGDL and PB models. In contrast, 59 PGDL and 64 PB source models were not chosen as a top model for any target lake (31 were never selected as sources in either model). Additionally, we summed the number of times each lake was predicted to be in the top 9 sources for the ensembles, and compared the raw lake attributes of the upper and lower quartiles (Figure 3.5). For both PGDL and PB transfer models, lakes that were transferred often were in general deeper, larger, and more monitored than minimally transferred lakes.

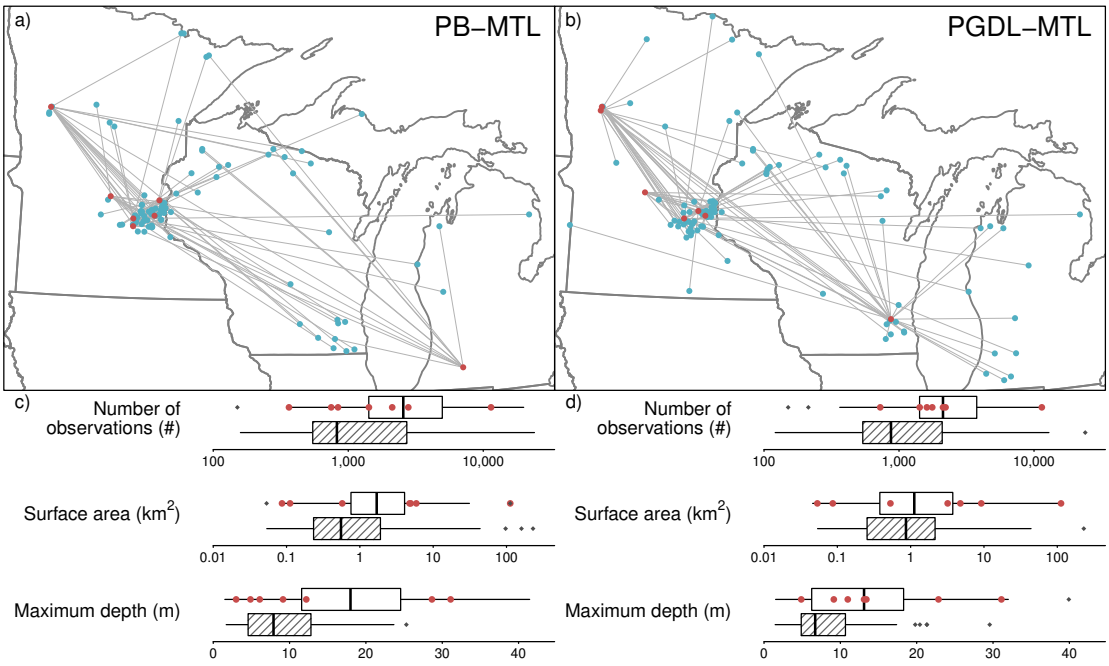


Figure 3.5: Top-selected source models compared to lesser-selected sources. In a), the seven process-based (PB) models chosen as a top source for ten or more target lakes by the meta transfer learning (MTL) model are shown in red, with grey lines connected to the paired target lake location; b) is the same as a) but for process-guided deep learning source models. In c), properties of lakes in the upper quartile of commonly chosen PB source models (white fill boxplot) are compared to the lowest quartile (hashed fill boxplot; based on MTL rank). Red dots represent the location of the seven source lakes featured in a). d) is the same as c), but for process-guided deep learning source models.

For PGDL, source models in the lower quartile of MTL selections had a median depth of 6.71 m, a median surface area of 0.86 km2, and a median of 872 training observations, compared to 13.1 m, 1.12 km2, and 2,117 observations for the upper quartile medians, respectively. The lower quartile of PB-MTL source models had medians of 7.9 m, 0.55 km2, and 824 calibration observations, with upper quartile medians of 18 m, 1.7 km2, and 2,557 calibration observations.

Metamodel performance

To assess the metamodel's ability to predict the performance of source lake models, we looked at both the RMSE of the predicted RMSE versus the actual RMSE when transferring source models in Experiment 1, and also the ability of the metamodel to

accurately rank source models from best to worst in the form of the Spearman rank correlation coefficient. The median meta-RMSE for PB-MTL was 0.853°C and the Spearman rank correlation coefficient r_s was 0.659, and for PGDL-MTL the meta-RMSE was 0.871°C with an r_s of 0.663 (Table 3.1). Then, in Figure 3.6, in addition to showing the distribution of actual ranks for the predicted best source PGDL model for each target system, we also show the distribution of ranks for sources within the 9 source ensemble PGDL-MTL9. Further visualization of the ranking ability of the metamodels is shown in Supplemental Information Figure S2. Here, we see that the two metamodels have similar predictive ability, with PGDL-MTL ranking slightly better as seen in the Spearman coefficient values.

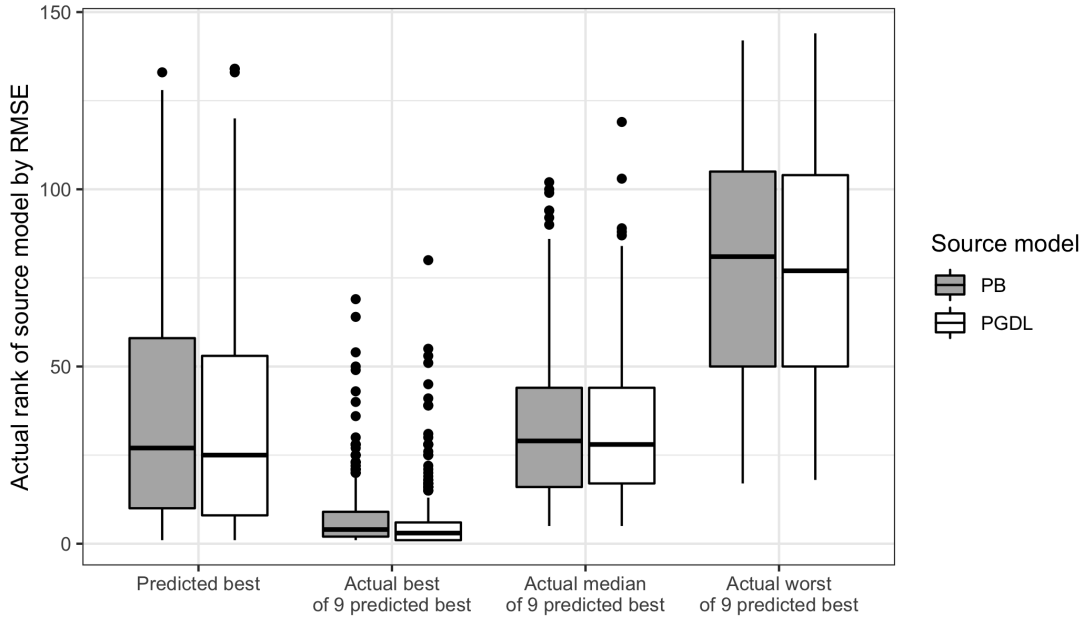


Figure 3.6: Plot showing the distribution of actual ranks of the metamodel-predicted top source models, for metamodels built on either PB sources (gray fill) or PGDL sources (white fill). Leftmost pair of bars: actual ranks for top-predicted models for each of the 305 target lakes. Other bars: best, median, and worst of the top-9-predicted sources.

Comparison with data-sparse target lake models

In Experiment 2, the median RMSE across 305 test lakes tended to decrease as the number of sampling dates used for training increased (Figure 3.7 and Table 3.4). Figure 3.7 shows performance of PGDL trained with differing numbers of temperature profiles compared to the MTL approach, and Table 3.4 shows the specific RMSE numbers for Figure 3.7. Here, the RMSE for each test lake is defined as the median RMSE across 5 randomly chosen sets of the same number of observations. Given that the RMSEs of the single-source PGDL-MTL and ensemble-of-sources PGDL-MTL9 from the previous experiment were 2.16 °C and 1.88 °C, respectively, PGDL models trained only on the target lake's data met or exceeded median MTL performance at between 5 and 15 observations for PGDL-MTL and between 35 and 40 observations for PGDL-MTL9. In other words, even for a reasonably well-monitored lake (up to 40 observations), it can be better to borrow a model from a different and better-monitored lake than to train

a model on the target lake observations. For context, 45 profiles is approximately the coverage a lake would have if it had a monitoring program that took a temperature profile monthly during the ice-free period for 6 years.

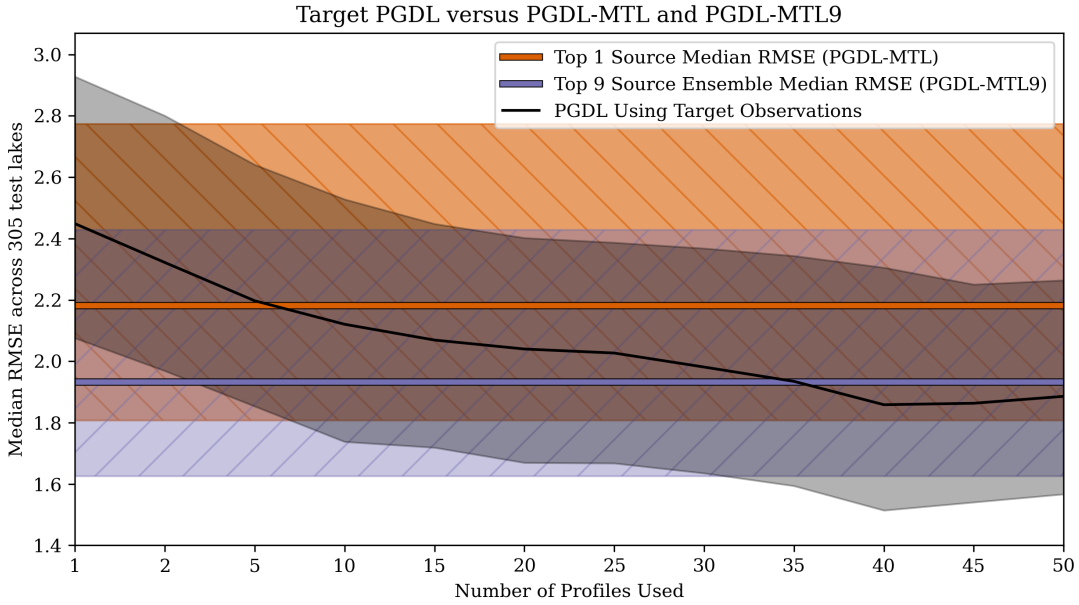


Figure 3.7: Median RMSE for PGDL trained with differing numbers of temperature profiles, with error bars representing upper and lower quartiles of the median RMSE across the 5 randomized selections of observations for each target lake. Colored horizontal lines represent the median RMSE with a band showing the range from lower to upper quartile for PGDL-MTL and PGDL-MTL9

Baseline performance of PB and PGDL source models Success in transfer learning depends both on (1) metamodel success in choosing the best of the available source models for a target lake and (2) the baseline performance of the source models that could be transferred. If the PGDL-MTL metamodel had selected the best available source PGDL model for every target lake, the median RMSE would have been 1.54 °C, versus an RMSE of 1.79 °C if the best PB model was selected every time. This difference, aligning with established knowledge that PGDL predicts more accurately than PB [227, 226], can explain how the RMSE across the test lakes of PGDL-MTL was lower than PB-MTL even though PB-MTL had a lower meta-RMSE predicting the performance of source models.

Table 3.4: *Data for Figure 3.7, Performance of PGDL Trained on Various Amounts of Target Lake Temperature Data Profiles*

Target Profiles	Median of Median RMSE (°C)	Lower quartile of Median RMSE (°C)	Upper quartile of Median RMSE (°C)
1	2.45	2.08	2.93
2	2.32	1.97	2.80
5	2.20	1.85	2.64
10	2.12	1.74	2.53
15	2.07	1.72	2.45
20	2.04	1.67	2.40
25	2.03	1.67	2.39
30	1.98	1.64	2.37
35	1.93	1.59	2.34
40	1.86	1.51	2.31
45	1.86	1.54	2.25
50	1.89	1.57	2.26

Note. Medians of medians are calculated as the median across 305 test lakes of the median of 5 models trained with different random selections of observations.

3.4 Discussion

In this chapter, we show Meta Transfer Learning (MTL) can be used to address monitoring gaps in environmental and ecological sciences by predicting in unmonitored systems. Even with the data deluge resulting from modern sensor developments, the majority of lakes and streams are unmonitored or have sparse observations. This has made it difficult to calibrate process-based models for these systems due to risk of overfitting, and even more inaccessible for traditional deep learning models which can require thousands or millions of data points. The MTL paradigm in this chapter harnesses data from many other systems to accurately predict temperature in unmonitored systems. Specifically, the transfer process leverages observations from highly monitored systems, simulated temperature data from process models, past model performance measures, and thousands of past transfer learning experiences to alleviate the drawbacks of both deep learning and process-model calibration in unmonitored systems. As experts in the water resources community have called for integration of process-based and data-driven methods [268, 223], MTL involves a collection of approaches harnessing both ML and process knowledge. Here, we use the ML technique of gradient boosting regression for the meta-learning task of predicting the transferability of source models including those that employ Process-Guided Deep Learning (PGDL), which iteself integrates process knowledge into ML. Limnology domain expertise was also used in defining the candidate meta-features offered to the metamodel. The top selected meta-features matched our process understanding from dozens of studies that show relationships between the properties of lakes (surface area, depth) and physical responses to external drivers [236, 237]. This chapter shows that these lake characteristics, which are more widely available than water quality data themselves, can be used to transfer information from highly monitored to unmonitored systems. Different types of lake-specific data were used to determine which sources should be transferred. Lake maximum depth difference between the source and target lake emerged as the most important in both the PGDL-MTL and PB-MTL approaches. Surface area differences were also included in both, but of less importance. This aligns with existing process-based lake modeling knowledge that maximum depth and surface area are key factors in lake stratification and thermodynamics [236, 237]. Other meta-features related to source model quality,

like the number of observations and mean observation temperatures, were also included in both metamodels. This is consistent with common modeling intuition that more data can lead to both better calibrated process models and better trained ML models [227, 226]. We also saw the PB0 meta-feature, GLM stratification percentage, as the 2nd most important feature for PB-MTL, and included with less importance in PGDL-MTL. Top sources had lower mean observation temperatures, which possibly indicates either more balanced measurements between surface and deeper depths, or a better spread of observations across seasons, in a given lake. For example, source lakes that use mostly surface temperatures would have higher mean observation temp, and source lakes that have many deeper measurements would have lower mean observation temp. Inspecting the characteristics of the most frequently selected source lakes could guide future monitoring and MTL modeling efforts. Only eleven unique source models were used to predict almost one third of target lake water temperatures using both PGDL-MTL and PB-MTL, and top source lakes were generally deeper, larger, and more well-monitored compared to rarely or never selected source models (Figure 3.5). Differences between source and target in lake depth and area, as well as the observation count of source lakes, were important meta-features used to select source lakes. While these features likely explain why some lake models are generally more transferable, the unique properties of some target lakes and their selected source models are important to consider when designing lake monitoring campaigns or evaluating future model transfer methods. For example, PGDL source models that were rarely selected (chosen one, two, or three times as a top source) still helped overall test lake performance and were often better actually-ranked options for their target lakes compared to the ranks of commonly chosen (ten or more times) source models (based on ranking the performance of all possible source model transfers to each target lake; median actual PGDL-MTL rank for rare source transfers: 20 of 145, and common source transfers: 36.5 of 145; n=100 and 103, respectively). This pattern did not hold for PB-MTL transfers, with generally worse actual ranks for rare sources compared to common sources (median rank for rare and common were 28.5 and 23, and n=80 and 95, respectively), and additional research may be necessary to understand these differences. The important meta-features used in this chapter's study (e.g., differences in maximum depth and area, and the number of observations used to train or calibrate the source model) differ from previous

process-based modeling parameter transfer methods that have been applied to rivers. These previous works have instead focused on spatial proximity, spatial fields of hydrologic signatures, or global parameterization [269]. Because lake temperature is an ecological "master factor" [270], predictions at broader scales can support a wide variety of science and management efforts, from improved modeling of biota [271, 272] to improved thermoelectric power plant heat management [273]. PGDL-MTL models can output predictions at scale wherever meteorological and essential lake attribute data are available, and the MTL approach could eventually be developed into use for forecasting applications. A forecasting variant of MTL could be developed by building base models specifically for forecasting (e.g. with probabilistic outputs), and optimizing transfer performance to new systems by simulating forecasting performance instead of hindcasting RMSE. Below, we discuss the various ways the MTL approach can scale to other systems. The applicability of MTL scales beyond just unmonitored systems to a large range of monitored systems as well, bridging the gap between local accuracy and broad-scale modeling. In Experiment 2, we investigated the point at which, for sparsely monitored systems, it would be better to transfer models from different better-monitored systems as opposed to training PGDL models on what little target data is available. This is a pertinent question for broad scale modeling; while a majority of lakes in this region are unmonitored, a large fraction of monitored lakes have <40 observations [232]. Though PGDL models have been shown to outperform calibrated process-based models on even a small number of water temperature sampling dates by taking advantage of process-based simulation data and process-informed learning constraints [227], MTL presents the opportunity to improve prediction by harnessing more simulation data, observation data, and metadata from past modeling experiences across many other lake systems. There is also opportunity to expand the MTL framework to incorporate sparse data available in many lakes, where the transferred source models could be fine-tuned using data from the target lake itself. Another major benefit of MTL with PGDL in particular is the scalability and efficiency of ML models once the meta-learning model and source models are trained. MTL can be built with data that are easier to obtain than temperature observations (e.g. maximum depth and surface area), and MTL does not require any new models to be trained. Therefore, it can scale to a much larger number of lakes than the ones used in this chapter's study. To demonstrate this scalability, we

Table 3.5: *Method Comparison Across Broad-Scale Modeling of 1882 Lakes in the Midwestern United States*

Method	Median RMSE (C)	Lower Quartile RMSE	Upper Quartile RMSE
PB0	2.28	1.84	2.94
PGDL-MTL	2.06	1.59	2.74
PGDL-MTL9	1.80	1.40	2.38

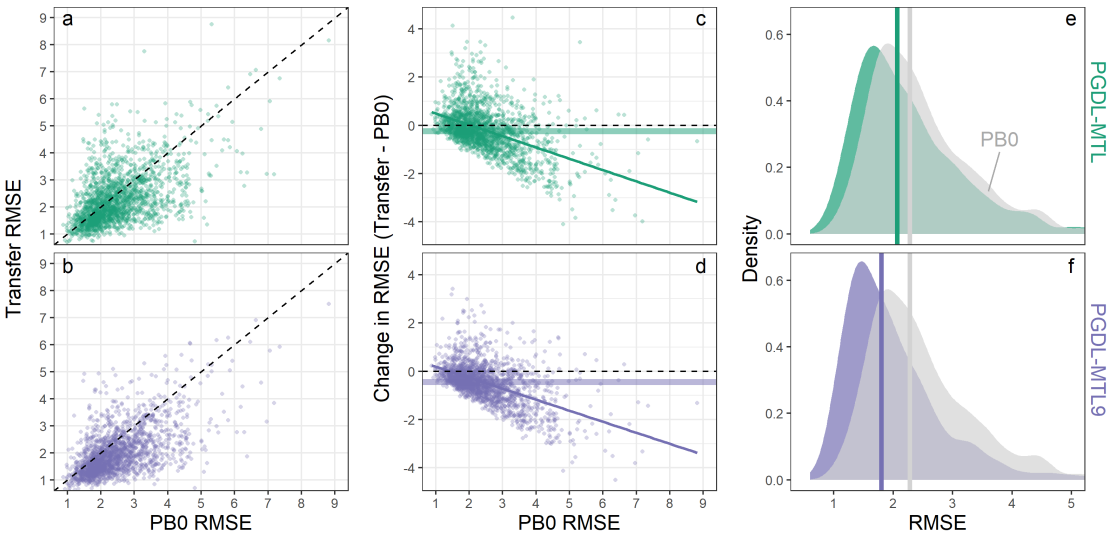


Figure 3.8: Comparison of model performance of PGDL-MTL (a,c,e) and PGDL-MTL9 (b,d,f) RMSE relative to PB0 across 1882 test lakes. a-b) RMSE of PB0 relative to the two transfer models, where the dotted line shows the 1:1 relationship. c-d) The difference between RMSE of the transfer and PB0 models, where the black dotted line shows the zero or no change line, and the solid colored lines show the linear regression fit of the change in RMSE as a function of PB0 RMSE. e-f) The distribution of RMSE from PB0 and transfer models, where the vertical gray and colored lines are the median PB0 and transfer RMSE, respectively.

applied the transfer models to 1882 additional lakes that were less monitored than our initial 305 lakes. The transfers maintained a significant accuracy improvement over a purely process-based modeling approach (PB0). For this expanded set of lakes, median RMSE was 1.80 °C for PGDL-MTL9, 2.06 °C for PGDL-MTL, and 2.29 °C for PB0 (Table 3.5). Temperatures in a majority of lakes were more accurately predicted by the transfer models compared to PB0; for PGDL-MTL9 1484 of 1882 lakes improved over PB0, and for PGDL-MTL 1206 of 1882 improved over PB0 (Figure 3.8). Finally, given the demonstrated generalizability of PGDL and PB models using MTL, this approach opens doors to new research directions, like transferring source models into new spatial domains, including remote sensing surface observation data, incorporating uncertainty quantification, and aggregating models more effectively. Though this application was limited to 5 Midwestern states in the United States, this could be expanded to include a much larger variety of lake types and locations. A remaining question for this transfer approach is, when expanding to new types of lakes, how should an optimal set of source lakes be identified? Another research direction includes uncertainty estimation in the metamodel construction. Uncertainty estimates could be used to reject a target lake for which all the source model error estimates are confidently high. Furthermore, the ensembling approach could be improved, using more complex methods than a simple average to combine top source models. One promising option is generalized stacking of neural networks [274], where all the source neural networks would be connected by an averaging layer. Remote sensing data integration could also help in adding surface temperature data to the source models and could allow corrective measures to be taken for predictions in lakes unmonitored by in-situ data. Though, remote sensing observations have known drawbacks in this application such as being limited to only surface temperature on larger lakes [275, 40]. Given the successful prediction of environmental variables using MTL approaches, there are many research opportunities in different types of applications and data scenarios. For example, predicting only lake surface temperature would allow for the use of MTL without the need for maximum depth measurements, which could allow for predictions in many more lakes. Also, different types of source models could also be used in different scenarios. Some process-based models likely work better for some lakes than others; for example, process models built specifically for reservoir dynamics could be important source models in regions where reservoirs are a

common lake type. Other environmental variables could also be targeted for prediction like water quality (e.g. dissolved oxygen, conductivity) and water quantity in lakes, streams, wetlands and other water bodies.

Chapter 4

Entity-aware LSTM estimates of daily surface temperatures for unmonitored lakes

4.1 Introduction

Measured or estimated water temperatures are necessary to understand basic aquatic functions (such as microbial decomposition rates and gas exchange; [276] and to assess habitat suitability for numerous species (e.g. Fang et al. [277]). Diversity in lake temperatures results from unique combinations of weather, climate, and lake-specific properties that modulate responses to meteorological inputs [240, 239]. Observing lake water temperatures at a temporal resolution sufficient to resolve short-term dynamics (such as temperature drops resulting from cold fronts) and of temporal duration sufficient to measure long-term trends is challenging and often prohibitively expensive, especially when attempting to capture diverse thermal regimes across many lakes. Despite these challenges, water temperature is the most common variable in the U.S.'s Water Quality Portal [24], and numerous satellite data products include a measure of surface water temperature [275, 278]. This seemingly high abundance of temperature measurements has been aided by the low cost and simplicity of thermistor sensors for

in situ measurements as well as advances in atmospheric correction and emissivity algorithms in remote sensing. However, of the over 270,000 U.S. lakes in the National Hydrography Dataset PlusV2, fewer than 5% have in situ temperature observations and only 62% are resolvable by satellite [275]. These numbers are significantly lower when accounting for the millions of smaller waterbodies in the U.S. not included in NHDPlusV2, ultimately meaning that temperature of the vast majority of U.S. lakes is unobserved on most days.

New environmental modelling methods that are equipped to leverage existing data are improving prediction accuracy and being used to create useful data products. Machine learning algorithms are increasingly viable prediction methods for water resources applications due to surging availability of observational data and computational power [279]. In particular, deep learning algorithms composed of large, multilayer artificial neural networks can extract hierarchical features from raw data and have increased accuracy without the need for feature construction by experts [16, 280]. The entity-aware long short term memory (EA-LSTM) network is one deep learning architecture specifically developed for environmental time series prediction using a mix of static and dynamic input drivers [281]. These modeling and data advances provide powerful tools to support the need to create broad-coverage foundational datasets (such as water temperature). We have used the EA-LSTM approach to reconstruct the daily historical surface temperature record for 185,549 lakes in the conterminous United States from 1980-2020. Here, we describe the dataset, methods used to create it, provide an overview of the evaluation of the predictions, and compare this data resource to other existing methods or datasets.

4.1.1 Associated Dataset Description

This dataset, summarized in Figure 4.1, includes predicted daily surface water temperatures for 185,549 lakes and reservoirs (hereafter referred to simply as lakes) in the conterminous United States (the lower 48 states and the District of Columbia) from 1980-2020. Lake surface temperatures were predicted using an advanced deep learning model that is described in the methods section; this model was compared to two published models for predicting lake surface temperatures: the ERA5 climate reanalysis

aggregation of the process-based FLake model [282, 283] and the empirical linear regression model developed by Bachmann et al. [3]. The dataset also includes data used to develop and evaluate the deep learning model, including observed water temperatures, historical downscaled weather conditions, lake-specific properties, and model evaluation metrics. Temperature observations, weather data, and lake properties were compiled from publicly available data portals and existing data publications. All data are referenced to the national hydrography dataset [284] high-resolution waterbodies using the NHD's PermID field (this dataset prefixes the value of this field with "nhdhr_"), with the exception of gridded weather data. Weather data are referenced by the longitude and latitude index of the source dataset grid cells because more than one lake can be contained within a single grid cell.

Lake surface temperature predictions are accessible from three NetCDF [285] files covering sections of the conterminous United States as broken up by longitude and latitude boxes. Each file contains data for all lakes with surface area larger than 4 ha within each file's spatial boundary. These data include dimensions for time and NHD lake identifier and variables for surface temperature in degrees Celsius, the elevation of the lake, and the latitude and longitude of the lake centroid. Meteorological data used to drive daily temperature models are included in three additional NetCDF files that share the same spatial extents of the temperature prediction files. Meteorological data include downward longwave radiation flux, downward shortwave radiation flux, air temperature 2m above the surface, and zonal and meridional wind speeds at 10m above the surface. All lake surface temperature observations are included in a single comma-delimited file, with a column for lake identifier, time, observed water temperature in degrees Celsius, and estimated temperatures from each of the three temperature models. All lake-specific static values that were used to quantify lake properties were inputs to the predictive model, describe model error, or are used to connect to the appropriate NetCDF file names or indices, and are included in a single metadata file. Model accuracy was calculated using a cross validation technique (see Methods for additional details), and the root-mean square error (RMSE; $^\circ$ C) of predicted versus observed temperatures for lakes in each validation fold is included in the metadata file. Additionally, as mentioned above, the match-ups for daily predicted and observed temperatures for each fold are available in a data file and can be used for analyzing additional dimensions of model

performance not presented in this paper (e.g., estimated accuracy of predictions in a certain time of year for a particular subset of lakes).

All data files are available for download directly from `https://doi.org/10.5066/P9CEMS0M` using the web interface, or programmatically with the sbtools R package [286]. Example workflows for extracting surface temperatures for a single lake or all lakes for a single date are shared in the data release code repository (see "readme.md").

4.2 Methods

Our objective was to produce the most accurate and comprehensive predictions of daily surface water temperatures for lakes in the conterminous U.S. and to expose all underlying data that were used to build, drive, and evaluate these predictions to enable future expansion and comparison. Based on prior information from existing datasets and modelling efforts [3, 24, 287, 288], we excluded predictors that may be useful in temperature models but were be available broadly due to data limitations (e.g., lake depth and water clarity). We also treat our predictions as daily mean water temperatures even though observed values may be at specific times throughout the day for simplicity. Here we describe the methods used to select models and assemble the various data included in this dataset. The code to reproduce these results is available at (`https://doi.org/10.5281/zenodo.6210917`).

4.2.1 Model Descriptions

We compared three different approaches to broad-scale lake surface temperature modelling, the entity-aware long short-term memory neural network (EA-LSTM) [281], the process-based Fresh-water Lake model (FLake) [289, 282, 290] used in the European Centre global reanalysis ERA5 at 0.1° latitude and longitude grid resolution [2] at 17:00 UTC (Coordinated Universal Time; approximately noon local time for much of the U.S. domain), and the linear regression model (LM) for summer temperature prediction described in Bachmann et al. [3]. This choice of methods represents state-of-the-art deep learning in the EA-LSTM, the only process-based simulation model with comprehensive global coverage via FLake and ERA5, and a simpler data-driven model in the linear regression. EA-LSTM is an adaptation of the standard deep learning LSTM architecture

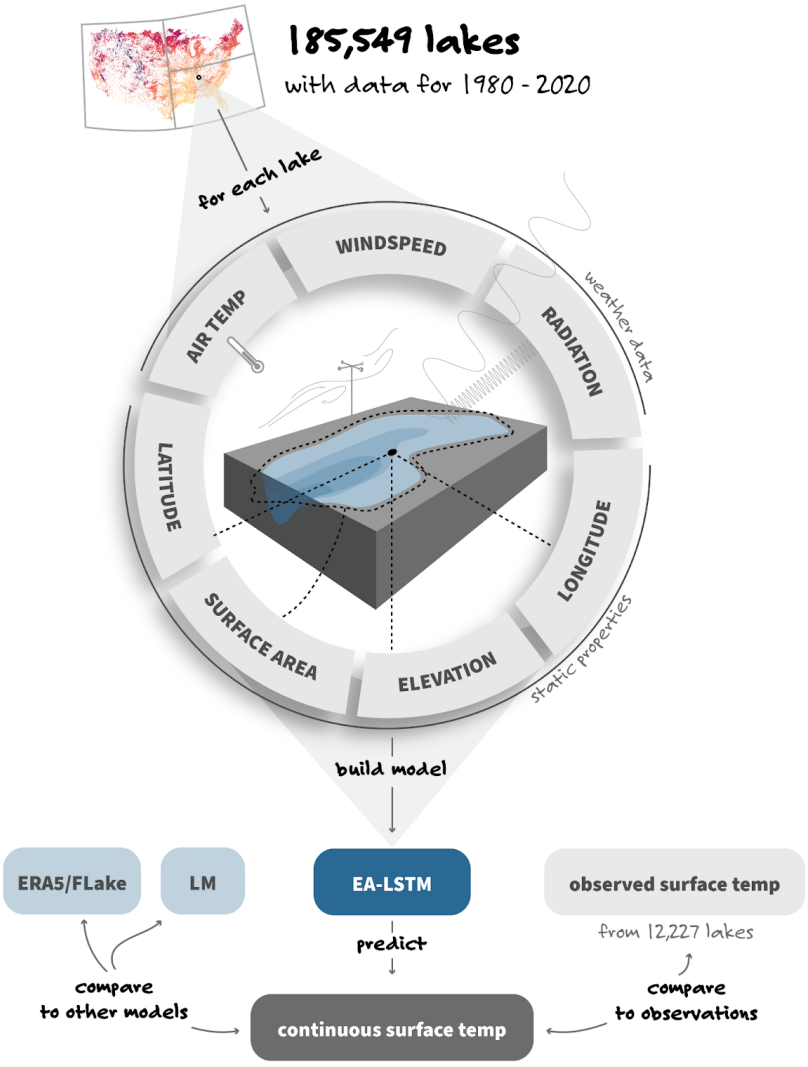


Figure 4.1: Overview of the data and modelling flow used to create the daily surface temperature predictions. The entity-aware long short-term memory neural network (EA-LSTM), a deep learning approach designed for time series and other sequential data, is built using the seven input drivers shown below as well as observed surface temperatures. EA-LSTM outputs are compared against the ERA5 reanalysis-simulated epilimnetic lake temperature outputs [2] and a linear model (LM) described in Bachmann et al. [3]. Each data component shown is available as part of the associated data release [4]; https://doi.org/10.5066/P9CEMS0M). Inset map displays summer predictions for a single date and the spatial division used to break up the largest files (prediction and weather data) into three NetCDF files.

[228] for time series modelling that includes additional architectural distinction between static (e.g., lake surface area) and dynamic (e.g., air temperature) input features. Many temporal processes in environmental and engineering systems that involve complex temporal dependencies cannot be captured by a simple feed-forward artificial neural network (ANN). LSTM models have been shown to outperform ANN models for lake temperature prediction in Jia et al. [291], and [162] showed ANNs to have superior performance compared to support vector regression and boosted regression trees. However, providing time-awareness to simpler machine learning models via additional inputs (such as lagged meteorological conditions and day-of-year time vectors) can substantially boost performance [292], but these inputs must be selected *a priori* or learned from independent data to avoid overfitting to the training data. The EA-LSTM in particular has previously been applied in continental-scale rainfall–runoff modelling where it substantially outperformed all calibrated process-based hydrological models, and also showed learned similarities between different catchments that matched prior expert hydrological understanding [281]. ERA5 makes use of the one-dimensional FLake model, a two-layer parametric representation of the dynamic water temperature profile and the integral energy budgets of these layers (for further FLake details see Mironov et al. [282]). The FLake model is forced at the surface by reanalysis-derived data of wind, temperature, precipitation, humidity, and shortwave and longwave radiation. ERA5 cells for some near-coastal lakes did not have temperature estimates, and were not included in model evaluation.

4.2.2 Input: Meteorological conditions and lake-specific characteristics

Both EA-LSTM and LM models predicted surface water temperature from lake-specific characteristics that were static over time (log-transformed surface area, latitude, longitude, and elevation) and daily meteorological drivers that changed over time for each lake (air temperature, longwave radiation, shortwave radiation, and components of wind speed). EA-LSTM used all of these inputs, while LM only used air temperature (8 day lag-averaged), latitude, longitude, elevation, and month of the year. The choice of these dynamic features come from the well-established understanding of connections between

meteorological conditions and water temperature change [293, 294, 295]. Latitude, longitude, and elevation features allow the model to learn spatial coherence in temperature, and surface area has a known role mediating lake responses to meteorological drivers [296]. Because neural networks benefit from input normalization [297], an additional z-score normalized version of the inputs was created for the EA-LSTM based on the mean and standard deviation for each input calculated across the 12,227 observed lakes in the dataset.

The national hydrography dataset (NHD; [284]) high-resolution polygons (based on 1:24,000 scale data) were downloaded as geodatabase files for each of 48 states in the conterminous United States, as well as the District of Columbia. Lakes and reservoirs were extracted using the "NHDWaterbody" layer from the geodatabase and filtered to values in the "FType" attribute that corresponded to 390, 436, and 361 (lake/pond, reservoir, and playa respectively). The Great Lakes, several improperly labeled coastal lagoons, and lakes less than 4 ha (based on the value in the "AreaSqKm" NHD attribute) were removed from the dataset, and the remaining 185,549 lakes defined the complete lake coverage used in this data release.

Hourly meteorological data for the five variables described above were downloaded from the North American Land Data Assimilation System (NLDAS; [298]); we used a NASA earthdata login to access NetCDF files through https://hydro1.gesdisc.eosdis.nasa.gov/dods/NLDA and daily datasets were created by applying a U.S. central time zone offset for the entire spatial range and calculating the daily mean for each variable. The 0.125° NLDAS latitude and longitude grid was then used to assign NLDAS grid IDs to each lake's centroid using the "st_centroid" and "st_intersects" functions from the "sf" R package [299]. All grid cells that did not contain a lake were excluded and the remaining daily dataset was transformed from a spatial grid (latitude, longitude, and time) into a flatter and smaller discrete sampling geometry NetCDF format [300] indexed to grid ID and time.

Approximate lake surface area and elevations were calculated based on the vector polygon data ("st_area"; [299]) and lake centroid, respectively. Lake surface elevation was estimated for each lake using the "get_aws_points" function from the "elevatr" package [301] at the zoom level of nine, providing a centroid-based value in meters for each lake from an elevation raster from the Shuttle Radar Topography Mission data [302].

4.2.3 In-situ lake temperature data

Lake temperature data were compiled from two main sources: digitized or spreadsheet-based historical records shared directly with researchers [303] and through programmatic access to discrete monitoring data in the joint Environmental Protection Agency and U.S. Geological Survey water quality portal (WQP; [24]). High-frequency buoy data and remote sensing data were not used in this dataset due to extreme differences in temporal coverage that would favor a small number of lakes (as in the case of buoy data) and the large drop in measurement accuracy in satellite-based estimates of surface water temperatures when compared to in-situ observations (e.g., mean absolute error ranging from 1.34°C to 4.89°C depending on distance to lake shore with the Landsat analysis ready surface temperature product; [304]). Prior compiled data from Read et al. [303] included temperatures from lakes in U.S. midwestern states and were combined with updated national pulls of water temperature data from the WQP from 1980 to 2020. Unique WQP lake monitoring sites with temperature data were captured by breaking the spatial extent of the conterminous United States into 2.5° by 2.5° latitude/longitude cells and calling "whatWQPdata" function from the "dataRetrieval" R package [305] for "Lake, Reservoir, Impoundment" siteTypes and "Temperature," "Temperature, sample," "Temperature, water," and "Temperature, water, deg F" characteristicNames on each cell's bounding box. Monitoring sites were then ranked according to expected number of observations (the "resultCount" value from the "whatWQPdata" result) and broken up into site groups containing no more than 500,000 total results or less than 200 unique sites, and each site group was queried for all available temperature data using the same characteristicNames as listed above. Resulting data were converted into standard depth as measured in meters and temperature as measured in degrees Celsius and then all observations deeper than 1m were removed and basic quality control measures were applied (see Technical Validation Methods section). Monitoring site locations, which are defined by a single spatial location, were joined to lakes by using point-in-polygon analysis and sites falling outside of the 185,549 lakes in this data release were excluded. The above process resulted in 306,553 in situ temperature observations from 12,227 lakes for model development. Geographic coverage density of the observed lakes is shown in Figure 4.2a, and the temporal coverage is shown in Figure 4.2b.

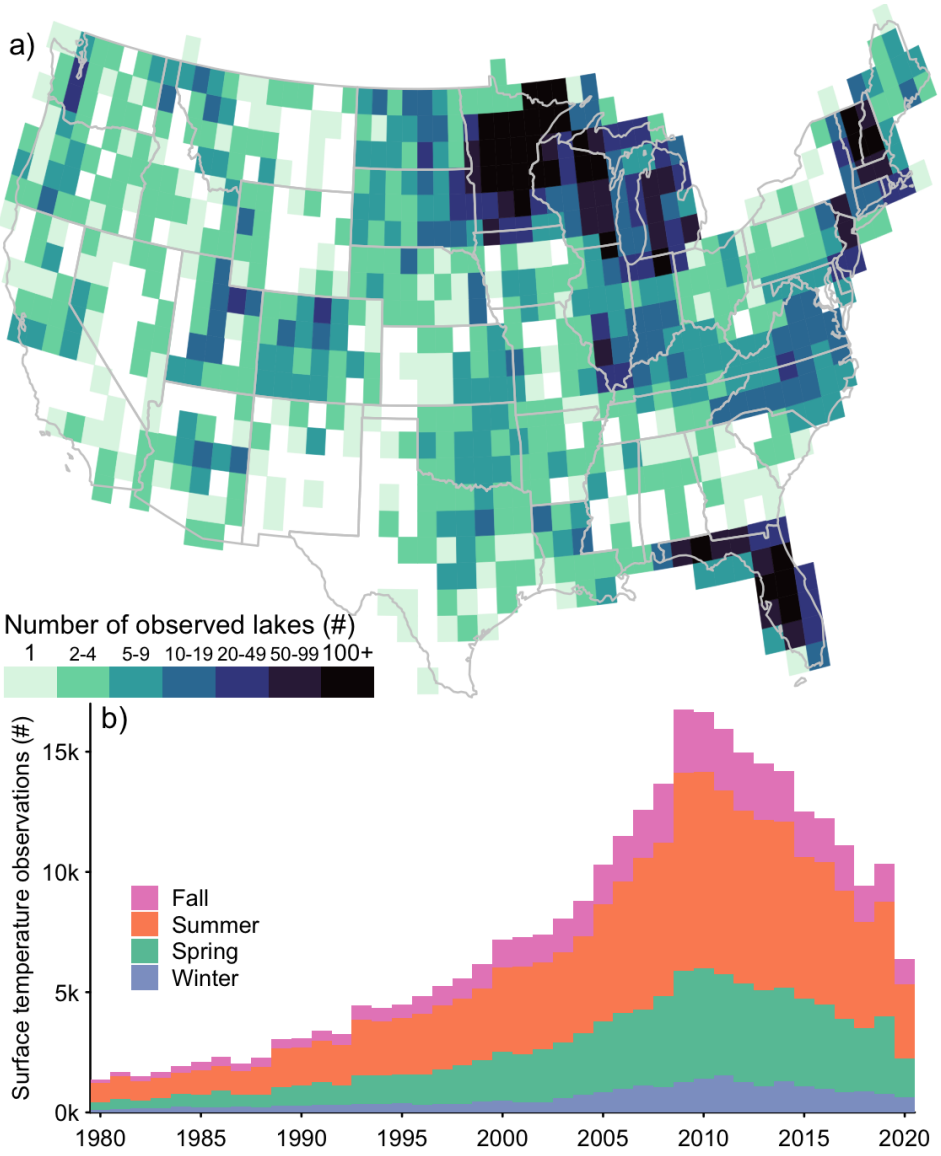


Figure 4.2: Geographic and temporal coverage of in-situ surface temperature data. Panel (a) shows geographic coverage of the 12,227 observed lakes across single degree latitude and longitude cells in the conterminous United States. Panel (b) shows observations by season and by year between 1980 and 2020.

4.2.4 Oversampling

Only 955 temperature observations (0.3% of total) were greater or equal to 33℃. To compensate for a lack of very high temperatures leading to an observation distribution imbalance, we used a simple random oversampling method. Oversampling duplicates samples from a minority class, or in this case a minority temperature range, to address data imbalances for statistical or machine learning models [306, 307]. First, we defined a histogram with forty 1℃ bins covering the range of temperatures from 0 to 40℃. Then, a normal distribution curve was fit to the histogram (mean μ=20.32, standard deviation σ=6.89). The normal curve is a common distribution to use for oversampling [308] that includes a smooth decline with an asymptote at 0°C. For each temperature bin between 33 and 40℃ with sample counts below the normal curve, we randomly oversampled with small added noise (0.0125/0.125 variance Gaussian noise on normalized features / unnormalized observations, respectively) until the bin height matched the mean of normal curve points at both sides of the bin. This added an additional 20,377 (6.6%) observations ranging from 33℃ to 40℃ to the final training dataset. For the cross-validation setup used for hyperparameter tuning and error estimation described in the following subsections, oversampling was specifically done on only the training data and no observations from the test data were duplicated.

4.2.5 Hyperparameter tuning

As with most deep learning models, EA-LSTM requires tuning of hyperparameters for optimal performance. In machine learning, a hyperparameter is a parameter used to control the learning process and/or the network architecture. By contrast, the values of other parameters (typically network weights) are tuned during training. Here, we tuned the hyperparameter that defined the number of epochs used to train the model and also recorded the training MSE at the optimal number of epochs as another stopping condition. The number of epochs was tuned within the inner loop of the 5-fold nested cross-validation [309] shown visually in Figure 4.3. To ensure lake diversity representation across folds, the lakes were first divided into 16 clusters using k-means clustering [310] on latitude, longitude, and the natural log of the surface area values that had been z-score normalized to a mean of 0 and standard deviation of 1. Each cluster was then

equally divided among the 5 folds to create the final fold groupings. The oversampling method previously described was used in each training dataset. The number of epochs was found for each of the 5 test folds by calculating where the mean validation MSE across the remaining 4 training folds was the lowest. The optimal values of epochs for each of the folds 1-5 were 250, 160, 210, 280, and 280, respectively. We also computed the mean training MSE across the 4 folds at the optimal epoch of each instance as another measure of model fitting which were 1.98, 2.01, 1.98, 2.02 (taken from previous studies on lake temperature prediction using LSTM [311, 17], and 2.10℃ respectively. Other EA-LSTM hyperparameters set were a sequence length of 350 days, 256 hidden unit size, learning rate of 0.005, use of the Adam optimizer [312] and an MSE loss function, gradient clipping set to 1.0 of the 2-norm of the network weights, and a batch size of 3000 sequences. All final values are also captured in the modelling code release (`https://doi.org/10.5281/zenodo.6210917`).

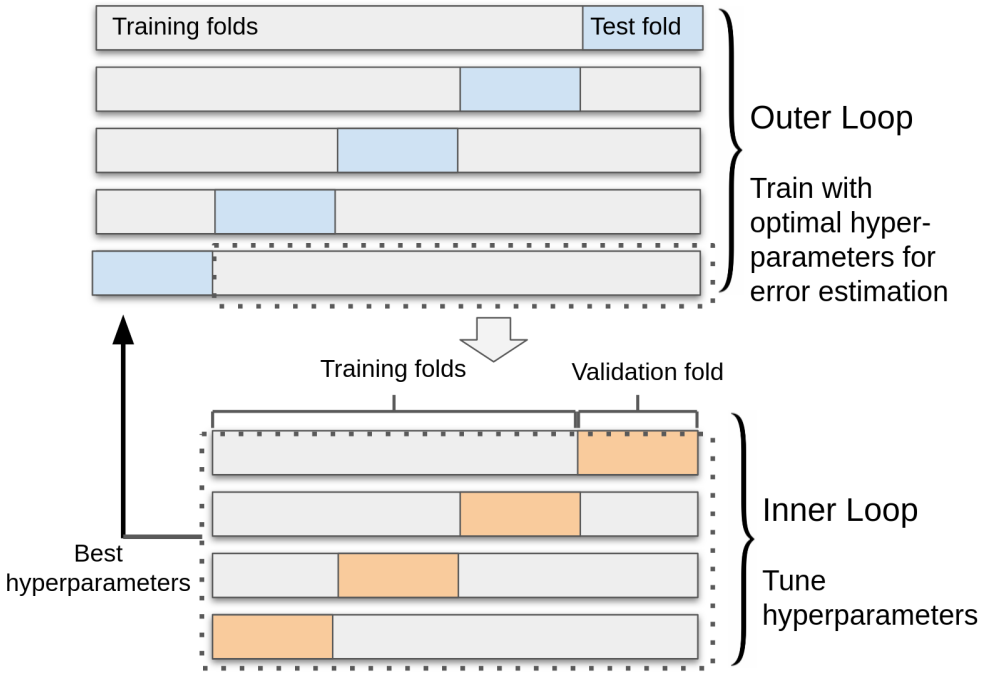


Figure 4.3: Nested cross-validation process. Performance is aggregated over the 5-fold outer loop where each instance of training folds also contains an inner 4-fold loop for hyperparameter tuning on validation data. Hyperparameters are selected to minimize error across validation folds.

4.2.6 Error estimation

To estimate model performance for the two data-driven approaches, we used the outer loop of the 5-fold cross-validation shown in Figure 4.3 and compared the mean out-of-fold test error across folds for each model. Each set of test data was held out of any model training or hyperparameter tuning, and also the 70 lakes not covered by ERA5 were included in training but excluded from test error calculation. Hyperparameters for each of the 5 models were found through the inner cross-validation loop described previously, and training data consisted of observations from the remaining 80% of lakes that were not included in the test fold. The previously described oversampling

method was also used to augment each training dataset with more high temperature observations, and the data splits for EA-LSTM and LM were identical. Compared to the following LM fit published in Bachmann et al. [3],

$$\hat{T} = 16.14 + 0.673\text{Air} - 0.0846\text{Lat} + 0.0172\text{Long} - 0.00131\text{Elev} - 0.147\text{Mon} \quad (4.1)$$

the average over the folds of CV for each of the coefficients (air temperature (Air), latitude (Lat), longitude (Long), elevation (Elev), and month(Mon)) became the following equation:

$$\hat{T} = 20.368 + 0.580\text{Air} - 0.159\text{Lat} + 0.0347\text{Long} - 0.0015\text{Elev} - 0.177\text{Mon} \quad (4.2)$$

For the ERA5 process-based model used for comparison, we also bias-corrected the output by adding 3.31℃ to all predictions (referred to as ERA5*). This bias correction addressed a clear cold bias that currently exists in ERA5 in U.S. lakes (e.g., Betts et al., [313] found a 4°C cold bias of ERA5 on Lake Champlain in late spring; Muñoz-Sabater et al. [2] reported a general cold bias across many lakes). The amount of bias correction was decided based on the intercept of a linear regression with slope 1 fit to observed versus ERA5 predicted temperatures.

4.2.7 Training EA-LSTM and prediction of 185,549 lakes

The final model used to generate predictions for 185,549 lakes was trained on all available surface temperature observation data from 12,227 lakes. Hyperparameter values that minimized validation error across all inner loops in the nested cross-validation were selected for the final aggregate data model (220 for the number of training epochs and 2.03 for the training MSE stopping condition). The remainder of the hyperparameters and model architecture were kept the same as during the error estimation phase, and oversampling was also applied. Using the trained model, predictions were generated for all 185,549 lakes.

4.2.8 Technical Validation Methods

We used the test data from the error estimation phase to estimate overall prediction accuracy, in addition to analyses of accuracy across geographical regions in the United States, different water temperature ranges, different years and seasons, and different lakes. We also sought to identify potential data concerns or limitations that may affect future users of this data. All technical validation described here is transparent and reproducible using the code repositories linked at the beginning of the paper. Technical validation performed includes the error estimation for modeled temperature, assessment of model bias in various conditions, and the quality assurance and quality control (QAQC) procedures for building the in-situ dataset.

The previously described error estimation method was the primary validation of overall accuracy, where all prediction errors were calculated on lakes not used for model training or hyperparameter tuning to mirror the situation of predicting on unmonitored lakes. The folds and clusters used to divide the lakes for training and validation are representative of the broader population of lakes due to (1) the k-means clustering grouping lakes with respect to geographical location and lake size, and (2) the even split among each cluster distributed evenly among the testing folds.

Observed temperature data were screened and unrealistic values were removed using a variety of techniques, including visual inspection, comparison to published models, and evaluating based on date or season to find likely errant data sources. While some of these steps were manual (e.g., visual inspection and contacting monitoring organizations to confirm and fix errant data entry), all alterations to the data, including unit conversions and data screening, were captured in code (see "lake-surface-temperature-prep" code at `https://doi.org/10.5281/zenodo.6210917fordataprocessing`). In the Water Quality Portal data, numerous sites had data that were entered incorrectly for some or all measurements (see Sprague et al. [314] for an overview of similar issues with nutrient metadata). Any observations that likely represented conditions from environments other than the lake water were removed, including by examining metadata fields or contacting data contributors directly. Patterns in temperature time series that suggested the data were flawed were also used to remove values and sites; sites were removed based on various visual or statistical cues (e.g., single measured values that repeated without any deviation) that suggested all site data were suspect. Additionally,

the lower resolution (0.25° lat/lon) aggregated version of the ERA5 temperature estimates were used to determine extreme outliers based on exceeding 10°C above or below a bias-corrected temperature estimate (ERA5+3.47°C) [315] and the resulting outliers were removed from the dataset. If more than one observation was reported on the same day at the same depth on the same lake, we applied the following strategy: we selected the shallower observation followed by the warmer measurement (in the case of identical depths).

4.3 Results of Technical Validation

After outlier removal and the selection of single values to represent a unique lake on a given date, the final dataset of observed temperatures included 306,553 near-surface (between 0-1 meters deep, inclusive) observations from 12,227 lakes. Outliers removed include the following: (1) 7,056 values were removed because "Temperature at lab" was mentioned in the "ResultCommentText" even if the other metadata indicated the measurement was made from the lake, (2) 7,464 additional values were removed that included "Lab" in the "ResultAnalyticalMethod/MethodIdentifier" field as this metadata value indicated these observations of temperature were related to a laboratory measurement or extraction of another variable, (3) 3,746 values from all monitoring sites prefixed with "IL_EPA" and a "CharacteristicName" of "Temperature, sample" were removed after confirmation that these temperatures were not measured directly from the lake, (4) 961 values were discarded when several monitoring sites from various agencies were removed after discovering the data were unrealistic (these sites were removed based on visual comparison to neighboring sites, because values were repeated constantly throughout the season without changing, or because reported depths were likely referenced from the bottom of the lake instead of the surface), and (5) 981 additional values were removed because they exceeded 10°C above or below the bias-corrected aggregated ERA5 temperature estimate. Despite this effort to remove errant data, it is very likely that observation errors beyond the expected range of sensor accuracy still exist in the final dataset, but we expect these issues are rare by comparison.

For the 12,227 lakes with observed temperature, 70 did not overlap ERA5 grid cells (these lakes were near coastlines), and were not included in model evaluation. The

			Median RMSE (℃) by Lake Size (ha)				Median Bias by Observation Temperature (℃)			
	Median lake-specific RMSE	Overall RMSE	< 10 (1,901)	10-100 (6,638)	100-1000 (2,937)	> 1000 (681)	0-10 (27,219)	10-20 (93,949)	20-30 (167,634)	30+ (14,777)
EA-LSTM	**1.24**	**1.61**	**1.24**	**1.18**	**1.27**	**1.60**	**0.38**	**0.29**	**0.10**	**-0.08**
ERA5*	1.79	2.34	1.74	1.76	1.83	2.17	NA	NA	NA	NA
ERA5	3.95	4.06	3.79	4.04	3.94	3.43	-2.34	-3.28	-3.42	-3.02
LM	2.01	2.35	1.70	1.98	2.10	2.14	NA	2.29	-0.39	-0.51

Table 4.1: Performance comparison of the three modelling approaches across the five test folds in cross-validation. Here, ERA5* is the bias-corrected version of ERA5 (an offset of +3.31°C was applied to the ERA5 data), and LM is only tested on data from June to September. From left to right, Median lake-specific RMSE and overall RMSE assess overall performance, then median RMSE is shown for lakes within different size ranges, and lastly median bias of all observations in different temperature ranges is shown (all values are in ℃). Bias for bias-corrected ERA5* is not shown because observations were used in the bias correction itself, and bias in the lowest temperature range is not shown for LM due to lack of data. Numbers in parentheses represent the number of lakes (lake size) and observations (temperature group) in each data partition with the exception of the LM observations, which are lower due to their restriction to the summer months, and the ERA5 comparisons, which have 2,974 fewer observations from 70 coastal lakes that are not resolved in the dataset.

remaining 12,157 lakes and 303,579 observations had a median lake-specific RMSE (1st to 3rd quartile) for all test folds of 1.24℃ (0.86℃ to 1.73℃) for EA-LSTM and 3.95℃ (3.12℃ to 4.84℃) for ERA5 (Table 1). After addressing the cold bias of ERA5 by subtracting 3.31°C (denoted ERA5*), the median lake-specific RMSE of ERA5* was 1.79℃ (1.25℃ to 2.57℃). The original Bachmann et al. [3] model was constrained to periods between June 1 and September 30th, which was followed by re-training and evaluating that model only using observations from those months. The associated data released with that study was also limited to those months and was on a smaller scale than is shown here (Bachmann et al., [3] used 1905 lakes). Here, LM predictions had a lake-specific median RMSE of 2.01℃ (1.32℃ to 2.57℃), compared to 1.17℃ (0.78℃ to 1.68℃) for EA-LSTM and 1.70℃ (1.12℃ to 2.43℃) for ERA5* during the same months. Overall RMSE for the summer months was 1.55℃ for EA-LSTM, 2.27℃ for ERA5*, and 2.35℃ for LM. All other presentations of LM predictions hereafter (in figures and text) are restricted to this time period as well. 534 lakes had observations only outside the summer period and were excluded from the LM error calculations.

The global accuracy of each model (assessed by calculating the RMSE of all data across all test folds at once) was 1.61°C for EA-LSTM, 2.34°C for ERA5*, 4.06°C for ERA5, and 2.35°C for LM (Table 1; Figure 4.4b, 4.4e, 4.4h). The cold bias in ERA5 is greatly reduced by applying a simple offset of +3.31°C to all ERA5 predictions (RMSE of 4.06°C to 2.34°C; Table 1; Figure 4.5e). Spatial patterns in prediction accuracy (estimated by calculating RMSE from test fold data in 1° latitude/longitude cells) showed no clear latitudinal differences for EA-LSTM and ERA5* but temperature predictions from the LM were more accurate in the southern state of Florida compared to the similarly data-rich states of Minnesota and Wisconsin (Figure 4.4a, 4.4d, 4.4g). Predictive accuracy varied over time. Year-specific RMSE for EA-LSTM decreased through time; the maximum single year RMSE was 2.30°C in 1980 and minimum was 1.41°C in 2019, with a clear negative trend (Figure 4.4c). Yearly ERA5* and LM RMSEs did not have a clear temporal trend (Figure 4.4f, 4.4i) and ranged from 2.13°C to 2.89 and 2.09°C to 2.76, respectively.

Predictions from all three models were biased for some or all data subsets (Figure 4.5). Temperature predictions from the ERA5 had the greatest overall bias (specifically,

the model was biased cold for all data subsets). The median bias across ten degree temperature bins ranged from -0.08℃ to 0.38℃ for the EA-LSTM and -0.51℃ to 2.29℃ for LM (Table 1). Bias was greatest for all models for the coldest and warmest temperatures when finer temperature bins were used (Figure 4.5b, 4.5e, 4.5h). The EA-LSTM model had a consistent warm bias across all years (Figure 4.5a). When evaluated across temperature bins and seasons, predictions from EA-LSTM were most frequently warm biased, although cold biases existed for both cold/winter conditions and for extremely warm temperatures, which were substantially underpredicted by the model (Figure 4.5b, 4.5c). The warmest temperatures were underpredicted by both LM and ERA5 models as well (Figure 4.5e, 4.5f, 4.5h, 4.5i). The LM overpredicted temperatures at the lower end of the temperature distribution (5h), but these temperature conditions were rare in the truncated June to September datasets that the LM model was trained and evaluated on (for example, the 4.86°C median LM warm bias in the 10-12°C observed temperature range is based on 0.3% of test observations). Similarly, the extremely warm observations that all models struggled to reproduce were comparatively rare, as the -1.90°C, -4.03°C, -1.01°C LM, ERA5, and EA-LSTM median biases in the 32-34°C range included only 0.8% of test observations and only 0.1% of data were in the 34-36°C temperature range.

The complete set of 306,553 observations were validated against the final EA-LSTM model trained using all of the same data to see if the model was overfitting and verify prediction performance. The median lake-specific RMSE (1st to 3rd quartile) was 1.17℃ (0.82℃ to 1.63℃) indicating a small decrease in error and suggesting overfitting of this model is unlikely.

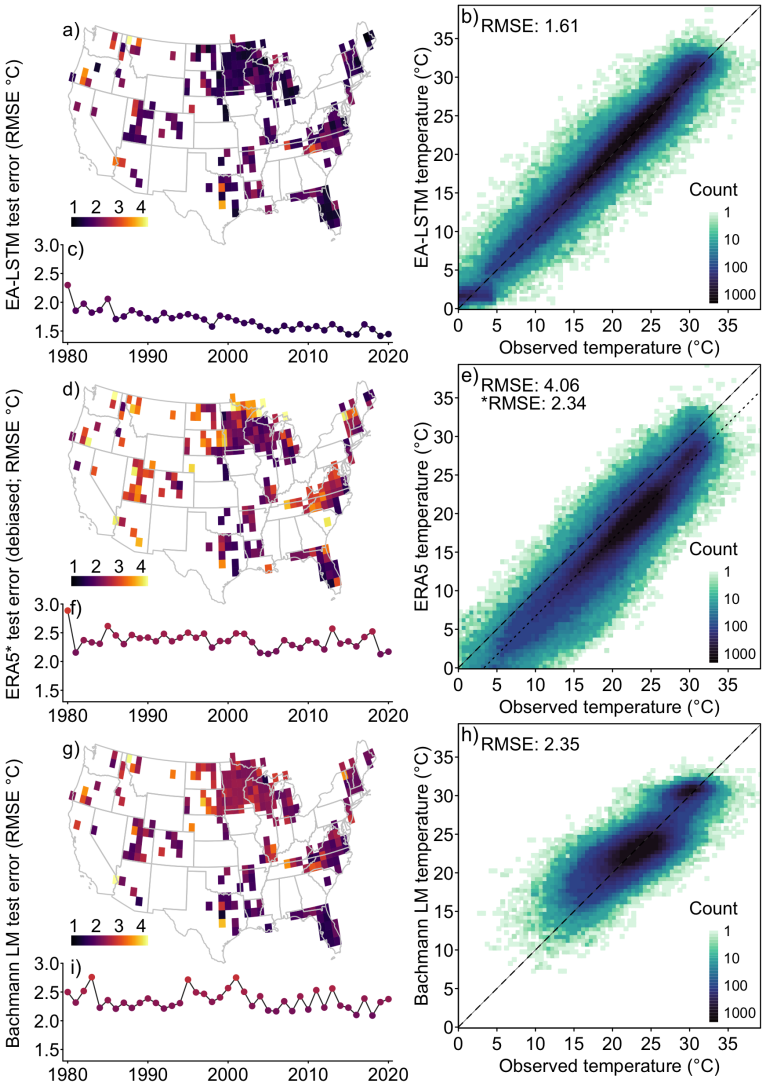


Figure 4.4: Root mean square error (RMSE) for predicted compared to observed water temperatures within a single degree latitude and longitude cell for each of the three methods is shown in panels (a), (d), and (g). Only cells with at least 100 observations are shown. Panels (c), (f), and (i) show year-specific RMSE per modelling method. Panels (d) and (f) show the bias-corrected ERA5 errors (ERA5* in Table 1). The distributions of all 303,579 observations along with a 1:1 line are shown in panel (b) for EA-LSTM, panel (e) for ERA5, and (h) showing the same for LM but with only summer months included (n=187,774 observations). An additional 1:1 dotted line is shown in panel (e) with a y-intercept of -3.31 to represent the ERA5* bias-correction.

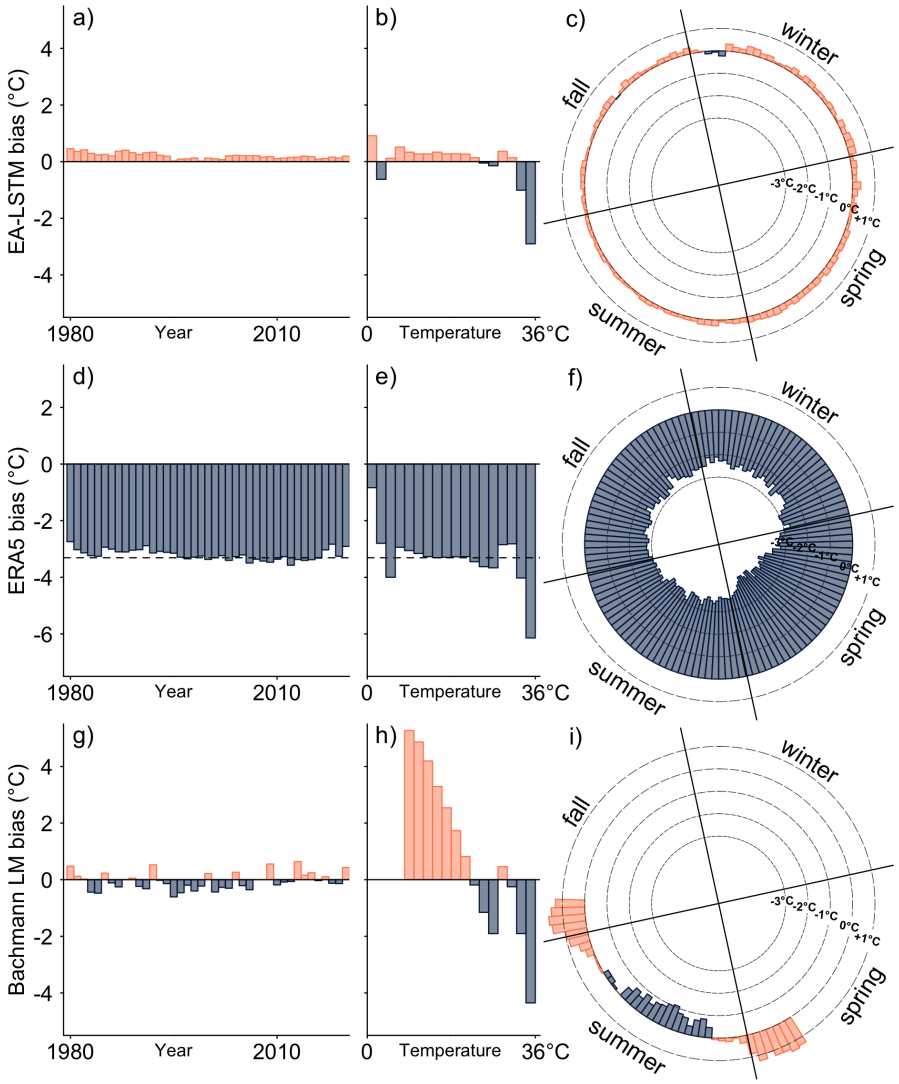


Figure 4.5: Bias of predicted compared to observed water temperatures for all three approaches. Panels (a), (d), and (g) show median bias per year ranging from 1980 to 2020. Panels (b), (e), and (h) show bias per 2℃ temperature bins ranging from 0-36℃. Day of year median bias is shown in panels (c), (f), and (i) with bins covering three days and positive and negative bias visualized as pointing outward and inward, respectively. Dashed rings denote biases at different radii of the plot and lines separate seasons (January 1st is the top of these plots). The dotted line in panels (d) and (e) represents the -3.31℃. shift for bias corrected ERA5 predictions (ERA5* in Table 1).

4.4 Data Use and Recommendations for Reuse

Surface water temperature estimates are useful for improving scientific outcomes in fisheries biology, limnology, and climate science. Specifically, these data 1) facilitate improved understanding of lake temperature dynamics in under-monitored and unmonitored locations, 2) enable investigators to scale up from traditional single or multisite field science to science at broad spatial scales, and 3) extend a foundational limnological data resource (LAGOS-US; [316]) by linking these weather and temperature predictions to numerous lake properties through common lake identifiers. Across applications, this dataset provides the best available surface temperature accuracy at the scale of the conterminous United States. Additionally, example data access scripts for both Python and R are included in the "lakesurf-data-release" code at `https://doi.org/10.5281/zenodo.6210917` to facilitate future users.

At the local to regional scale, this dataset provides essential data to parameterize models that use surface water temperature as an input (e.g., harmful algal bloom prediction [317], gas solubility estimates [318], and fish bioenergetics models [319]. This dataset has the potential to similarly inform improvements to other limnological data products by refining ancillary temperature estimates, including satellite derived surface temperatures [320, 321, 304]. When combined with additional observational data, the historical reconstruction of temperature provided here can further our understanding of how temperature correlates with overall water quality dynamics, nutrient loading [322], and algal bloom frequency [323].

This landscape-scale dataset could support a more systematic understanding of the extent of lake synchrony in response to multi-scale forcings like climate and land use change [324]. Lake temperature as a major ecological control is also important for quantifying other macroscale ecosystem properties, such as the contribution of aquatic ecosystems to continental and global carbon cycles [325, 326, 276]. Existing approaches for quantifying lake contributions to carbon budgets rely on spatiotemporally inconsistent data (including temperature) and can be substantially improved by using comprehensive landscape-scale datasets [327].

Across scales, surface temperature estimates can be used to estimate changes in thermal parameters related to fish spawning, growth, and abundance. Previous work

has shown that population dynamics of cool- and warmwater fishes are well-predicted by surface temperature metrics (e.g. [328, 329], even in stratified lakes with diverse thermal habitats. However, changes in surface water temperatures alone may be a poor proxy for estimating changes to the thermal environment of coldwater fishes or other organisms occupying the bottom waters of stratified lakes [330, 331]. Understanding these shifts in thermal regimes will become increasingly important as climate velocities (the pace of warming compared to a species' ability to migrate to cooler habitats) increase throughout the next century [332, 333]. This dataset provides an essential baseline of historical temperatures upon which to compare these future changes.

4.5 Comparison with existing datasets

Daily surface water temperature predictions for lakes in the conterminous United States using the EA-LSTM are more accurate and less biased when compared to currently available models with similar or greater temporal and spatial coverage (Figure 4.4; Figure 4.5). The EA-LSTM outperformed ERA5 and LM temperature predictions based on the RMSE of all data subsets assessed, including global RMSE and RMSE for binned lake size classes (all observations; Table 1). Spatially, the EA-LSTM was best for 74% (63% accounting for ERA5*) of the 12,157 lakes used for model evaluation, as well as across 82% (80% including ERA5*) of the 220 1° latitude/longitude cells that had at least 100 observations (Figure 4.4). The EA-LSTM maintained the lowest RMSE across all 41 study years regardless of ERA5 debiasing.

We found a significant cold bias in ERA5 predictions that was similar for all years (Figure 4.5d), but varied seasonally and across the range of temperatures (Figure 4.5e, 4.5f) which is consistent with Betts et al. [313]. Bias-correction may be needed for most uses of the current ERA5 mixed layer temperature predictions. The EA-LSTM outputs included in this dataset have a small warm bias that is mostly consistent seasonally and across years (Figure 4.5a, 4.5c), but predictions are cold compared to the warmest observations and warm compared to the coldest observations (Figure 4.5b). The Bachmann LM model had no bias across years (Figure 4.5g), but was substantially biased across the range of temperatures, overpredicting colder temperatures and underpredicting warmer temperatures (Figure 4.5h), and this pattern also appeared as strong

seasonal model biases (Figure 4.5i).

The accuracy of the EA-LSTM predictions compare favorably to other efforts on smaller numbers of lakes, including the global analysis of 235 lakes by O'Reilly et al. ([334]; RMSE 1.68°C-2.15°C from linear regression), and regional process-based predictions of temperatures by Winslow et al. ([335]; epilimnetic temperature RMSE of 1.91°C; n=72,232). Recent summer surface temperature predictions for 2,186 U.S. lakes by Kreakie et al. [292] had similar accuracy to the EA-LSTM (1.48°C vs 1.50°C RSME when comparing summer errors in 2007 and 2012, the two years of their model) but the performance of their random forest model was not evaluated on unseen lakes and conditions (e.g., the additional 39 years and 10,041 lakes included in this study). Satellite-based remote sensing sources of estimated surface temperature are promising, and can approach the accuracy of the EA-LSTM model presented here in certain cases (e.g., Schaeffer et al. [304] found mean absolute error (MAE) of Landsat water pixels ¿180m from shore was 1.34°C; the EA-LSTM presented here has an MAE of 1.16°C).

The in situ measurements shared in this dataset have two orders of magnitude more observations compared to those made available in Bachmann et al., ([3]; 306,553 and 2,655 observations, respectively) and an unprecedented number of U.S. lakes (12,227 lakes). While the in-situ data in this dataset can be accessed elsewhere, the significant effort to query, download, and screen data, in addition to the process to match temperature monitoring sites to individual lakes has resulted in a dataset that can be rapidly leveraged for future studies. Specifically, the QAQC of data from the Water Quality Portal [24] and site-linking to lakes adds substantial value to those existing resources. A similar global compilation effort by Sharma et al. (2015) produced summer temperatures and metadata for 291 lakes that has been used extensively to quantify the effect of climate change on lake temperatures (e.g., [336, 334]), and we expect these in-situ data to also support new aquatic science efforts. The dataset described in this article does not include data collected using automated sensors nor remotely sensed data, but either could be combined with these observations to extend the dataset.

The predicted surface temperatures for 185,549 lakes includes full coverage of lakes with surface area larger than 4 ha in the conterminous United States, which is a substantial expansion in scale or resolution compared to other available modeled temperature data products. The ERA5-simulated epilimnetic lake temperatures provide coverage of

the great majority of lakes globally, but the gridded cells overlapping the lake centroids of this conterminous United States dataset have far fewer unique timeseries (42,354 for ERA5 versus 185,549 here). Many of the ERA5 0.1° latitude and longitude grid cells aggregate multiple lakes into the lake tiles that are available in the ERA5 dataset. However, the ERA5 dataset does include hourly temperatures that could be useful for comparing minimum and maximum temperature ranges; our model generates a single prediction for each lake-day. Other existing process-based lake temperature predictions from Winslow et al. [335] and Read et al., [303] cover a smaller spatial extent, and within those regions, represent a smaller number of lakes due to a requirement parameterizing lake depth for the individual models. Semi-process-based approaches have been applied at a larger scales with good results in Gillis et al. [337] and also with the air2water model [338, 339, 340]. However, these approaches are also limited by the requirement of lake depth which is readily available only for a small subset (n = 17,675) of all lakes in the conterminous United States for lakes with surface area bigger than 1 ha (3.7% of 479,950) that are available in LAGOS-US [341]. The ERA5 predictions overcome this limitation by using an estimated lake depth product that is available globally [342]. Our modeled temperatures have a similar coverage to the possible extents of the data-driven approach of Bachmann et al. [3], but those models were not released with predictions or inputs beyond the observed lakes used for training and testing the models and are additionally limited to the summer months.

We used NHD HR permanent identifiers to enable synergistic interactions with existing datasets including LAGOS-US [316], the National Anthropogenic Barriers Dataset [343, 344], and the National Lakes Assessment [345]. Using GIS, the data provided can be linked to additional lake and catchment properties within the Water Quality Portal [24], HydroLakes [346], and the Global Lake Area, Climate, and Population dataset [347]. In combination, these macroscale datasets provide a suite of lake and catchment properties and multitemporal measurements of water quality, anthropogenic stressors, land use, and meteorological variables. This wealth of information creates novel opportunities for modelling lake systems and examining synoptic patterns in freshwater resources at the landscape scale. While the contribution of estimated and observed water temperatures provided here is highly valuable as a stand alone resource, the inclusion of lake-level climate and meteorological data at the daily timescale provides additional

benefits not currently captured in the datasets discussed above.

Leveraging the above interconnected datasets and/or future datasets of lake properties could likely lead to modelling efforts that outperform the EA-LSTM model presented here. With future development in mind, and to maximize the utility of the provided dataset, all modelling inputs, data partitioning, training data, modelling code, and EA-LSTM predictions are accessible through this dataset. By providing this end-to-end pipeline, we aim to create continued opportunities for comparison and modelling improvements. Data such as upstream inflow, reservoir release information, and land use may allow a future model to better capture abrupt changes in temperature or to predict more accurate temperature extremes.

Chapter 5

Transfer learning and broad-scale machine learning modeling of water temperature in unmonitored stream sites

5.1 Introduction

Stream water temperature, known as an ecosystem "master factor" affecting metabolism, water chemistry, and wildlife [348]; is of great interest to water resource scientists and managers. In particular, widespread predictions of water temperature in unmonitored stream reaches is a problem of urgent societal importance and can enable decision makers to enact better responses to changes caused by spontaneous disturbance events. Though hydrological modeling to predict stream water quality (e.g. temperature, salinity) has traditionally relied on process-based models, previous work using parameter transfers and regionalizations from well-observed systems to larger spatial scales or unmonitored basins have experienced mixed success [349]. On the other hand, deep learning models using combined aggregate datasets from heterogeneous regions across the US have experienced success predicting in unmonitored basins for other hydrological applications

such as modeling rainfall-runoff streamflow [281, 167], snowpack dynamics [57], baseflow [58], and dissolved oxygen [59], but efforts in stream temperature modeling at the continental or multi-regional scale are limited. Given the cost of data collection, data-driven models that can efficiently use existing in-situ data and transfer information to unmonitored systems are critical to closing our information gaps [350].

Numerous machine learning (ML) methods exist for prediction in unmonitored locations as mentioned in Chapter 2. Though most of these approaches were developed in rainfall-runoff streamflow modeling due to both societal importance and the wealth of streamflow data compared to other variables like stream water quality, these efforts are expanding as data collection and modeling continue to advance. As described in Chapter 2 and Willard et al. [351], we divide approaches for unmonitored environmental time series prediction into two categories. The first is referred to as *broad-scale* modeling, where the idea is to incorporate inherent characteristics of different entities (e.g. stream sites) to improve prediction performance in a single broad-scale model using all available entities or multiple broad-scale models built on a subgroup of entities. In the context of hydrological modeling, this involves using site characteristics, often treated as static inputs, in multiple possible ways. The most common way is to use characteristics concatenated with the dynamic input forcing data, though there are more complex ways like encoding static characteristics [66] or using a graph neural network to capture dependencies between sites [81]. These direct concatenation of input-based approaches likely stem from a landmark result for streamflow modeling, where Kratzert et al. [352] show an entity-aware long short-term memory (LSTM) model built a large number of geographically diverse catchments within the Catchment Attributes and Meteorology for Large-sample Studies (CAMELS) dataset was able to predict more accurately on unseen data on the same test sites than state-of-the-art process-based models calibrated to each basin individually. Similar results have been seen in streamflow modeling for many other global regions (e.g. [54, 55, 165, 56]) and also in other hydrological variable prediction tasks (e.g. [57, 58, 59]). The LSTM model is the most common type of entity-aware deep learning model for hydrological time series modeling [351], due to its recursive nature and memory structures allowing it to model cumulative environmental system status [19].

The second class of techniques is *transfer learning*, which is a powerful framework

for applying knowledge learned from one task to another, typically to compensate for missing, nonexistent, or unrepresentative data in the new problem domain. The idea is to transfer knowledge from a related task, i.e., the source system, where sufficient data is available, to a new but related task, i.e., the target system, often when data is scarce or absent [113, 114]. In the context of environmental modeling, transfer learning for ML is analogous to calibrating process-based models in well-monitored systems and transferring the calibrated parameters to models for unmonitored systems, which has shown success in hydrological applications [115, 116]. Deep learning is particularly amenable to transfer learning because it can make use of massive datasets from related problems and alleviate data paucity issues common in applying data-hungry deep neural networks to environmental applications [16, 117]. Transfer learning using deep learning has shown recent success in water applications such as flood prediction [118, 119], soil moisture [120], and lake and estuary water quality [121, 1]. Willard et al. [351] note that often these approaches are not compared in detail with each other or sufficiently benchmarked such that researchers know what to use in a given prediction task.

This study in particular focuses on purely ML approaches applied within streams in the United States and compares broad-scale modeling at both the regional scale and continental-scale with meta transfer learning, a data-driven model selection framework to decide source models to transfer to unmonitored sites [1]. Broad-scale prediction models built on a large and diverse training set and transfer learning using accurate models built on highly-monitored sites represent two different but promising directions to be tested. It is not clear which would be the best prediction method in a given scenario, or what the most important attributes and features for predicting and model transfer are. We also evaluate multi-linear regression (MLR) and extreme gradient boosting (XGBoost) as baseline models representing widely applied statistical and classical ML approaches respectively.

5.2 Methods

5.2.1 Data

Stream temperature and meteorological data for this study has been acquired using BASIN-3D data integration tool [353], and the watershed attributes (treated as static)

from the Geospatial Attributes of Gages for Evaluating Streamflow (GAGES-II) dataset [354] augments this data. BASIN-3D includes daily meteorological data from DayMet [355] which constitutes the primary dynamic input data. In-situ stream temperature was acquired from the United States Geological Survey (USGS) stream temperature dataset that spans the conterminous United States [23]. All input and water temperature data was limited to the years from 1980 to 2021 for the purposes of this study.

The input data to the models consists of both dynamic meteorological and streamflow discharge data and also site characteristics which are treated as static (though some, e.g. land use, are realistically dynamic). The meteorological values are day length, air temp (mean, maximum, and minimum), snow water equivalent, vapor pressure, solar radiation, and precipitation; and the discharge data is taken as the log-transformed cubic feet per second value. The site characteristics used in building all the model types were chosen based off the prior LSTM stream temperature modeling study [22]. These consist of 23 expert-chosen values include properties like dam storage density in the watershed, drainage area, stream density, number of dams and average distance to them, land use features, other topographic features, location, and meteorological statistics (for further details see [22]).

The training and testing data was divided as follows; all sites with GAGES-II site attributes and water temperature data spanning at least 5 years were denoted as training data, constituting 973 sites spanning all 18 USGS-defined hydrological regions (see `https://water.usgs.gov/GIS/regions.html`), and the testing data consisted of all sites containing between 1 and 5 years of water temperature data, also with GAGES-II static features, which results in 736 sites spanning all but one of the 18 hydrologic regions. However, since our ML models require consistent inputs for the entire lookback period when making a prediction, a number of water temperature observation were excluded due to either missing streamflow input, missing meteorological input, or both. From the 973 sites selected for training, 491,575 water temperature observations were excluded resulting in 786 remaining sites containing 4,304,977 observations as the final training data. Similarly for the 736 testing sites, 69,151 observations were excluded resulting in 580 remaining sites containing 561,964 observations.

5.2.2 Model descriptions

Here we describe the ML model frameworks and architectures used for stream temperature prediction, as well as any preprocessing, hyperparameter tuning, and feature selection that was done. All of these models are trained on the 787 data-rich sites and then used to predict stream water temperature in the 580 psuedo-unmonitored target systems.

Baseline models

Both baseline models MLR and XGBoost were built using the Scikit-learn python library [261] (for further model details see Gray [356] for MLR and Chen [357] for XGBoost). MLR was chosen as a statistical model due its frequent use for both stream temperature modeling [358, 359, 360, 361] and hydrological prediction in ungauged basins (e.g. [362, 363, 364]), and XGBoost was chosen as a classical ML model due its vast array of successes across many applications [365]. Both models are easy to implement, efficient, interpretable, and can serve as effective baselines.

Since these models are not naturally time series models, additional preprocessing must be done to incorporate lagged features from previous timesteps. For stream water temperature specifically, lagged meteorological features are known to significantly associate with water temperature and improve model performance [366]. We incorporate the selection of lag periods to include within a feature selection framework.

Both models undergo recursive feature elimination with cross validation (RFECV)[260]. Recursive feature elimination is a feature selection method that fits a model and iteratively removes the weakest features until an ideal set that produces the lowest cross-validation error is reached. To do this, we used the Scikit-learn RFECV class [261]. For building the base models, we used LinearRegression and XGBRegressor Scikit-learn classes. Feature selection was performed using the RFECV process with 5-fold cross validation (CV), mean squared error loss, and base models using default parameters. All static and dynamic inputs were designated as candindates for the feature selection, in addition to dynamic input feature having possible lag periods of 1, 2, 3, 5, 7, 14, and 28 days. Here, feature importance within the feature selection was calculated by the XGBRegressor as a measure of how each feature affected mean squared error across

nodes in the decision trees, weighted by how often those nodes are reached, and for MLR by the absolute value of coefficients for the model built on normalized data with mean 0 and standard deviation 1.

After feature selection, we performed hyperparameter optimization only in the case of XGBoost because MLR does not contain hyperparameters. We used 200 iterations of random search across the uniform distribution of two XGBoost hyperparameters, the number of estimators and learning rate. For number of estimators we tested between 50 and 2000, and for learning rate we tested between 0.05 and 0.5.

Broad-scale LSTM

The LSTM is a type of recurrent neural network that includes dedicated memory that can store information over long time periods [367]. This memory function is analogous to a system state vector in dynamical systems and process-based modeling, making it a popular architecture for modeling watershed processes [35, 351]. Compared to other variants of recurrent neural networks, LSTMs are not vulnerable to the problem of exploding and vanishing gradients during training. In contrast to the baseline methods, feature selection and time lags were not incorporated since deep learning contains feature extraction properties intrinsically [368, 369] and LSTM models automatically incorporate past time steps from a lookback period defined during model building. For further details on the LSTM recurrent neural network architecture see Hochreiter et al. [367].

To implement the LSTM model that is additionally aware of modeling a large number of different entities, it is necessary to provide it with information on the stream site characteristics. Different "entity-aware" LSTM architectures have been developed for increased model interpretability and a customized learning ability for catchment-wise adaptation through differentiation between static and dynamic inputs within the LSTM gates [281, 35, 370]. However, we chose to use the standard LSTM architecture alongside a concatenation of static entity-specific inputs and dynamic inputs as it has shown better performance overall in multiple applications [281, 35, 371]. We used the Pytorch [372] LSTM class for implementation.

Broad-scale LSTM models were built both at the continental and regional scale. At the regional scale, the LSTM was built for each of the 18 USGS-defined hydrological

regions. When predicting in unmonitored sites at the regional scale, each target site is mapped one of the 18 corresponding broad-scale model built on the region it lies in.

Hyperparameter tuning for each LSTM consisted of a 10-fold cross validation random search run under a 20 hour timeout across 8 Nvidia A100 GPUs. The distributed hyperparameter tuning routine was implemented using the Ray Tune library [373]. The parameter ranges explored included number of hidden units between 100 and 300 and the number of network layers between 1 and 4. Training epochs were decided using an early stopping routine [374]. The early stopping used the final 20% of the training data as validation data and a patience of 300 epochs. The final hyperparameters were 225 hidden units, 3 network layers, a batch size of 300, a sequence length of 200 days, and a sliding window shift of 100 days at a time. The sliding window shift allows for loss to only be calculated on the latter 100 days in the 200 day sequence for each data tensor, allowing the LSTM sufficient time to build up memory prior to predicting. Furthermore, L1 and L2 regularization along with dropout rate were set to 0, initial weights were set using the Xavier normal distribution [375], and the AdamW [376] optimizer was used with a mean squared error loss function.

ML models also have uncertainty in model parameters after training due to a variety of factors including stochasticity in model intitialization and shuffling data between training epochs among other factors. Since it has been shown that ensemble results from multiple model runs will facilitate better overall ML model performance and robustness and also allow for the quantification of uncertainty [377], the stream temperature prediction result for all model types in the following sections will therefore be an ensemble average of five model realizations.

Meta transfer learning using LSTM source models and XGBoost meta-model

Meta transfer learning is a framework that addresses the fundamental challenge in transfer learning which is to either decide which model to transfer from a known related task or how to build a transferable model [252, 1, 133]. It accomplishes this by using meta-learning [129, 130], which is a type of learning that learns from other ML models and ML modeling experiences. One way a meta-model can be formulated is by using meta-features about a source site model (e.g. model structures, source site characteristics, or input data statistics) to predict model performance (e.g. prediction error). Here, we

build a metamodel in the same manner based on previous works Willard et al. [1] and Ghosh et al. [133]. We describe the method at a high level, but further details can be found in Willard et al. [1]. In summary, meta transfer learning follows the following procedure for water temperature prediction in stream sites in the United States

1. Build and train five source LSTM models for each of the 787 well-monitored stream site.

2. For each source site, use all 786 models built on other individual source sites to predict daily stream temperatures and evaluate prediction accuracy.

3. Train the meta-learning XGBoost regression model to predict the 787*786 collected model RMSE performance values from (2) based on the stream site characteristics as meta-features that we hypothesized could be important for selecting good transfer models.

4. Given an artificially unmonitored stream site, where data is only used for final evaluation, and its meta-features, use the meta-learning model to predict model performance of each source model. Use the source models with the lowest predicted errors to model the target.

In this work we perform this procedure twice, differing only within step (1). The first framework will train LSTM source models on each stream site individually, and the second will incorporate an additional pre-training stage on all available training data which is then fine-tuned using data from the specific site. We will refer to these as *MTL* and *MTL_PT* (PT=pre-train) respectively.

LSTM models were chosen as source models for the meta transfer learning framework based on their recent surge of successful demonstrations for water resources time series ML models [351] and also successful adoptions within meta transfer learning frameworks [1, 133]. Unlike many classical ML and feed forward neural networks models, LSTM also does not require tuning and selection of time lag input input features due to its recurrent structure. Performing detailed hyperparameter tuning on all 787 source models would be computationally infeasible. Also, in the same manner as the broad-scale models, we train 5 different model realizations for each source site. Furthermore, some pre-training and fine-tuning frameworks in transfer learning ML will freeze layers in the neural

network after pre-training, aiming to preserve general information from the pre-training within the first layers, and only alter the final one or more layers during fine-tuning [123]. Since our LSTM model has three layers, we tested freezing either the first or first two layers after pre-training but found it did not help performance within cross validation on the training dataset.

Due to its success in the previous study Willard et al. [1], ease of implementation, GPU acceleration capability, and ability to illustrate the relationships between predictors and the response, we chose extreme gradient boosting (XGBoost) regression to predict the RMSE of source-target pairs from meta-features.

The XGBoost meta-model underwent both hyperparameter tuning using random search and feature selection using RFECV in the same manner as the XGBoost base model described earlier. These meta-features consisted of the same static features used in the previously described broad-scale model, in addition to dynamic input statistics (e.g. mean and standard deviation of min/max air temperature), and source quality features (e.g. number of water temperature observations available for training data).

Given an unmonitored target site, we select the 10 sites that contain the source models with the lowest predicted error by the meta-model. Selecting more than one source model and combining them in an ensemble was found in Willard et al. [1] to significantly outperform the single source model transfer. These source models are combined in an ensemble and the final predictions are an average of the 50 models (5 model realizations per source site * 10 selected source sites).

5.2.3 Experiment description

We evaluate performance of the six different previously described modeling frameworks in a real-world scenario: predicting water temperature in unmonitored stream locations. The six models are hereby referred to after as *MLR_conus* (continental-scale multi-linear regression), *XGB_conus* (continental-scale extreme gradient boosting), *LSTM_conus* (continental-scale LSTM), *LSTM_regional* (regional-scale LSTM), *MTL* (meta transfer learning), and *MTL_PT* (meta transfer learning with pre-training). We use the data described in Section 5.2.1 as the training and testing data split policy described there.

We examine performance in terms of the RMSE when predicting the water temperature of the 580 psuedo-unmonitored stream sites compared to the baseline models.

Although useful, other error metrics such as Nash–Sutcliffe model efficiency coefficient (NSE) and estimates of bias or variance were not included in the study. In addition, we also analyze the metamodels themselves used in *MTL* and *MTL_PT*. Test data was only used to independently evaluate the accuracy of temperature predictions after training and metamodeling was complete, and the test dataset had no influence on the model training phase.

We also calculate feature importances for both the primary prediction models and the two metamodels in the *MTL* and *MTL_PT* frameworks. For MLR_conus we use the absolute values of the coefficients for normalized input data [378], and for *XGB_conus* and also the two metamodels we used the Python XGBoost library's "gain" feature importance (default in version 1.7.4 [357]), which is a measure of how each feature increased accuracy across nodes in the decision trees, weighted by how often those nodes are reached. For the LSTM models, *LSTM_conus* and *LSTM_regional*, we used permutation feature importance [379]. Permutation feature importance is calculated as the increase in the prediction error of the model (e.g. RMSE) after the feature's values are randomly permuted, which breaks the relationship between the feature and the true outcome but maintains the feature distribution. Notably, the feature importances for the baseline models include explicitly defined time-lagged features, but since the LSTM intrinsically includes memory of past time steps, the permutation importance is defined generally across all time steps. Furthermore, for the LSTM permutation importances, we include three "combined" permutation of features that are strongly related or transformations of one another. So, "combined air temp" would be the combination of maximum, mean, and minimum daily temperature (*tmax, tmean, tmin*), "combined discharge" would be the combination of river discharge (*rdc*) and log-transformed river discharge (*logrdc*), and "combined precipitation" would be the combination of precipitation *(prcp)* and log-transformed precipitation (*logprcp*). We include these combined features so that the correlation of features doesn't strongly affect the importance of a single permuted feature when the other features are still present and informative.

5.3 Results

5.3.1 Performance on 580 test stream sites

Table 5.1 gives an overview of the 4 applied ML models and the two benchmark models *MLR_conus* and *XGB_conus*. Of the 580 test sites *MTL_PT* performed the best for 148 sites, *LSTM_conus* for 143, *MTL* for 102, LSTM_regional for 90, *XGB_conus* for 60, and *MLR_conus* for 37 sites. A spatial depiction of the top two performing methods across the 580 test sites is shown in Figure 5.2. We see a median per-site RMSE of 2.14°C and 1.66°C for the *MLR_conus* and *XGB_conus* models respectively, 1.62°C for LSTM_regional, 1.45°C for *LSTM_conus*, and 1.65°C and 1.41°C for the MTL-based frameworks *MTL* and *MTL_PT* respectively.

Performance per USGS-defined hydrological region is shown in Table 5.2. We see of the 17 regions, *MTL_PT* had the lowest RMSE for 14 of the regions, with the exceptions being *XGB_conus* performing the best for the Lower Mississippi, and similarly *MLR_conus* for the Lower Colorado and *LSTM_regional* for Great Basin.

We also examine the performance of each model per year Figure 5.1. For reference, a graph of the number of water temperature observations per year can be found in Appendix Section A.1.

5.3.2 Prediction performance

Method	Median per-site RMSE (°C)	RMSE standard deviation (°C)
MLR_conus	2.14	0.97
XGB_conus	1.66	1.28
LSTM_conus	1.45	1.05
LSTM_regional	1.62	1.85
MTL	1.65	1.09
MTL_PT	**1.41**	**1.03**

Table 5.1: Overall RMSE statistics across the 580 testing stream sites

Region	n_obs	n_sites	MLR_conus	XGB_conus	LSTM_conus	LSTM_regional	MTL	MTL_PT
01 (New England)	9784	10	2.20(0.61)	1.42(0.46)	1.21(0.40)	1.44(0.43)	1.52(0.68)	**1.18(0.41)**
02 (Mid-Atlantic)	88224	121	2.02(0.82)	1.66(0.86)	1.60(1.02)	1.63(1.01)	1.75(1.15)	**1.53(1.00)**
03 (South Atlantic-Gulf)	64533	77	2.02(0.84)	1.74(0.95)	1.34(0.60)	1.54(1.28)	1.57(0.81)	**1.29(0.59)**
04 (Great Lakes)	18522	22	2.04(0.54)	1.38(0.35)	1.24(0.47)	1.45(0.69)	1.44(0.53)	**1.21(0.44)**
05 (Ohio)	28431	41	2.46(0.86)	2.09(1.67)	1.83(0.88)	2.02(0.86)	1.99(0.96)	**1.75(0.83)**
06 (Tennessee)	3174	5	1.76(0.44)	1.20(0.12)	1.11(0.43)	1.50(0.67)	1.32(0.59)	**1.03(0.37)**
07 (Upper Mississippi)	28053	29	2.42(0.90)	2.30(1.64)	1.90(1.00)	2.24(1.48)	1.93(0.89)	**1.83(1.02)**
08 (Lower Mississippi)	3813	5	2.69(1.18)	**1.99(0.66)**	2.20(1.34)	2.99(1.22)	2.34(1.48)	2.16(1.37)
09 (Souris-Red-Rainy)	4188	5	2.75(0.55)	1.96(1.00)	1.47(0.39)	2.24(1.08)	1.49(0.37)	**1.44(0.34)**
10 (Missouri)	46083	54	2.56(0.94)	2.75(4.04)	1.86(1.35)	2.84(4.26)	1.97(1.05)	**1.71(0.84)**
11 (Arkansas-White-Red)	21989	23	2.25(0.56)	2.49(2.87)	1.64(0.55)	1.98(0.78)	1.76(0.46)	**1.54(0.51)**
12 (Texas-Gulf)	15377	16	2.27(0.87)	2.21(0.87)	1.96(1.16)	1.89(1.24)	2.37(1.12)	**1.89(1.07)**
14 (Upper Colorado)	27723	31	2.67(1.08)	2.59(1.17)	2.27(1.29)	2.69(1.63)	2.31(1.32)	**2.19(1.28)**
15 (Lower Colorado)	5742	7	**3.48(3.53)**	3.99(3.70)	3.68(3.52)	3.93(1.93)	3.76(3.57)	3.67(3.58)
16 (Great Basin)	17914	20	2.49(0.52)	2.07(0.99)	2.04(1.10)	**1.87(0.90)**	2.17(0.61)	1.95(1.02)
17 (Pacific Northwest)	67182	70	2.34(0.80)	2.06(1.70)	1.66(0.98)	1.78(1.12)	1.89(0.84)	**1.62(0.92)**
18 (California)	36043	44	2.74(1.29)	2.53(1.05)	2.36(1.07)	3.58(2.13)	2.49(1.22)	**2.31(1.06)**

Table 5.2: RMSE statistics per USGS-defined hydrological region (`https://water.usgs.gov/GIS/regions.html`). Values are the mean RMSE(°C) across sites in a region, with RMSE standard deviation in parentheses. Lowest mean values per region are shown in bold.

5.3.3 Feature Importances

By and large, we see feature importances consistent with existing hydrological knowledge that past and present daily air temperature is the primary driver for water temperature dynamics. Feature importance plots for the baseline *MLR_conus* and *XGB_conus* can be found in the Appendix A.2 and A.3. For the *LSTM_conus* and *LSTM_regional* models, importances are shown in Figures A.4 and A.5. Here, we see the largest importance on air temperature with permutation importance values of 4.08°C and 3.26°C respectively for the permutation of the combined air temperature features. Both models also benefit from the inclusion of day length, with importances of 0.36°C for *LSTM_conus* and .27°C for *LSTM_regional*. Lesser importances for both include combined discharge (0.17°C for *LSTM_conus* and 0.12°C for *LSTM_regional*) and very small importances between 0.01°C and 0.05°C for precipitation, vapor pressure, and solar radiation. All other features were less than 0.01°C importance. The feature importances for *MTL* and *MTL-PT* are shown in Figures A.6 and A.7. Similarly, we see the largest importance on air temperature with permutation importance values of 2.96°C and 4.10°C for the combined air temperature features, 0.18°C and 0.46°C for day length, 0.13°C and 0.18°C for

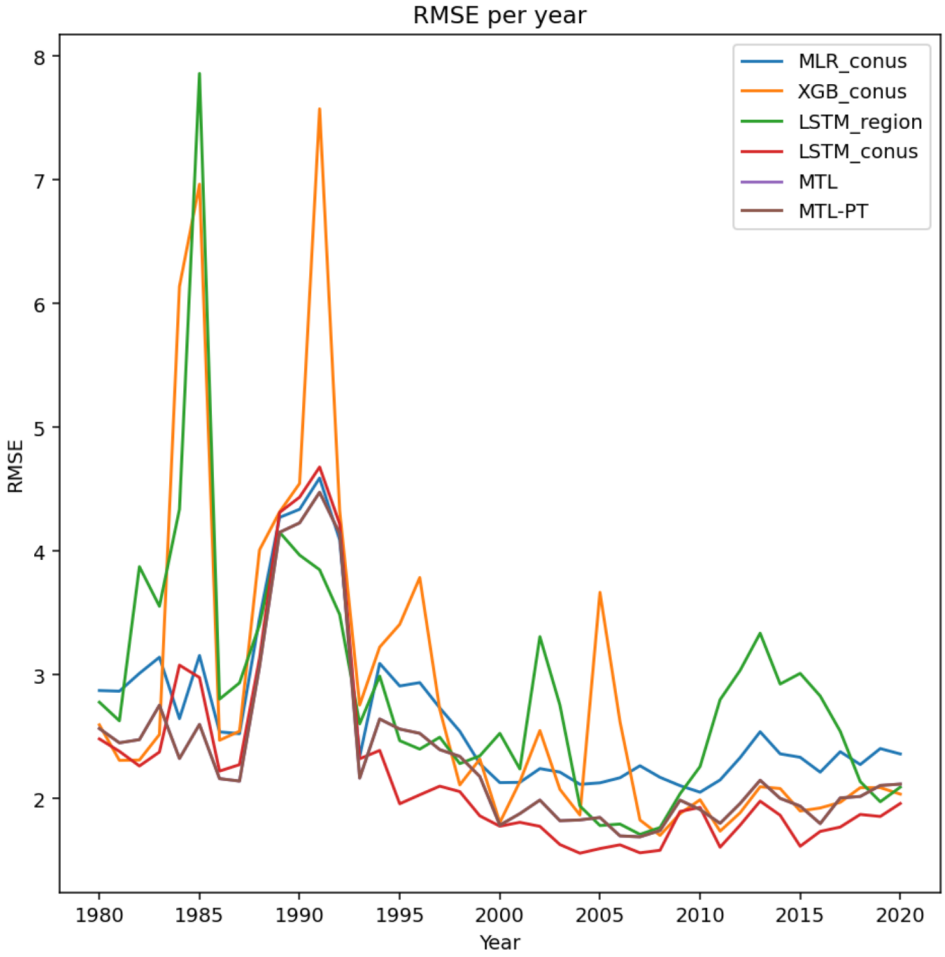


Figure 5.1: Per-year RMSE values for each method

combined discharge, and smaller values between 0.01°C and 0.05°C for vapor pressure, precipitation, and solar radiation. For all LSTM-based models including those used in MTL, the watershed attributes from the GAGES dataset had a combined importance equivalent to zero (less than 0.001°C).

The percentage-wise meta-feature importances for the XGBoost metamodels for *MTL* and *MTL-PT* are shown in Figures A.8 and A.9. For *MTL* we see over 20 features with top five importances for the difference between standard deviation of vapor pressure (15.6%), number of source observations (7.3%), differences in maximum air temperature in the basin (7.0%), difference in mean vapor pressure (6.7%), and mean source

observation temperature (4.8%). For *MTL_PT*, there are only seven meta-features used with the importances being 21.4% for difference in mean snow water equivalent, 15.7% for the difference in standard deviation for the log transform of river discharge, 14.7% for difference in longitude, 14% for difference in raw distance to nearest dam, 12.2% for difference in standard deviation of solar radiation, 11.8% for difference in standard deviation of minimum air temperature, and 10.0% for difference in standard deviation of snow water equivalent.

5.4 Discussion

In this study, we show both meta transfer learning and broad-scale modeling using LSTM can be used to address stream temperature monitoring gaps compared to benchmark models, and are also generalizable to different environmental variables. Even with improvements to sensors and monitoring infrastructure, the majority of streams in the United States, as well as in the world, are unmonitored. This has made it difficult to calibrate process-based models traditionally used due to lack of data and risk of overfitting. Both the continental-scale LSTM built using static and dynamic features and also the meta transfer learning framework using pre-trained source models are able to harness data from many other systems to predict temperature in unmonitored sites. Meta transfer learning in particular is further able to leverage over half a million past transfer learning experiences to better select models.

The results show generally a very similar performance of the top two tested machine-learning models, *LSTM_conus* and *MTL_PT*, with a median test RMSE difference of 0.04°C between models (1.45°C and 1.41°C respectively). In contrast, the models had a significantly improved performance when compared to the two benchmark models and the regional LSTM model, which yielded 2.14°C median per-site RMSE for *MLR_conus*, 1.66°C for *XGB_conus*, and 1.62°C for LSTM_regional. Further investigation could be done to look at when and where it is better to transfer more-specific models using MTL rather than build a continental-scale broad-scale model. Notably, XGBoost and standard LSTM models are much easier to build than MTL and both perform well relative to existing process-based or empirical models, so they may be preferred for simplicity's sake.

For our experiments we chose to prune training and testing data to only use water temperature measurements that were preceded by 200 days of continuous streamflow inputs (our sequence length for LSTM), which eliminated 560,726 water temperature observations from usable data. However, gap-filling ML methods [380] could help increase the coverage of streamflow by using modeled data which would allow the use of many more observations. Simulated streamflow has been shown to improve LSTM stream temperature prediction models in other works [381].

The results of this work call into question the necessity the use of streamflow as an input for ML predictions of stream temperature values which is widely used [381, 22, 382, 20]. Heat advection through upstream, downstream, tributaries, and groundwater flow alongside anthropogenic discharge (e.g. wastewater or cooling water from power plants) are known to affect stream temperature [383, 384, 385, 386]. We see from the feature imporances listed in Figures A.4-A.7 that model performances decrease only by 0.17°C, 0.12°C, 0.13°C, and 0.18°C for *LSTM_conus* , *LSTM_regional*, *MTL*, and *MTL_PT* when a permutation is applied to destroy the streamflow information to the model, a 12%, 7%, 8%, and 13% increase in error respectively. When taking into account the aforementioned exclusion of observations due to lack of streamflow input values, it's possible to exclude streamflow as an input and correspondingly include a large amount of data. This would also allow for much greater geographical coverage to additional watersheds and other areas that have no streamflow data available.

Finally, given the demonstrated generalizability of meta transfer learning and broad-scale entity-aware LSTM models, this approach opens doors to many new research directions. In particular, as experts in the water resources community have called for integration of domain knowledge with data-driven methods [16], both MTL-based and broad-scale entity-aware models have the potential to incorporate process knowledge. Previous MTL work on lake temperature has had success implementing pre-training on process-model output and conservation of energy loss function terms [1], and similar methods could be applied to stream temperature. Other avenues include transferring source models into new spatial domains potentially to different continents, incorporating remote sensing observation data, implementing uncertainty quantification methods like Monte Carlo dropout or Bayesian neural networks [387], and deriving domain knowledge from more involved explainable AI techniques like Shapley additive values [388].

like transferring source models into new spatial domains, including remote sensing surface observation data, incorporating uncertainty quantification, and aggregating models more effectively.

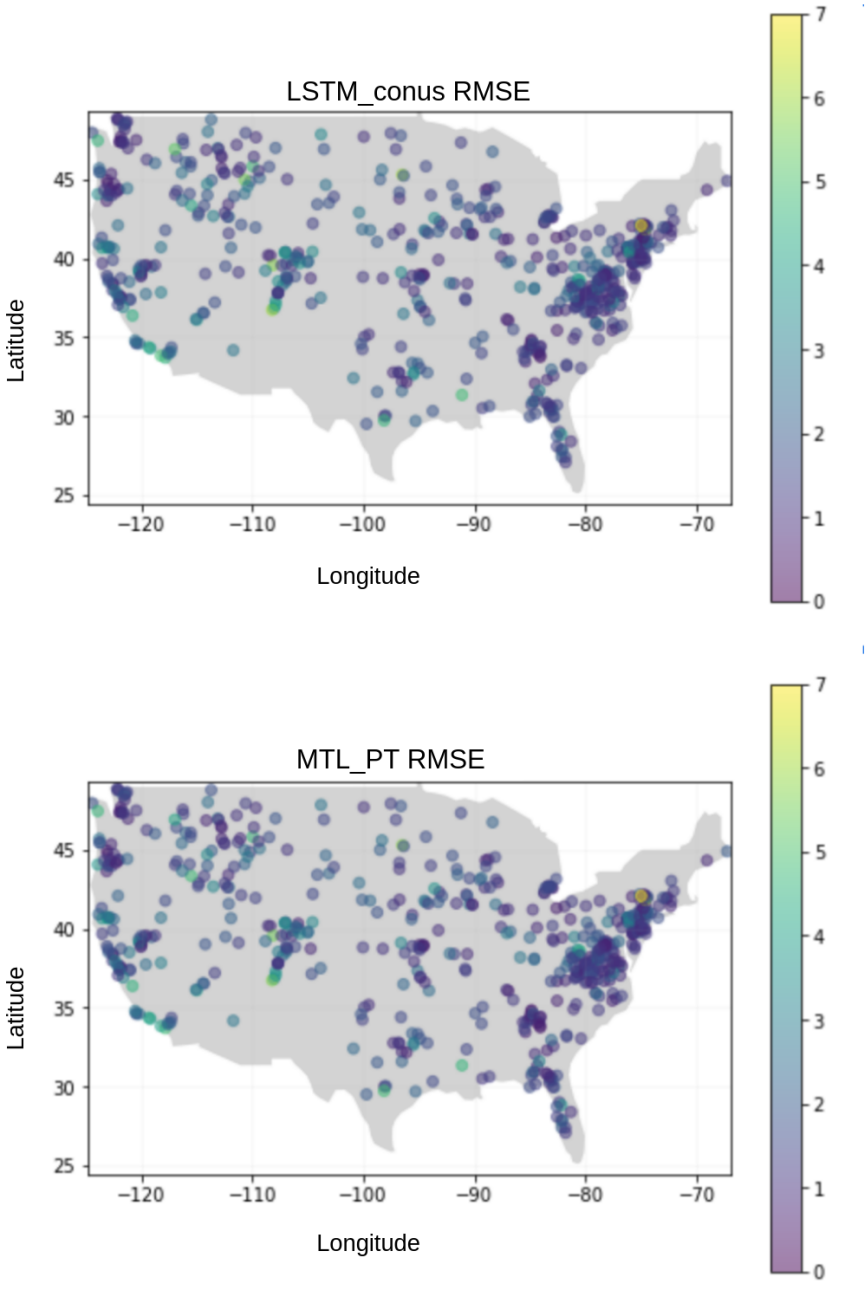


Figure 5.2: Spatial distribution of RMSE values for the continental-scale LSTM model (*LSTM_conus*) model and the meta transfer learning with pre-training (MTL-PT) framework over 580 testing stream locations

Appendix

Appendix A: Data details

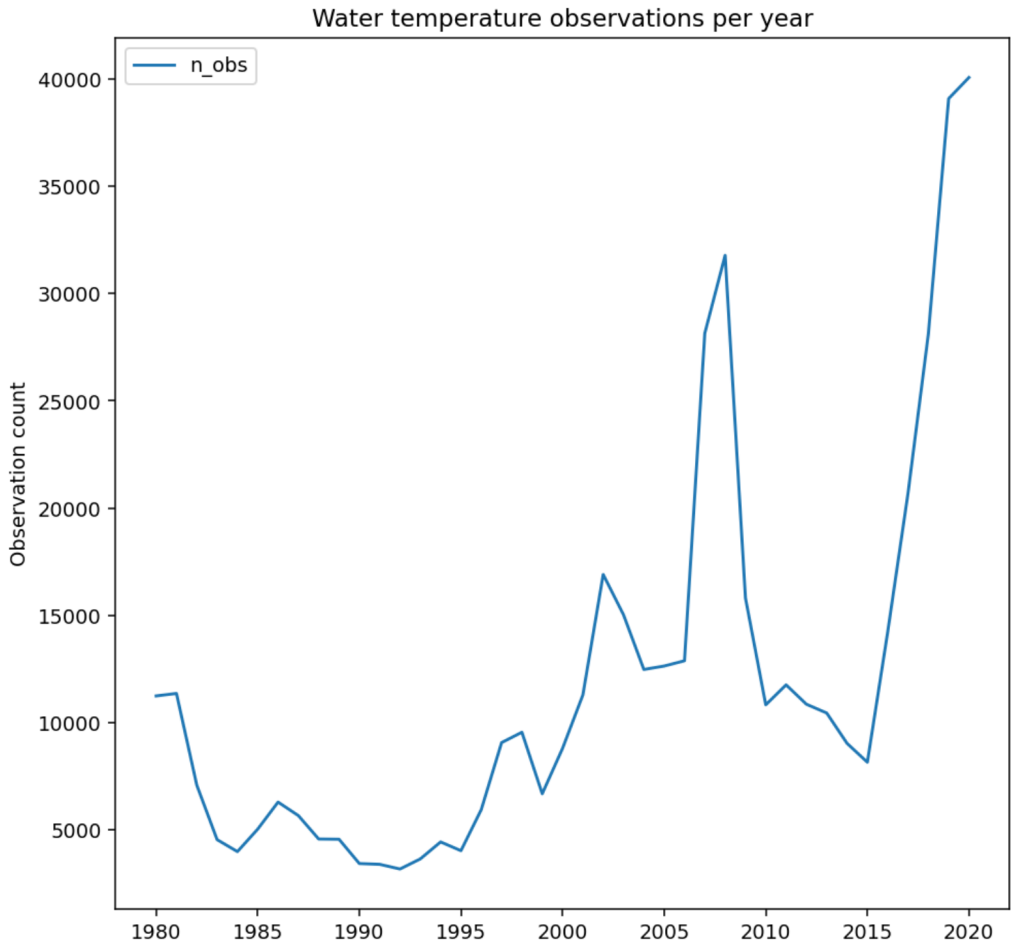


Figure A.1: Observations per year for the 580 test sites

Appendix B: Feature Importances

Feature Abbreviation	Description
alt	Altitude
dayl	Duration of the daylight period in seconds per day.
DRAIN_SQKM	Watershed drainage area
ELEV_MEAN_M_BASIN	Mean watershed elevation
FORESTNLCD06	Watershed percent forest
HIRES_LENTIC_PCT	Percent of watershed area covered by lake/pond/reservoir
lat	Latitude
long	Longitude
NDAMS_2009	Number of dams in watershed
rdc	Daily river discharge
PERDUN	Dunne overland flow as percentage of total streamflow
PLANTNLCD06	Watershed percent agriculture (plant)
prcp	Daily total precipitation in millimeters per day, sum of all forms converted to water-equivalent
PPTAVG_BASIN	Mean annual precipitation for watershed for 1971-2000
RAW_AVG_DIS_ALL_MAJ_DAMS	Raw average straight line distance of gage location to all major dams in watershed
RAW_AVG_DIS_ALLDAMS	Raw average straight line distance of gage location to all dams in watershed
RAW_DIS_NEAREST_DAM	Raw straight line distance of gage location to nearest dam in watershed
RAW_DIS_NEAREST_MAJ_DAM	Raw straight line distance of gage location to nearest major dam in watershed.
RH_BASIN	Watershed average relative humidity
SLOPE_PCT	Mean watershed slope
srad	Daily solar radiation
STOR_NID_2009	Dam storage in watershed per watershed area
STREAMS_KM_SQ_KM	Stream density (stream length per area of watershed)
swe	Snow water equivalent
T_MAX_BASIN	Average monthly maximum temperature from 1971-2000
T_MAXSTD_BASIN	Standard deviation of monthly maximum temperature from 1971-2000
T_MIN_BASIN	Average monthly minimum temperature from 1971-2000
T_MINSTD_BASIN	Standard deviation of monthly minimum temperature from 1971-2000
tmean	Daily mean air temperature
tmax	Daily maximum air temperature
tmin	Daily minimum air temperature
vp	Water vapor pressure in pascals. Daily average partial pressure of water vapor

Table A.1: Feature Name Descriptions. Any feature appended by "*t-xx*" where *xx* is an integer is the time lagged value by *xx* days.

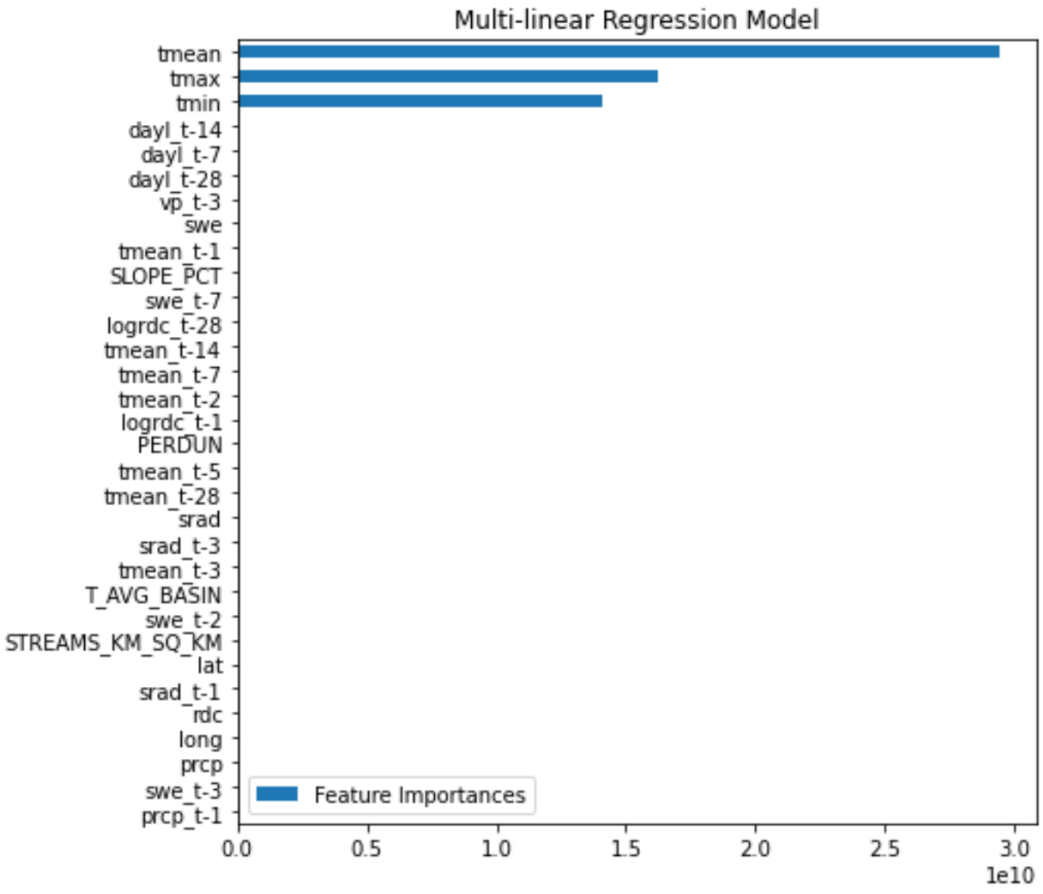


Figure A.2: Selected Features and Importances for the *MLR_conus* model

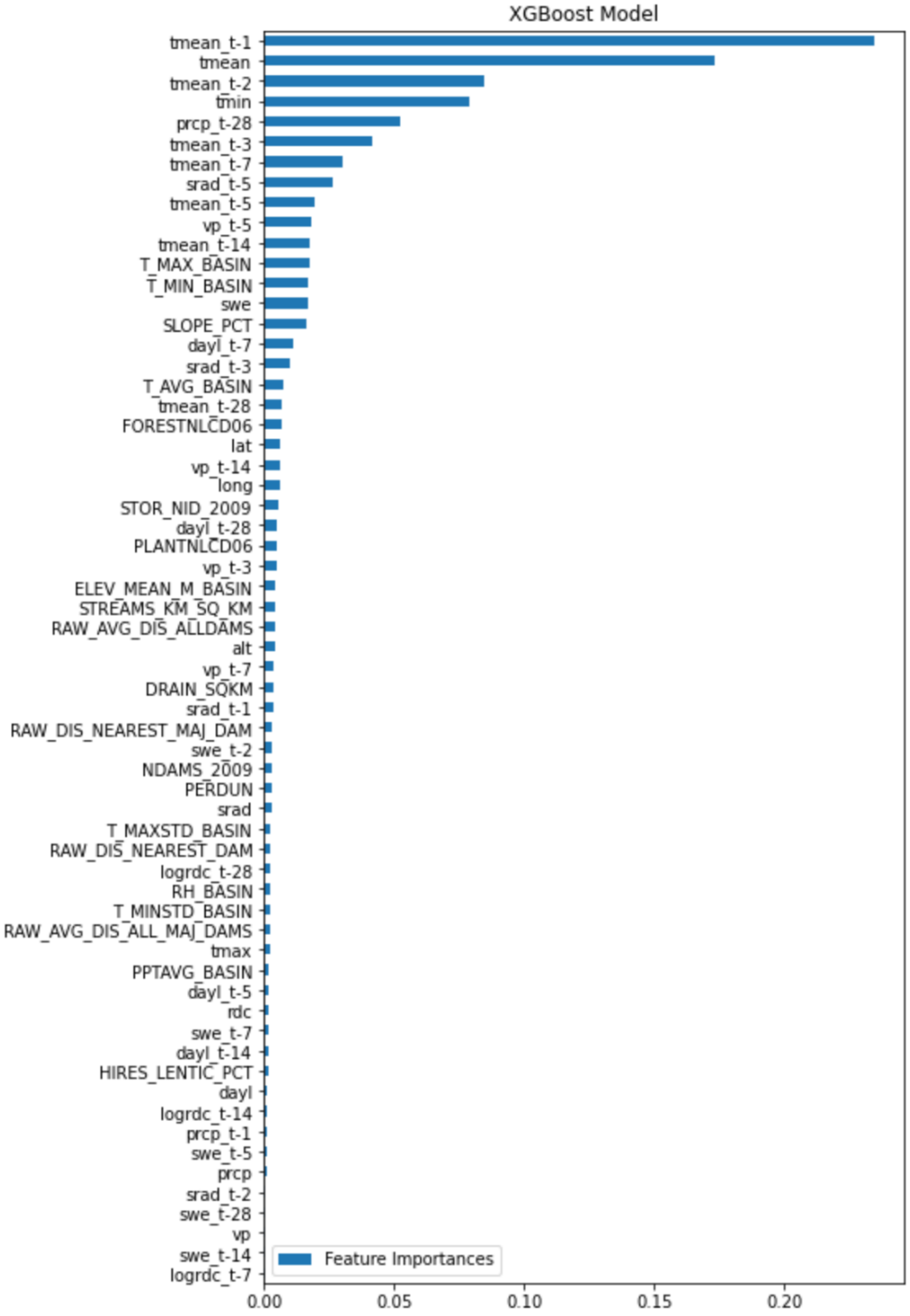


Figure A.3: Selected Features and Importances for the *XGB_conus* model

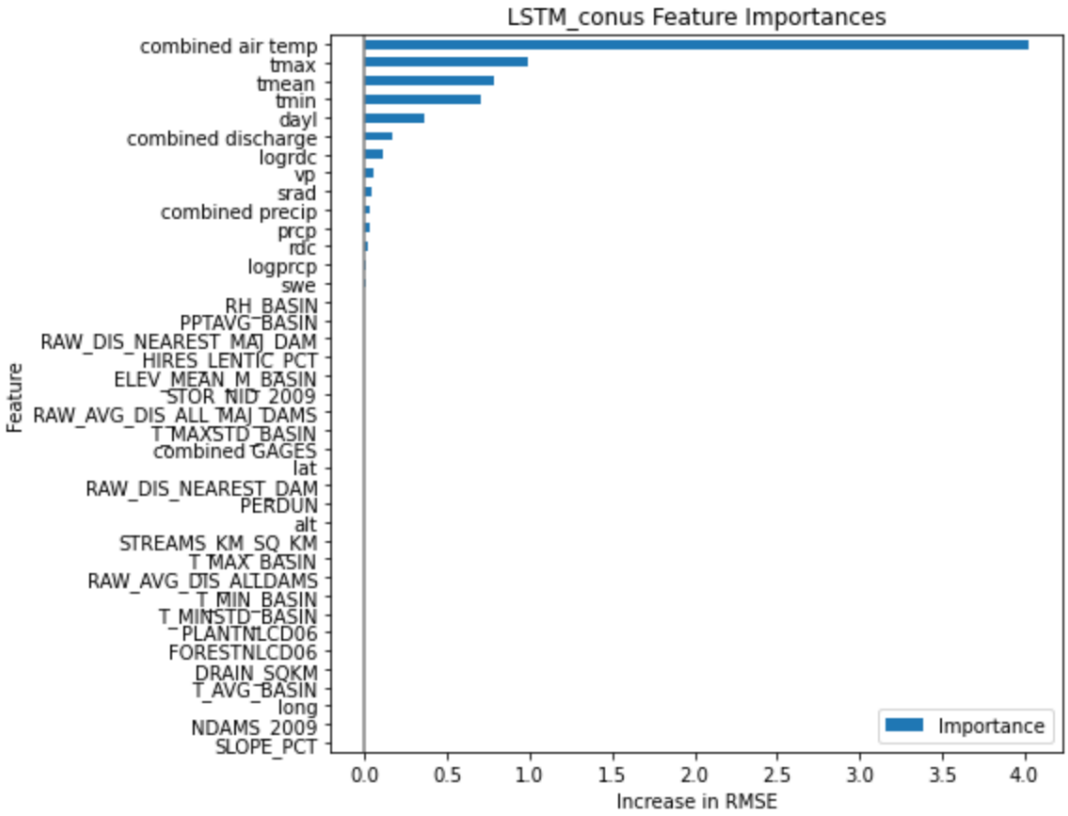


Figure A.4: Selected Features and Importances for the *LSTM_conus* model

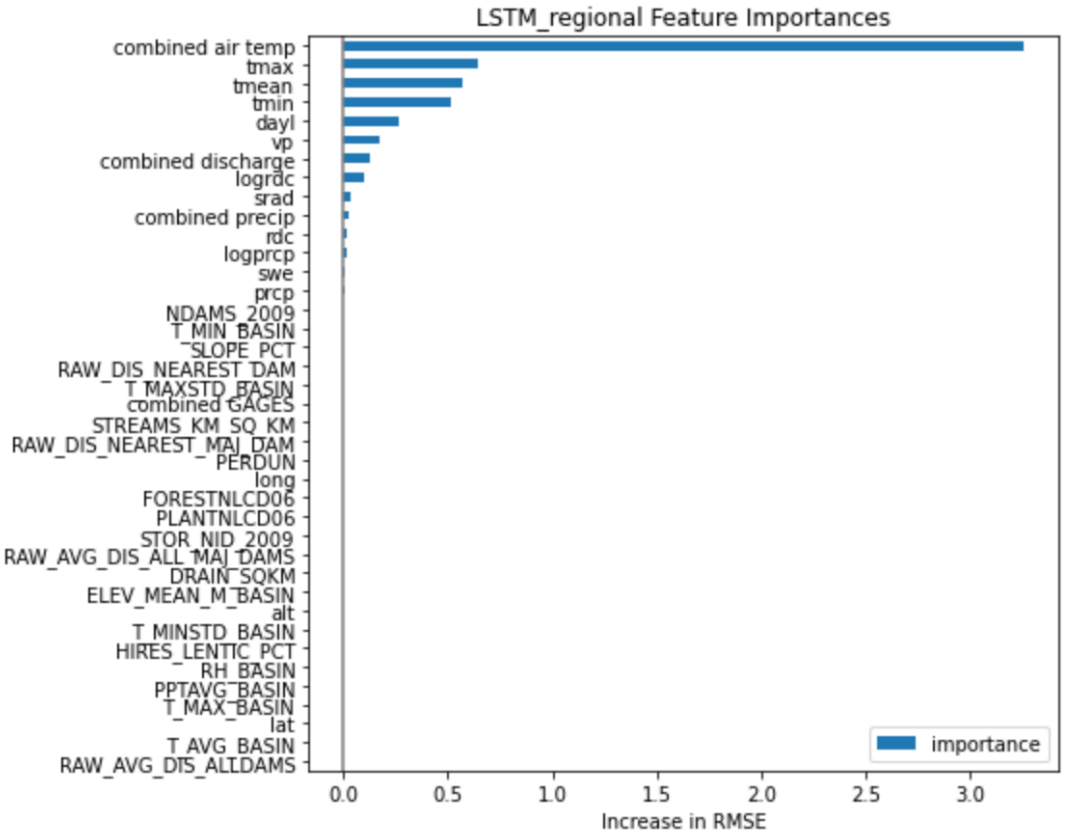


Figure A.5: Selected Features and Importances for the *LSTM_regional* model

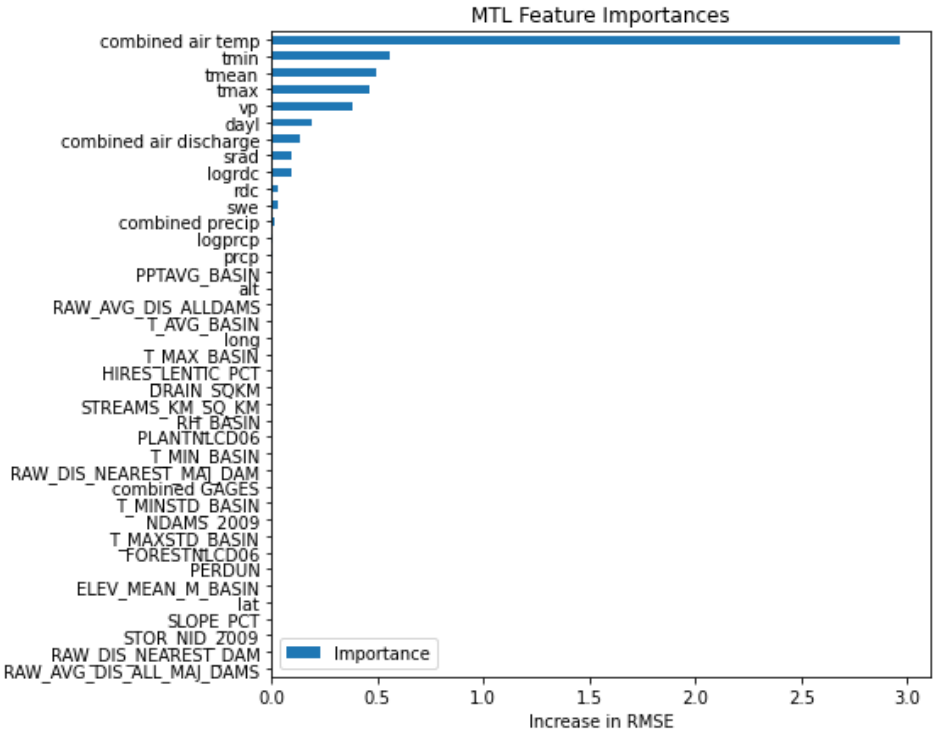


Figure A.6: Feature Importances for the *MTL* model

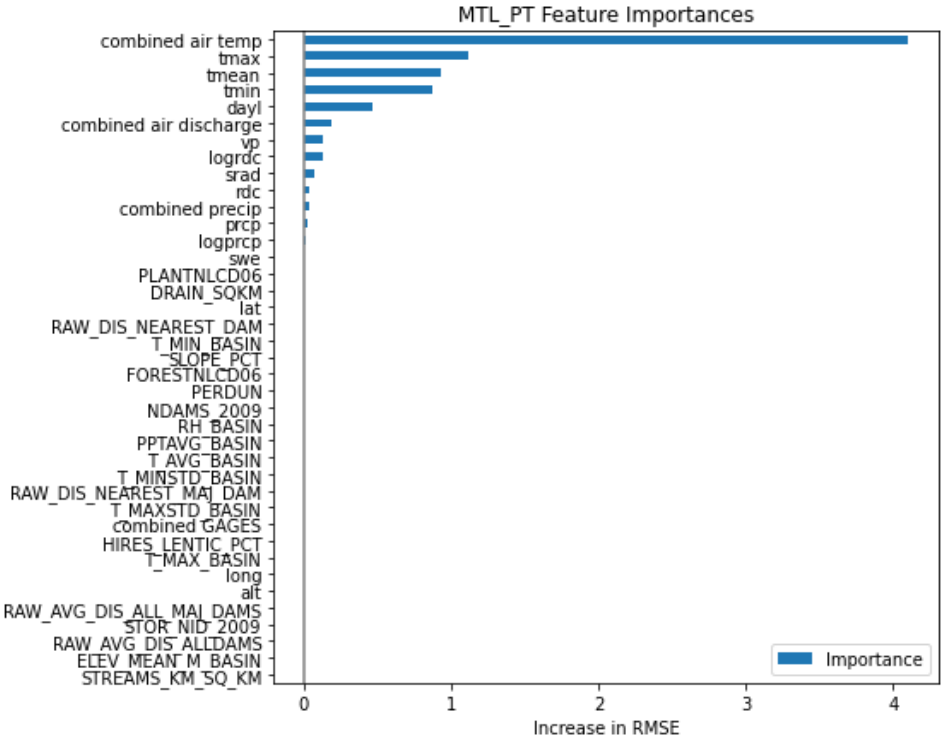


Figure A.7: Feature Importances for the *MTL_PT* model

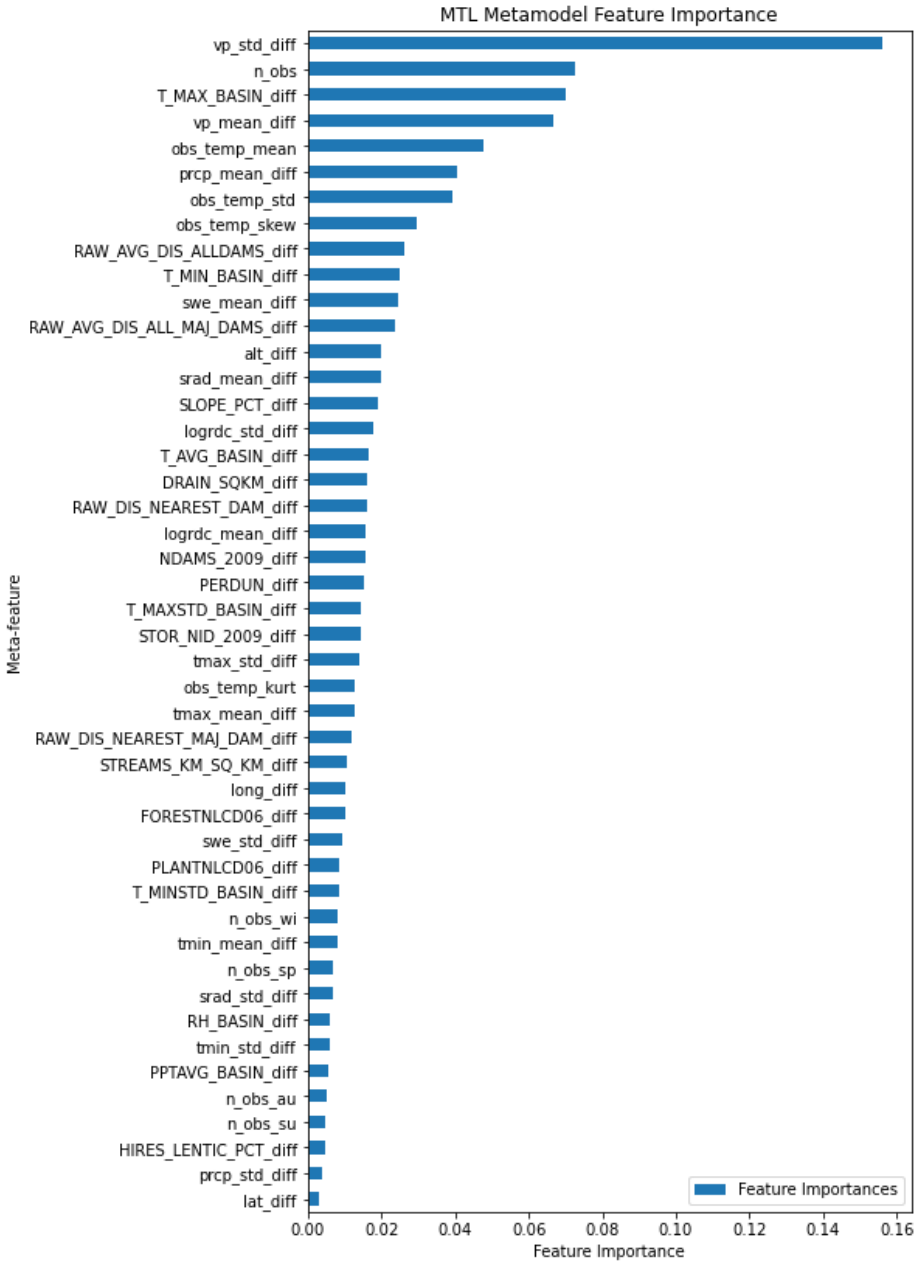


Figure A.8: Meta-feature Importances for the *MTL* metamodel. Features appended by "_diff" represent the difference calculated between the source and target system as target value minus source value.

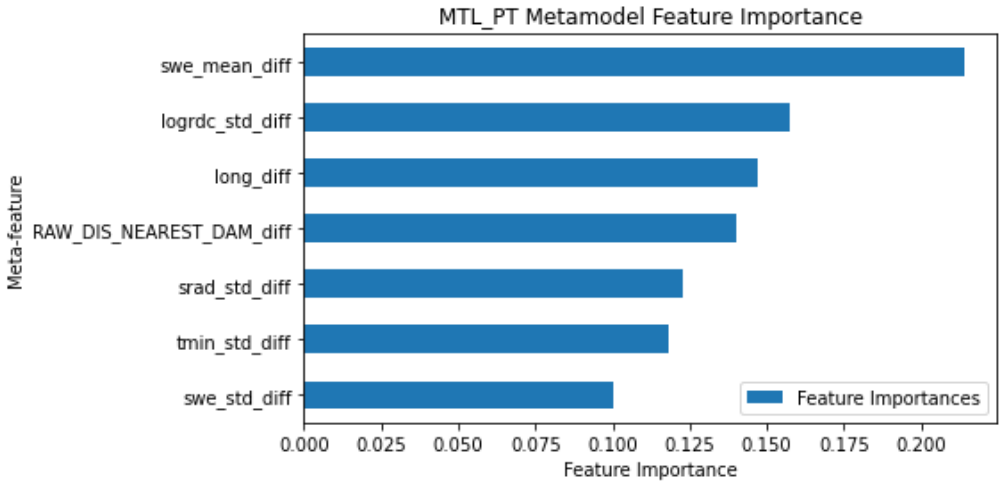


Figure A.9: Meta-feature Importances for the *MTL_PT* metamodel. Features appended by "_diff" represent the difference calculated between the source and target system as target value minus source value.

Chapter 6

Open Questions and Future Directions in Unmonitored Prediction

Though the works reviewed in this thesis encompass many techniques and applications, there are still many open issues to be addressed as the water resources scientific community increasingly adopts ML approaches for unmonitored prediction. Here we highlight questions for further research that are widely applicable and agnostic to any specific target environmental variable, and should be considered as the field moves forward.

6.1 Is more data always better? How do we construct optimal training datasets?

Based on the current state of literature, it is reasonable to assume that ML models, especially deep learning models, benefit from large and comprehensive datasets of heterogeneous entities. This challenges the longstanding notion in traditional process-based and empirical modeling that transferring hydrological models and knowledge from one basin or system to another requires that they must be functionally similar [29, 46]. In Section 2.2.1, we saw many applications in which using *all* available data across heterogeneous sites was the preferred method for training ML models as opposed to fitting to

individual or a subset of sites. Many recent studies continue the traditional practice of developing unsupervised, process-based, and data-driven functional similarity metrics and homogeneity criteria when selecting either specific sites or subgroups of sites to build models on to be transferred to unmonitored sites as we can see from Section 2.2.2. Notably, some of these works show models built on subgroups of sites outperform models using all available sites. Further research is needed to develop robust frameworks to discern how many sites need to be selected for training, what similarity needs to be leveraged to do so, and if excluding sites or regions can benefit broad-scale ML models when given different environmental variable prediction tasks.

When building an unmonitored prediction framework for a given environmental variable, an overarching research question should be, "Which training dataset minimizes the target site error?". Work in streamflow modeling that has explicitly analyzed the effect of merging data from heterogenous entities on prediction performance [112] is a great example demonstrating one step in deciding between using all available data versus a subset of functionally similar entities. This further begs the question of how to optimally select functionally similar entities to construct a training dataset to minimize target site prediction error. Many approaches exist including using a derived unsupervised similarity between sites (e.g. using network science [389]), using metalearning to select training data (e.g. active learning-based data selection [390]), or comparing existing expert-derived metrics like hydrological signatures. There are also methods to combine training for large-scale entity-aware modeling while also specifying a target region or class of similar sites exist (further explained in Section 6.2), and this is another example of where functional similarity could be applied.

Approaches also exist to use ML frameworks like neural networks to develop the similarity encodings themselves, which could be used to select subgroups of sites. [44] demonstrate a custom LSTM architecture that delineates static and dynamic inputs, feeding the former to the LSTM input gate and the latter to the remaining gates. The idea is to use the input gate nodes to encode the functional similarity between stream gauge locations based on the site characteristics alone, and they show this to reveal interpretable hydrological similarity that aligns with existing hydrological knowledge. This framework as-is will not exclude any sites directly, but still offers insight into the usefulness of embedded functional similarity. We also see the static feature encoding

from Section 2.2.1, differing from the previously mentioned method by using a separate ANN for static features as opposed to different gates in the same LSTM. Future research in developing these similarity encodings can also extend into adversarial-based ML methods that could discern valuable training entities.

Numerous other factors can be considered in training dataset construction when deciding whether to include entities other than functional similarity as well. First, the training data should be representative of all types of entities relevant to the prediction tasks, and not too biased towards a particular region or type of site which can correspondingly bias results. When building a model to transfer to a particular set of unmonitored sites, it must be considered whether the training data is representative of those target sites. Environmental monitoring paradigms from the past may be in line with current priorities. Another consideration is the quality of data, where some sites may have higher quality of data than other sites which may have some highly uncertain characteristics. In cases like these, uncertainty quantification methods can be used to increase the reliability of predictions [387], or different weighting can be assigned to different entities based on uncertainty metrics or what the training dataset needs to be representative. It has also been shown that assigning a vector of random values as a surrogate for catchment physical descriptors can be sufficient in certain applications [371].

6.2 How do we include specificity of place when applying broad-scale models?

Unlike traditional process-based and conceptual models, we see from studies like [112] and [179] that deep learning models are most accurate when trained on large-sample, geographically diverse datasets. However, water management stakeholders, decision-makers, and forecasters often seek to prioritize specific individual locations. Many of the broad-scale approaches to unmonitored prediction mentioned in this survey are built without any knowledge of the specific testing sites they are going to be applied to. While training without any knowledge of the testing data is a common practice in supervised machine learning, unmonitored prediction efforts may benefit from including information on specific test sites during training. For example, characteristics from the

test sites are used in the meta transfer learning framework described in Section 2.2.3 to select source models to apply to the target or test system. Surveys on transfer learning [141, 140] have described this distinction as the difference between *inductive* transfer learning, where the goal is to find generalizable rules that apply to completely unseen data, with *transductive* transfer learning, where the input data to the target or test system is known and can be used in the transfer learning framework. Transductive transfer learning methods like meta transfer learning have been proposed, but there is a lack of transductive methods that can harness the power of the highly successful entity-aware broad-scale models.

In the same way that transfer learning has facilitated the pre-training of ML models in hydrology on data-rich watersheds to be transferred and fine tuned efficiently with little data in a new watershed, for example in flood prediction [118], we imagine there could be ways to harness to benefits of large-scale entity-aware modeling and also fine tune those same models to a specific region or class of sites. For example, the entity-aware models using all available data described in Section 2.2.1 could be fine tuned to specific groups like in Section 2.2.2, or the individual source models described in transfer learning approaches in Section 2.2.3 could be pre-trained using all available data.

6.3 How do we address non-stationarity in site characteristics?

As seen throughout this work, static characteristic inputs to ML models can be used in multiple ways to develop site (e.g. basin) similarity mappings. For example, the CAMELS dataset includes static characteristics for each basin's topography, climate, streamflow, land cover, soil, and geology, and the [391] lake temperature data release contains lake characteristics like bathymetry, surface area, stratification indices, and water clarity estimates. Though this treatment of certain static characteristics as static is intuitive for values like location and geology, many of these characteristics like water quality, quantity, land cover/use, or climate are realistically dynamic in nature. This can affect prediction performance in cases where the dynamic nature of certain characteristics treated as static are vital to prediction. For example, land use is a key dynamic predictor for river water quality in areas undergoing urbanization [392], but

is treated as static in most hydrological datasets including CAMELS and the GAGES (Geospatial Attributes of Gages for Evaluating Streamflow [354]). In lake temperature modeling, water clarity is treated as static in [1] but realistically has a notable dynamic effect on water column temperatures [239]. Though this problem exists in both monitored and unmonitored prediction scenarios, characteristics are particularly important in unmonitored prediction since often that is the only knowledge available concerning a location.

Multiple ML methods can be used to deal with a single variable that may be missing its dynamic nature that take advantage of other dynamic inputs or simulation data. One approach is to predict the value as an intermediate quantity in a deep learning model using a custom architecture [180, 20], where the intermediate variable can be inferred from other dynamic inputs or simulation data. Methods like inverse modeling that have been applied successfully in the monitored prediction scenario in hydrology [393], could also be used in unmonitored scenarios. Since inverse modeling requires many target variable observations, simulation data would need to be substituted for this to be feasible. However, this is difficult as process-based models are often dependent on the same characteristics for generating output.

As data collection from environmental sensors continues to improve, new methods to deal with the dynamic nature of certain characteristics that can be increasingly captured (e.g. remotely sensed land cover or water clarity) must be developed. For example, the static nature of stream site similarities created for graph neural networks in works like [81] is challenged since the static graph would become dynamic, and the dynamic nature of certain meta-features treated as static for the meta transfer learning in [1] would require alterations in the metamodeling.

Additionally, these changes in characteristics often operate at different time scales than dynamic input drivers like daily meteorological data. For instance, climate or land use may change over the scale of years rather than days. Though temporally multi-scale modeling has been explored extensively within the discipline for process-based modeling [394, 395], statistical modeling [396], and wavelet-based modeling (optionally coupled with classical ML) [397, 398, 399, 400]; this has not yet been done for the deep learning approaches that are increasingly being used. However, these advances in deep learning have been seen in other disciplines where several neural network architectures have been

proposed for multi-scale LSTM models in web traffic prediction [401], natural language processing [402], and fault diagnostics for manufacturing [403].

6.4 Can we use auxiliary data to improve modeling?

Many real-world situations arise where, despite unavailable daily target data, researchers can access auxiliary information relevant to physical processes. For instance, when streamflow is unavailable, auxiliary data (also called soft data) like streamflow distributions (e.g. flow duration curve), soil moisture data, or values concerning river biodiversity and ecosystem traits may be available. Auxiliary data differs from site characteristics in that it is often partially available, derived from a process-based model, or contains a large degree of uncertainty. Though auxiliary data has historically been used to constrain process-based models [404, 405], it has not seen much use in ML applications for unmonitored prediction. However, many globally-available auxiliary data sources exist within both satellite products (e.g. Soil Moisture Active Passive (SMAP) mission [406]), or derived/simulated products (e.g. WaterGAP for groundwater depletion [407]). The use of an encoder neural network for auxiliary data described in Section 2.2.1 is just one way of utilizing auxiliary data. Further research could integrate this auxiliary data, which may only be partially available, into ML to provide additional relevant context and data.

Addressing and quantifying uncertainty is also necessary for using auxiliary data products, as many satellite products deal with noise and inconsistency and many derived products from process-based models experience a range of accuracy. Various uncertainty quantification methods in ML [387] such as Monte Carlo dropout, Bayesian neural networks, or variational autoencoders could be used to deal with these issues by assessing the difference in prediction uncertainty before and after adding in auxiliary or derived data.

Auxiliary data can also vary drastically in what is available in a given water resources entity or system. For example, certain watersheds of interest like the U.S. Department of Energy's Watershed Function Scientific Focus Area [408] include a wealth of heterogeneous data including hydrological, genomic, biogeochemical, climate, vegetation, geological, and remote sensing data. Similarly for lakes, efforts like the Long Term

Ecological Research Network [409] includes additional information regarding biophysical setting, changing land use and land cover, and position in groundwater flow, among many other variables not found in most lakes. New ML frameworks that can leverage relevant partially available auxiliary data like these when available are required to augment existing broad-scale modeling in many environmental variable prediction scenarios.

6.5 How can we leverage process understanding for unmonitored prediction?

The success of ML models achieving better prediction accuracy across many hydrological and water resources variables compared to process-based models has led to the question posed by [167] of, "What role will hydrological science play in the age of machine learning?". Given the relevant works reviewed in this study showing mixed results comparing KGML approaches using process understanding with domain-agnostic black box approaches, more research is required to address the role of domain knowledge in unmonitored prediction. From Section 2.2.4 we see that using graph neural networks has potential to encode spatial context relevant for unmonitored prediction and improving over existing methods, but also that hybrid models have not been as effective as domain-agnostic entity-aware LSTM counterparts. A key research direction will be finding which context is relevant to encode in graphs or other similarity or distance-based structures, whether that be spatial or based on expert domain knowledge. A preferable alternative to existing hybrid process-ML models may be the DPB models explained in Section 2.2.4, which exhibit many side benefits like being able to output accurate intermediate variables and demonstrating interpretability, but the performance achieved remains similar to existing process-agnostic models like the entity-aware LSTM models. There is potential to further research and develop these DPB approaches, for instance they stand to benefit from assimilating multiple data sources since they simulate numerous additional variables.

There are also many applications of other types of KGML modeling techniques, like informed loss functions, informed model architecture, and multi-task learning which have not translated from monitored to unmonitored prediction scenarios. However,

these techniques are still largely applicable in the absence of target data. Each of these techniques offers an opportunity for hydrological science to play a role in ML, and each comes with possible benefits in hydrology such as improved prediction performance, efficiency, and interpretability (see [150] for a full survey); therefore they should be strongly considered during method development. For example, knowledge-guided loss function terms can impose structure on the solution search space in the absence of labeled target data by forcing model output to conform to physical laws (e.g. conservation of energy or mass). Examples of successful implementations of knowledge-guided loss functions to improve prediction in gauged scenarios include the conservation of energy-based term to predict lake temperature [17], power-scaling law-based term to predict lake phosphorous concentration [160], and advection–dispersion equation-based terms to predict subsurface transport states [410]. These results show that informed loss functions can improve physical realism of the predictions, reduce the data required for good prediction performance, and also improve generalization to out-of-sample scenarios. Since loss function terms are generally calculated on the model output and do not require target variable data, they can easily be transferred from monitored prediction to unmonitored.

Knowledge-guided architecture can similarly make use of the domain-specific characteristics of the problem being solved to improve prediction and impose constraints on model prediction. As opposed to soft constraints as imposed by a loss function term, architectural modifications can impose hard constraints. Successful examples of modified neural network architectures for hydrological prediction include a modified LSTM with monotonicity constraints for lake temperatures at different depths [162], mass-conserving modified LSTMs for streamflow prediction [155], and an LSTM architecture that includes auxiliary intermediate processes that connect weather drivers to streamflow [180]. Many hydrological prediction tasks involve governing equations such as conservation laws or equations of state that could be leveraged in similar ways to improve ML performance in unmonitored sites.

Domain knowledge in the form of auxiliary tasks can also inform prediction through the multi-task learning [411]. Multi-task learning allows for two or more learning tasks to be solved simultaneously, ideally while exploiting commonalities and differences across tasks. This can improve learning efficiency and predictions for one or more of the tasks. An example of an auxiliary task in a multi-task framework might be related

to ensuring physically consistent solutions in addition to accurate predictions. For example, pairing streamflow and water temperature prediction tasks has been shown to improve streamflow prediction but not water temperature in a monitored prediction scenario [412]. This type of approach could be applied to an prediction scenario where one variable (e.g. water temperature) is unmonitored and another auxiliary variable (e.g. streamflow) is monitored. Further research is needed to determine what auxiliary variables are useful in a multi-task learning setting and if these methods extend to unmonitored prediction.

6.6 How can we leverage model ensembles for unmonitored prediction?

Using ensembles of models for prediction is a longstanding technique in hydrology that spans both process-based models [413, 414] and more recently ML models [61]. Ensemble learning is a general meta approach to model building that combines the predictions from multiple models for better predictive performance. In traditional water resources prediction, ideally, models in the ensemble will differ with respect to either meteorological input dataset (e.g. [415]), process-based model parameters (e.g. [416]) or multiple process-based model structures (e.g. [417]). Different types of techniques are seen across ensemble learning more generally in the ML community, with common techniques such as (1) bagging, where many models are fit on different samples of the same dataset and averaging the predictions, (2) stacking, where different models types are fit on the same data and a separate model is used to learn how to combine the predictions, and (3) boosting, where ensemble members are added sequentially to correct the predictions made by previous models. Some of the main advantages of model ensembles in both cases is that the uncertainty in the predictions can be easily estimated and predictions can become more robust, leading them to be ubiquitous within many forecasting disciplines. Diversity in models is key, as model skill generally improves more from model diversity rather than from a larger ensemble [418].

There are key differences in ensemble techniques in process-based modeling versus ML. For instance, expert-calibrated parameters have very specific meanings in process-based models whereas the analogous parameters in ML (usually known as weights) are

more abstract and characteristic of a black box. When tweaking parameters between models to assemble an ensemble, physical realism is important in the process-based model case. Parameterization has a rich history in process-based models and the work can be very domain-specific, whereas ML ensemble techniques are often done using existing code libraries through a domain-agnostic process. Furthermore, ML ensemble techniques usually do not modify input datasets, though they could through adding noise [419]. However, this still differs from using different meteorological data products for process-based models where the differences are more structured and diversity between products is more apparent.

We see most ML applications reviewed in this work do not attempt to use ensemble techniques even though the few that do, see positive results (e.g. [14] for stream temperature, [67] for streamflow, [111] for water level). A recent survey by [61] finds that ensemble ML strategies demonstrate "absolute superiority" compared regular (individual) ML model learning in hydrology, and this result has also been seen in the machine learning community more generally for neural networks [420]. Many opportunities exist to develop ensemble frameworks in water resources prediction that harness numerous diverse ML models. In the same way that the hydrology community often uses ensembles of different process-based model structures, the many different architectures and hyperparameters in deep learning networks can achieve a similar diversity. Given the common entity-aware broad-scale modeling approach seen widely throughout this review, opportunity exists to use resampling techniques like bootstrap aggregation [421] to vary training data while maintaining broad coverage, as seen in [14] for stream temperature. Other ensemble methods like in [67] vary which site characteristics are used as inputs to LSTMs for streamflow prediction.

6.7 Can we use existing explainable AI techniques to derive domain knowledge?

Historically, the difference between ML methods and more process-based or mechanistic methods has been described as a tradeoff between "predictive performance" and "explainability" [422]. However, there has been a deluge of advances in recent years in the field of explainable AI (XAI) [423] and applications of these are increasingly being

seen in geosciences [424, 425]. For example, recent work has shown how XAI can help to calibrate model trust and provide meaningful post-hoc interpretations [426], identify how to fine-tune poor performing models [427], and also accelerate scientific discovery [428]. This has led to a change in the narrative of the performance and explainability tradeoff as calls are increasingly made for the water resources community to adopt ML as a complementary or primary avenue toward scientific discovery [429]. Though the majority of work using XAI in water resources time series prediction has been seen in the monitored prediction scenario (e.g. [19, 430]), opportunity exists to analyze how ML models are able to transfer hydrologic understanding between sites in the unmonitored prediction scenario to help address one of the most fundamental problems in hydrology.

We see that many water resources researchers still choose the more interpretable classical ML models like random forest or XGBoost due to their ease of interpretability, and initial investigations of interpretability of deep learning frameworks listed in this work have mostly addressed simple questions like feature attribution and sensitivity (e.g. [81, 191]). The concept of DPB models discussed in Section 2.2.4 shows potential to take this further and make an end-to-end interpretable model mimicking environmental processes but with the trainability and flexibility of deep neural networks. DPB models can provide more extensive interpretability compared to simpler feature attribution methods by being able to represent intermediate process variables explicitly in the neural network with the capability of extracting their relationship to the inputs and outputs.

Future work on XAI for unmonitored prediction can pose research questions in directions that harness the existing highly successful ML models to both refine theoretical underpinnings and add to the current hydrologic or other process understandings surrounding regionalizations to unmonitored sites. For example, methods like layerwise relevance propogation, integrated gradients, or Shapley additive explanations (SHAP) [388] could be used to explore causations and attributions of observed variability in situations where ML predicts more accurately than existing process-based regionalization approaches. Both temporal and spatial attributes can be considered, for example when using methods like SHAP with LSTM the attributions of any inputs along the sequence length can be used to see how far back in time the LSTM is using its memory to perform predictions, or in GNNs to see where in space the knowledge is being drawn for prediction [431].

6.8 Can existing process knowledge help to build better explainable AI techniques?

Simple XAI methods like feature attribution can be difficult to implement on complex architectures like LSTM and other deep learning frameworks since they do not honor many desired mathematical properties like "sensitivity" or "implementation invariance" [432, 433], and they also face many nontrivial issues for individual problem setups including susceptibility to manipulation [434] and high capacity for human error [435]. Also, recent XAI comparison studies within geosciences have shown that the robustness and comprehensibility of feature attributions depends strongly on the prediction setting and ML model architecture, and that likely no universally optimal method exists [428, 436]. Correspondingly, calls have been made for domain expertise as a necessity in defining the meaning of interpretability for the given domain and the features for machine learning [435].

One way to implement domain expertise for the development of XAI methods is to construct benchmark scenarios and datasets. The validity of existing XAI methods is built upon a large amount of successes on benchmark applications where the process is well-understood and the attributions of each input feature is known *a priori*, but there is currently a lack of benchmark datasets in many geoscientific applications including hydrology [436]. In order to build confidence in XAI methods applied to predictions in unmonitored systems, the hydrology community must construct similar benchmark datasets. The rich history of process-based models in hydrology potentially offers an abundence of synthetic datasets where the processes and governing equations are explicitly known and understood, but there is a lack of work like this comparing different XAI approaches.

Domain science can also inform XAI in the same way that KGML techniques mentioned in Section 2.2.4 can inform prediction models. For example, using KGML techniques ML frameworks can include constraints to make sure they do not violate laws of physics in their output, and it is also important that XAI methods for hydrology do not yield explanations or relationships between variables that violate laws of physics. This can potentially yield better trust in models by ensuring the models cannot predict scenarios which are impossible and cannot learn physically inconsistent relationships for

prediction.

Chapter 7

Conclusion

The use of ML for unmonitored environmental variable prediction is an important research topic in hydrology and water resources engineering, especially given the urgent need to monitor the effects of climate change and urbanization on our natural and man-made water systems. In this thesis, we reviewed the latest methodological advances in ML for unmonitored prediction using entity-aware deep learning models, transfer learning, and knowledge-guided ML models. We summarized the patterns and extent of these different approaches, applied methods in and performed analysis of three real-world prediction scenarios in lake temperature and stream temperature, and enumerated questions and directions for future research. Addressing these questions sufficiently will likely require the training of interdisciplinary water resources ML scientists and also the fostering of interdisciplinary collaborations between ML and domain scientists. As the field of ML for environmental science and water resources progresses, we see many of these open questions can also augment domain science understanding in addition to improving prediction performance and advancing ML science. We hope this thesis can provide researchers with the state-of-the-art knowledge of ML for unmonitored prediction, offer opportunity for cross-fertilization between ML practitioners and domain scientists, and provide guidelines for the future.

References

[1] Jared D Willard, Jordan S Read, Alison P Appling, Samantha K Oliver, Xiaowei Jia, and Vipin Kumar. Predicting Water Temperature Dynamics of Unmonitored Lakes With Meta-Transfer Learning. *Water Resources Research*, 57(7):e2021WR029579, 2021.

[2] Joaquín Muñoz-Sabater, Emanuel Dutra, Anna Agustí-Panareda, Clément Albergel, Gabriele Arduini, Gianpaolo Balsamo, Souhail Boussetta, Margarita Choulga, Shaun Harrigan, Hans Hersbach, and others. ERA5-Land: A state-of-the-art global reanalysis dataset for land applications. *Earth System Science Data*, 13(9):4349–4383, 2021.

[3] Roger W. Bachmann, Sapna Sharma, Daniel E. Canfield, and Vincent Lecours. The Distribution and Prediction of Summer Near-Surface Water Temperatures in Lakes of the Coterminous United States and Southern Canada. *Geosciences*, 9(7):296, July 2019.

[4] J.D. Willard, J.S.. Read, S. Topp, G.J.A. Hansen, and V. Kumar. Daily surface temperature predictions for 185,549 U.S. lakes with associated observations and meteorological conditions (1980-2020): U.S. Geological Survey data release. *U.S. Geological Survey*, 2022.

[5] John H Porter, Paul C Hanson, and Chau-Chin Lin. Staying afloat in the sensor data deluge. *Trends in ecology & evolution*, 27(2):121–129, 2012.

[6] Markus Reichstein, Gustau Camps-Valls, Bjorn Stevens, Martin Jung, Joachim Denzler, Nuno Carvalhais, and others. Deep learning and process understanding for data-driven Earth system science. *Nature*, 566(7743):195–204, 2019.

[7] Lynne Caughlan and Karen L Oakley. Cost considerations for long-term ecological monitoring. *Ecological indicators*, 1(2):123–134, 2001.

[8] Günter Blöschl, Gunter Bloschl, Murugesu Sivapalan, Thorsten Wagener, Hubert Savenije, and Alberto Viglione. *Runoff prediction in ungauged basins: synthesis across processes, places and scales*. Cambridge University Press, 2013.

[9] JL Salinas, G Laaha, M Rogger, J Parajka, A Viglione, M Sivapalan, and G Blöschl. Comparative assessment of predictions in ungauged basins–part 2: Flood and low flow studies. *Hydrology and Earth System Sciences*, 17(7):2637–2652, 2013.

[10] Alejandro Sánchez-Gómez, Silvia Martínez-Pérez, Leduc Sylvain, Antonio Sastre-Merlín, and Eugenio Molina-Navarro. Streamflow components and climate change: Lessons learnt and energy implications after hydrological modeling experiences in catchments with a mediterranean climate. *Energy Reports*, 9:277–291, 2023.

[11] Yuhan Guo, Yongqiang Zhang, Lu Zhang, and Zhonggen Wang. Regionalization of hydrological modeling for predicting streamflow in ungauged catchments: A comprehensive review. *WIREs Water*, October 2020.

[12] Ashok Mishra, Tue Vu, Anoop Valiya Veettil, and Dara Entekhabi. Drought monitoring with soil moisture active passive (SMAP) measurements. *Journal of Hydrology*, 552:620–632, 2017.

[13] Rana Muhammad Adnan, Andrea Petroselli, Salim Heddam, Celso Augusto Guimarães Santos, and Ozgur Kisi. Comparison of different methodologies for rainfall–runoff modeling: machine learning vs conceptual approach. *Natural Hazards*, 105(3):2987–3011, 2021.

[14] Helen Weierbach, Aranildo R Lima, Jared D Willard, Valerie C Hendrix, Danielle S Christianson, Michaelle Lubich, and Charuleka Varadharajan. Stream temperature predictions for river basin management in the pacific northwest and mid-atlantic regions using machine learning. *Water*, 14(7):1032, 2022.

[15] Gary M Lovett, Douglas A Burns, Charles T Driscoll, Jennifer C Jenkins, Myron J Mitchell, Lindsey Rustad, James B Shanley, Gene E Likens, and Richard Haeuber.

Who needs environmental monitoring? *Frontiers in Ecology and the Environment*, 5(5):253–260, 2007.

[16] Chaopeng Shen. A transdisciplinary review of deep learning research and its relevance for water resources scientists. *Water Resources Research*, 54(11):8558–8593, 2018.

[17] Jordan S Read, Xiaowei Jia, Jared D Willard, Alison P Appling, Jacob A Zwart, Samantha K Oliver, Anuj Karpatne, Gretchen JA Hansen, Paul C Hanson, William Watkins, and others. Process-guided deep learning predictions of lake water temperature. *Water Resources Research*, 55(11):9173–9190, 2019.

[18] Abdulhalık Oğuz and Ömer Faruk Ertuğrul. A survey on applications of machine learning algorithms in water quality assessment and water supply and management. *Water Supply*, 2023.

[19] Thomas Lees, Steven Reece, Frederik Kratzert, Daniel Klotz, Martin Gauch, Jens De Bruijn, Reetik Kumar Sahu, Peter Greve, Louise Slater, and Simon J Dadson. Hydrological concept formation inside long short-term memory (lstm) networks. *Hydrology and Earth System Sciences*, 26(12):3079–3101, 2022.

[20] Xiaowei Jia, Jacob Zwart, Jeffery Sadler, Alison Appling, Samantha Oliver, Steven Markstrom, Jared D Willard, Shaoming Xu, Michael Steinbach, Jordan Read, and Vipin Kumar. Physics-Guided Recurrent Graph Model for Predicting Flow and Temperature in River Networks. *Proceedings of the 2021 SIAM International Conference on Data Mining (SDM)*, page 9, 2021.

[21] Hao Jing, Xin He, Yong Tian, Michele Lancia, Guoliang Cao, Alessandro Crivellari, Zhilin Guo, and Chunmiao Zheng. Comparison and interpretation of data-driven models for simulating site-specific human-impacted groundwater dynamics in the north china plain. *Journal of Hydrology*, page 128751, 2022.

[22] Farshid Rahmani, Chaopeng Shen, Samantha Oliver, Kathryn Lawson, and Alison Appling. Deep learning approaches for improving prediction of daily stream temperature in data-scarce, unmonitored, and dammed basins. *Hydrological Processes*, 35(11):e14400, 2021.

[23] U.S. Geological Survey. USGS Water Data for the Nation, 1994.

[24] Emily K. Read, Lindsay Carr, Laura De Cicco, Hilary A. Dugan, Paul C. Hanson, Julia A. Hart, James Kreft, Jordan S. Read, and Luke A. Winslow. Water quality data for national-scale aquatic research: The Water Quality Portal. *Water Resources Research*, 53(2):1735–1745, February 2017.

[25] Kevin T Smiley, Ilan Noy, Michael F Wehner, Dave Frame, Christopher C Sampson, and Oliver EJ Wing. Social inequalities in climate change-attributed impacts of hurricane harvey. *Nature communications*, 13(1):3418, 2022.

[26] W Neil Adger, Jon Barnett, Katrina Brown, Nadine Marshall, and Karen O'brien. Cultural dimensions of climate change impacts and adaptation. *Nature climate change*, 3(2):112–117, 2013.

[27] Brett F Sanders, Jochen E Schubert, Daniel T Kahl, Katharine J Mach, David Brady, Amir AghaKouchak, Fonna Forman, Richard A Matthew, Nicola Ulibarri, and Steven J Davis. Large and inequitable flood risks in los angeles, california. *Nature sustainability*, 6(1):47–57, 2023.

[28] Jan Seibert. Regionalisation of parameters for a conceptual rainfall-runoff model. *Agricultural and forest meteorology*, 98:279–293, 1999.

[29] Tara Razavi and Paulin Coulibaly. Streamflow prediction in ungauged basins: review of regionalization methods. *Journal of hydrologic engineering*, 18(8):958–975, 2013.

[30] Xue Yang, Jan Magnusson, and Chong-Yu Xu. Transferability of regionalization methods under changing climate. *Journal of Hydrology*, 568:67–81, 2019.

[31] Thorsten Wagener and Howard S Wheater. Parameter estimation and regionalization for continuous rainfall-runoff models including uncertainty. *Journal of hydrology*, 320(1-2):132–154, 2006.

[32] Satish Bastola, Hiroshi Ishidaira, and Kuniyoshi Takeuchi. Regionalisation of hydrological model parameters under parameter uncertainty: A case study involving

topmodel and basins across the globe. *Journal of Hydrology*, 357(3-4):188–206, 2008.

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

[34] Ludovic Oudin, Vazken Andreassian, Charles Perrin, Claude Michel, and Nicolas Le Moine. Spatial proximity, physical similarity, regression and ungaged catchments: A comparison of regionalization approaches based on 913 french catchments. *Water Resources Research*, 44(3), 2008.

[35] 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, 2019.

[36] Keith Beven and Jim Freer. Equifinality, data assimilation, and uncertainty estimation in mechanistic modelling of complex environmental systems using the glue methodology. *Journal of hydrology*, 249(1-4):11–29, 2001.

[37] Saman Razavi, David M Hannah, Amin Elshorbagy, Sujay Kumar, Lucy Marshall, Dimitri P Solomatine, Amin Dezfuli, Mojtaba Sadegh, and James Famiglietti. Coevolution of machine learning and process-based modelling to revolutionize earth and environmental sciences: A perspective. *Hydrological Processes*, 36(6):e14596, 2022.

[38] Rahul Ghosh, Arvind Renganathan, Kshitij Tayal, Xiang Li, Ankush Khandelwal, Xiaowei Jia, Christopher Duffy, John Nieber, and Vipin Kumar. Robust inverse framework using knowledge-guided self-supervised learning: An application to hydrology. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 465–474, 2022.

[39] Shengyu Chen, Jacob A Zwart, and Xiaowei Jia. Physics-guided graph meta learning for predicting water temperature and streamflow in stream networks. In

Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, pages 2752–2761, 2022.

[40] Simon N Topp, Tamlin M Pavelsky, Daniel Jensen, Marc Simard, and Matthew RV Ross. Research trends in the use of remote sensing for inland water quality science: Moving towards multidisciplinary applications. *Water*, 12(1):169, 2020.

[41] Daniel Odermatt, Anatoly Gitelson, Vittorio Ernesto Brando, and Michael Schaepman. Review of constituent retrieval in optically deep and complex waters from satellite imagery. *Remote sensing of environment*, 118:116–126, 2012.

[42] Mohammad Haji Gholizadeh, Assefa M Melesse, and Lakshmi Reddi. A comprehensive review on water quality parameters estimation using remote sensing techniques. *Sensors*, 16(8):1298, 2016.

[43] Claudia Giardino, VE Brando, Peter Gege, Nicole Pinnel, Eric Hochberg, E Knaeps, Ils Reusen, Roland Doerffer, Mariano Bresciani, F Braga, et al. Imaging spectrometry of inland and coastal waters: state of the art, achievements and perspectives. *Surveys in Geophysics*, 40(3):401–429, 2019.

[44] 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, December 2019.

[45] R Ghosh, Haoyu Yang, Ankush Khandelwal, Erhu He, Arvind Renganathan, Somya Sharma, Xiaowei Jia, and Vipin Kumar. Entity aware modelling: A survey. *arXiv preprint arXiv:XXXX.XXXXX*, 2023.

[46] Yuhan Guo, Yongqiang Zhang, Lu Zhang, and Zhonggen Wang. Regionalization of hydrological modeling for predicting streamflow in ungauged catchments: A comprehensive review. *Wiley Interdisciplinary Reviews: Water*, 8(1):e1487, 2021.

[47] Saeed Golian, Conor Murphy, and Hadush Meresa. Regionalization of hydrological models for flow estimation in ungauged catchments in ireland. *Journal of Hydrology: Regional Studies*, 36:100859, 2021.

[48] Chen Lin, Yuan Zhang, Julie Ivy, Muge Capan, Ryan Arnold, Jeanne M Huddleston, and Min Chi. Early diagnosis and prediction of sepsis shock by combining static and dynamic information using convolutional-lstm. In *2018 IEEE International Conference on Healthcare Informatics (ICHI)*, pages 219–228. IEEE, 2018.

[49] Molla Hafizur Rahman, Shuhan Yuan, Charles Xie, and Zhenghui Sha. Predicting human design decisions with deep recurrent neural network combining static and dynamic data. *Design Science*, 6:e15, 2020.

[50] Dingwen Li, Patrick G Lyons, Jeff Klaus, Brian F Gage, Marin Kollef, and Chenyang Lu. Integrating static and time-series data in deep recurrent models for oncology early warning systems. In *CIKM*, pages 913–936, 2021.

[51] 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, 2019.

[52] Richard Arsenault, Jean-Luc Martel, Frédéric Brunet, François Brissette, and Juliane Mai. Continuous streamflow prediction in ungauged basins: long short-term memory neural networks clearly outperform traditional hydrological models. *Hydrology and Earth System Sciences*, 27(1):139–157, 2023.

[53] Shijie Jiang, Yi Zheng, and Dimitri Solomatine. Improving AI System Awareness of Geoscience Knowledge: Symbiotic Integration of Physical Approaches and Deep Learning. *Geophysical Research Letters*, 47(13), July 2020.

[54] Georgy Ayzel, Liubov Kurochkina, Eduard Kazakov, and Sergei Zhuravlev. Streamflow prediction in ungauged basins: benchmarking the efficiency of deep learning. In *E3S Web of Conferences*, volume 163, page 01001. EDP Sciences, 2020.

[55] Francisco José Matos Nogueira Filho, Francisco de Assis Souza Filho, Victor Costa Porto, Renan Vieira Rocha, Ályson Brayner Sousa Estácio, and Eduardo Sávio

Passos Rodrigues Martins. Deep learning for streamflow regionalization for ungauged basins: Application of long-short-term-memory cells in semiarid regions. *Water*, 14(9):1318, 2022.

[56] Jeonghyeon Choi, Jeonghoon Lee, and Sangdan Kim. Utilization of the long short-term memory network for predicting streamflow in ungauged basins in korea. *Ecological Engineering*, 182:106699, 2022.

[57] Yuan-Heng Wang, Hoshin V Gupta, Xubin Zeng, and Guo-Yue Niu. Exploring the potential of long short-term memory networks for improving understanding of continental-and regional-scale snowpack dynamics. *Water Resources Research*, 58(3):e2021WR031033, 2022.

[58] Jiaxin Xie, Xiaomang Liu, Wei Tian, Kaiwen Wang, Peng Bai, and Changming Liu. Estimating gridded monthly baseflow from 1981 to 2020 for the contiguous us using long short-term memory (lstm) networks. *Water Resources Research*, 58(8):e2021WR031663, 2022.

[59] Wei Zhi, Dapeng Feng, Wen-Ping Tsai, Gary Sterle, Adrian Harpold, Chaopeng Shen, and Li Li. From hydrometeorology to river water quality: can a deep learning model predict dissolved oxygen at the continental scale? *Environmental Science & Technology*, 55(4):2357–2368, 2021.

[60] Junjie Jiang, Zi-Gang Huang, Celso Grebogi, and Ying-Cheng Lai. Predicting extreme events from data using deep machine learning: When and where. *Physical Review Research*, 4(2):023028, 2022.

[61] Mohammad Zounemat-Kermani, Okke Batelaan, Marzieh Fadaee, and Reinhard Hinkelmann. Ensemble machine learning paradigms in hydrology: A review. *Journal of Hydrology*, 598:126266, 2021.

[62] Jonathan M Frame, Frederik Kratzert, Daniel Klotz, Martin Gauch, Guy Shelev, Oren Gilon, Logan M Qualls, Hoshin V Gupta, and Grey S Nearing. Deep learning rainfall–runoff predictions of extreme events. *Hydrology and Earth System Sciences*, 26(13):3377–3392, 2022.

[63] A Viglione, J Parajka, M Rogger, JL Salinas, G Laaha, M Sivapalan, and G Blöschl. Comparative assessment of predictions in ungauged basins–part 3: Runoff signatures in austria. *Hydrology and Earth System Sciences*, 17(6):2263–2279, 2013.

[64] Zimeena Rasheed, Akshay Aravamudan, Ali Gorji Sefidmazgi, Georgios C Anagnostopoulos, and Efthymios I Nikolopoulos. Advancing flood warning procedures in ungauged basins with machine learning. *Journal of Hydrology*, 609:127736, 2022.

[65] Cristóbal Esteban, Oliver Staeck, Stephan Baier, Yinchong Yang, and Volker Tresp. Predicting clinical events by combining static and dynamic information using recurrent neural networks. In *2016 IEEE International Conference on Healthcare Informatics (ICHI)*, pages 93–101. Ieee, 2016.

[66] Kshitij Tayal, Xiaowei Jia, Rahul Ghosh, Jared Willard, Jordan Read, and Vipin Kumar. Invertibility aware integration of static and time-series data: An application to lake temperature modeling. In *Proceedings of the 2022 SIAM International Conference on Data Mining (SDM)*, pages 702–710. SIAM, 2022.

[67] Dapeng Feng, Kathryn Lawson, and Chaopeng Shen. Mitigating prediction error of deep learning streamflow models in large data-sparse regions with ensemble modeling and soft data. *Geophysical Research Letters*, 48(14):e2021GL092999, 2021.

[68] DG George, JF Talling, and E Rigg. Factors influencing the temporal coherence of five lakes in the english lake district. *Freshwater Biology*, 43(3):449–461, 2000.

[69] John J Magnuson, BJ Benson, and TK Kratz. Temporal coherence in the limnology of a suite of lakes in wisconsin, usa. *Freshwater Biology*, 23(1):145–159, 1990.

[70] Thomas G Huntington, GA Hodgkins, and RW Dudley. Historical trend in river ice thickness and coherence in hydroclimatological trends in maine. *Climatic Change*, 61(1-2):217–236, 2003.

[71] Daniel G Kingston, Glenn R McGregor, David M Hannah, and Damian M Lawler. River flow teleconnections across the northern north atlantic region. *Geophysical Research Letters*, 33(14), 2006.

[72] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and S Yu Philip. A comprehensive survey on graph neural networks. *IEEE transactions on neural networks and learning systems*, 32(1):4–24, 2020.

[73] Michael M Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. Geometric deep learning: going beyond euclidean data. *IEEE Signal Processing Magazine*, 34(4):18–42, 2017.

[74] Peter W Battaglia, Jessica B Hamrick, Victor Bapst, Alvaro Sanchez-Gonzalez, Vinicius Zambaldi, Mateusz Malinowski, Andrea Tacchetti, David Raposo, Adam Santoro, Ryan Faulkner, et al. Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*, 2018.

[75] Jie Zhou, Ganqu Cui, Shengding Hu, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. Graph neural networks: A review of methods and applications. *AI open*, 1:57–81, 2020.

[76] Huafei Yu, Tinghua Ai, Min Yang, Lina Huang, and Jiaming Yuan. A recognition method for drainage patterns using a graph convolutional network. *International Journal of Applied Earth Observation and Geoinformation*, 107:102696, 2022.

[77] Tao Bai and Pejman Tahmasebi. Graph neural network for groundwater level forecasting. *Journal of Hydrology*, page 128792, 2022.

[78] Qun Zhao, Yuelong Zhu, Kai Shu, Dingsheng Wan, Yufeng Yu, Xudong Zhou, and Huan Liu. Joint spatial and temporal modeling for hydrological prediction. *Ieee Access*, 8:78492–78503, 2020.

[79] Frederik Kratzert, Daniel Klotz, Martin Gauch, Christoph Klingler, Grey Nearing, and Sepp Hochreiter. Large-scale river network modeling using graph neural networks. In *EGU General Assembly Conference Abstracts*, pages EGU21–13375, 2021.

[80] Muhammed Sit, Bekir Demiray, and Ibrahim Demir. Short-term hourly streamflow prediction with graph convolutional gru networks. *arXiv preprint arXiv:2107.07039*, 2021.

[81] Alexander Y Sun, Peishi Jiang, Maruti K Mudunuru, and Xingyuan Chen. Explore spatio-temporal learning of large sample hydrology using graph neural networks. *Water Resources Research*, 57(12):e2021WR030394, 2021.

[82] Jun Feng, Haichao Sha, Yukai Ding, Le Yan, and Zhangheng Yu. Graph convolution based spatial-temporal attention lstm model for flood forecasting. In *2022 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2022.

[83] Arnold N Kazadi, James Doss-Gollin, Antonia Sebastian, and Arlei Silva. Flood prediction with graph neural networks. In *NeurIPS 2022 Workshop on Tackling Climate Change with Machine Learning*, 2022.

[84] Michael Stalder, Firat Ozdemir, Artur Safin, Jonas Sukys, Damien Bouffard, and Fernando Perez-Cruz. Probabilistic modeling of lake surface water temperature using a bayesian spatio-temporal graph convolutional neural network. *arXiv preprint arXiv:2109.13235*, 2021.

[85] Shengyu Chen, Alison Appling, Samantha Oliver, Hayley Corson-Dosch, Jordan Read, Jeffrey Sadler, Jacob Zwart, and Xiaowei Jia. Heterogeneous stream-reservoir graph networks with data assimilation. In *2021 IEEE International Conference on Data Mining (ICDM)*, pages 1024–1029. IEEE, 2021.

[86] Tianshu Bao, Xiaowei Jia, Jacob Zwart, Jeffrey Sadler, Alison Appling, Samantha Oliver, and Taylor T Johnson. Partial Differential Equation Driven Dynamic Graph Networks for Predicting Stream Water Temperature. In *2021 IEEE International Conference on Data Mining (ICDM)*, pages 11–20. IEEE, 2021.

[87] Xiaoyu Zhang, Yongqing Li, Alejandro C Frery, and Peng Ren. Sea surface temperature prediction with memory graph convolutional networks. *IEEE Geoscience and Remote Sensing Letters*, 19:1–5, 2021.

[88] Yongjiao Sun, Xin Yao, Xin Bi, Xuechun Huang, Xiangguo Zhao, and Baiyou Qiao. Time-series graph network for sea surface temperature prediction. *Big Data Research*, 25:100237, 2021.

[89] Zhongrun Xiang and Ibrahim Demir. High-resolution rainfall-runoff modeling using graph neural network. *arXiv preprint arXiv:2110.10833*, 2021.

[90] Shengyu Chen, Alison Appling, Samantha Oliver, Hayley Corson-Dosch, Jordan Read, Jeffrey Sadler, Jacob Zwart, and Xiaowei Jia. Heterogeneous Stream-reservoir Graph Networks with Data Assimilation. *arXiv:2110.04959 [cs]*, October 2021.

[91] Tadd Bindas, Chaopeng Shen, and Yuchen Bian. Routing flood waves through the river network utilizing physics-guided machine learning and the muskingum-cunge method. In *AGU Fall Meeting Abstracts*, volume 2020, pages H224–04, 2020.

[92] Giorgio Ciano, Alberto Rossi, Monica Bianchini, and Franco Scarselli. On inductive–transductive learning with graph neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(2):758–769, 2021.

[93] Zach Moshe, Asher Metzger, Gal Elidan, Frederik Kratzert, Sella Nevo, and Ran El-Yaniv. Hydronets: Leveraging river structure for hydrologic modeling. *arXiv preprint arXiv:2007.00595*, 2020.

[94] Alberto Rossi, Matteo Tiezzi, Giovanna Maria Dimitri, Monica Bianchini, Marco Maggini, and Franco Scarselli. Inductive–transductive learning with graph neural networks. In *Artificial Neural Networks in Pattern Recognition: 8th IAPR TC3 Workshop, ANNPR 2018, Siena, Italy, September 19–21, 2018, Proceedings 8*, pages 201–212. Springer, 2018.

[95] Biwei Yan, Guijuan Wang, Jiguo Yu, Xiaozheng Jin, and Hongliang Zhang. Spatial-temporal chebyshev graph neural network for traffic flow prediction in iot-based its. *IEEE Internet of Things Journal*, 9(12):9266–9279, 2021.

[96] Youngjoo Seo, Michaël Defferrard, Pierre Vandergheynst, and Xavier Bresson. Structured sequence modeling with graph convolutional recurrent networks. In

Neural Information Processing: 25th International Conference, ICONIP 2018, Siem Reap, Cambodia, December 13-16, 2018, Proceedings, Part I 25, pages 362–373. Springer, 2018.

[97] Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, and Chengqi Zhang. Graph wavenet for deep spatial-temporal graph modeling. *arXiv preprint arXiv:1906.00121*, 2019.

[98] Antonio Longa, Veronica Lachi, Gabriele Santin, Monica Bianchini, Bruno Lepri, Pietro Lio, Franco Scarselli, and Andrea Passerini. Graph neural networks for temporal graphs: State of the art, open challenges, and opportunities. *arXiv preprint arXiv:2302.01018*, 2023.

[99] Min Shi, Yufei Tang, Xingquan Zhu, David Wilson, and Jianxun Liu. Multi-class imbalanced graph convolutional network learning. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence (IJCAI-20)*, 2020.

[100] Deli Chen, Yankai Lin, Guangxiang Zhao, Xuancheng Ren, Peng Li, Jie Zhou, and Xu Sun. Topology-imbalance learning for semi-supervised node classification. *Advances in Neural Information Processing Systems*, 34:29885–29897, 2021.

[101] Scott R Wilson, Murray E Close, Phillip Abraham, Theo S Sarris, Laura Banasiak, Roland Stenger, and John Hadfield. Achieving unbiased predictions of national-scale groundwater redox conditions via data oversampling and statistical learning. *Science of the Total Environment*, 705:135877, 2020.

[102] Dennis P Lettenmaier, James R Wallis, and Eric F Wood. Effect of regional heterogeneity on flood frequency estimation. *Water Resources Research*, 23(2):313–323, 1987.

[103] Jonathan Richard Morley Hosking and James R Wallis. *Regional frequency analysis*. Cambridge University Press, 1997.

[104] Donald H Burn. Evaluation of regional flood frequency analysis with a region of influence approach. *Water Resources Research*, 26(10):2257–2265, 1990.

[105] Donald H Burn. An appraisal of the "region of influence" approach to flood frequency analysis. *Hydrological Sciences Journal*, 35(2):149–165, 1990.

[106] Hakan Tongal and Bellie Sivakumar. Cross-entropy clustering framework for catchment classification. *Journal of Hydrology*, 552:433–446, 2017.

[107] Ersin Aytac. Unsupervised learning approach in defining the similarity of catchments: Hydrological response unit based k-means clustering, a demonstration on western black sea region of turkey. *International soil and water conservation research*, 8(3):321–331, 2020.

[108] Elnaz Sharghi, Vahid Nourani, Saeed Soleimani, and Fahreddin Sadikoglu. Application of different clustering approaches to hydroclimatological catchment regionalization in mountainous regions, a case study in utah state. *Journal of Mountain Science*, 15(3):461–484, 2018.

[109] Arnan Araza, Lars Hein, Confidence Duku, Maurice Andres Rawlins, and Richard Lomboy. Data-driven streamflow modelling in ungauged basins: regionalizing random forest (rf) models. *bioRxiv*, pages 2020–11, 2020.

[110] Zhijun Chen, Zhenchuang Zhu, Hao Jiang, and Shijun Sun. Estimating daily reference evapotranspiration based on limited meteorological data using deep learning and classical machine learning methods. *Journal of Hydrology*, 591:125286, 2020.

[111] Steven M Corns, Suzanna K Long, Jacob Hale, Bhanu Kanwar, Samuel Vanfossan, et al. Deep learning for unmonitored water level prediction and risk assessment. Technical report, Missouri. Department of Transportation. Construction and Materials Division, 2022.

[112] Kuai Fang, Daniel Kifer, Kathryn Lawson, Dapeng Feng, and Chaopeng Shen. The data synergy effects of time-series deep learning models in hydrology. *Water Resources Research*, 58(4):e2021WR029583, 2022.

[113] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2009.

[114] Karl Weiss, Taghi M Khoshgoftaar, and DingDing Wang. A survey of transfer learning. *Journal of Big data*, 3(1):9, 2016.

[115] Rohini Kumar, Luis Samaniego, and Sabine Attinger. Implications of distributed hydrologic model parameterization on water fluxes at multiple scales and locations. *Water Resources Research*, 49(1):360–379, 2013.

[116] Vincent Roth, Tibebu Kassawmar Nigussie, and Tatenda Lemann. Model parameter transfer for streamflow and sediment loss prediction with swat in a tropical watershed. *Environmental Earth Sciences*, 75(19):1321, 2016.

[117] Ehsan Zabihi Naeini and Joshua Uwaifo. Transfer learning and auto-ml: A geoscience perspective. *First Break*, 37(9):65–71, 2019.

[118] Nobuaki Kimura, Ikuo Yoshinaga, Kenji Sekijima, Issaku Azechi, and Daichi Baba. Convolutional neural network coupled with a transfer-learning approach for time-series flood predictions. *Water*, 12(1):96, 2019.

[119] Gang Zhao, Bo Pang, Zongxue Xu, Lizhuang Cui, Jingjing Wang, Depeng Zuo, and Dingzhi Peng. Improving urban flood susceptibility mapping using transfer learning. *Journal of Hydrology*, 602:126777, 2021.

[120] Qingliang Li, Ziyu Wang, Wei Shangguan, Lu Li, Yifei Yao, and Fanhua Yu. Improved daily smap satellite soil moisture prediction over china using deep learning model with transfer learning. *Journal of Hydrology*, 600:126698, 2021.

[121] Wenchong Tian, Zhenliang Liao, and Xuan Wang. Transfer learning for neural network model in chlorophyll-a dynamics prediction. *Environmental Science and Pollution Research*, 26(29):29857–29871, 2019.

[122] Sadegh Sadeghi Tabas and Vidya Samadi. Structure learning and transfer learning for streamflow prediction across ungauged basins. In *AGU Fall Meeting 2021*. AGU, 2021.

[123] Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 109(1):43–76, 2020.

[124] Bernardino Romera-Paredes and Philip Torr. An embarrassingly simple approach to zero-shot learning. In *International conference on machine learning*, pages 2152–2161. PMLR, 2015.

[125] Lakhwinder Singh, Prabhash Kumar Mishra, Santosh Murlidhar Pingale, Deepak Khare, and Hitesh Prasad Thakur. Streamflow regionalisation of an ungauged catchment with machine learning approaches. *Hydrological Sciences Journal*, 67(6):886–897, 2022.

[126] Manh-Hung Le, Hyunglok Kim, Stephen Adam, Hong Xuan Do, Peter Beling, and Venkataraman Lakshmi. Streamflow estimation in ungauged regions using machine learning: Quantifying uncertainties in geographic extrapolation. *Hydrology and Earth System Sciences Discussions*, pages 1–24, 2022.

[127] Donald H Burn and David B Boorman. Estimation of hydrological parameters at ungauged catchments. *Journal of Hydrology*, 143(3-4):429–454, 1993.

[128] Babak Vaheddoost, Mir Jafar Sadegh Safari, and Mustafa Utku Yilmaz. Rainfall-runoff simulation in ungauged tributary streams using drainage area ratio-based multivariate adaptive regression spline and random forest hybrid models. *Pure and Applied Geophysics*, pages 1–18, 2023.

[129] Christiane Lemke, Marcin Budka, and Bogdan Gabrys. Metalearning: a survey of trends and technologies. *Artificial Intelligence Review*, 44(1):117–130, June 2015.

[130] Pavel B. Brazdil, editor. *Metalearning: applications to data mining*. Cognitive technologies. Springer, Berlin, 2009. OCLC: ocn298595059.

[131] Arghya Pal and Vineeth N Balasubramanian. Zero-shot task transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2189–2198, 2019.

[132] Matthew R. Hipsey, Louise C. Bruce, Casper Boon, Brendan Busch, Cayelan C. Carey, David P. Hamilton, Paul C. Hanson, Jordan S. Read, Eduardo de Sousa, Michael Weber, and Luke A. Winslow. A General Lake Model (GLM 3.0) for linking with high-frequency sensor data from the Global Lake Ecological Observatory

Network (GLEON). *Geoscientific Model Development*, 12(1):473–523, January 2019.

[133] Rahul Ghosh, Bangyan Li, Kshitij Tayal, Vipin Kumar, and Xiaowei Jia. Meta-transfer learning: An application to streamflow modeling in river-streams. In *2022 IEEE International Conference on Data Mining (ICDM)*, pages 161–170. IEEE, 2022.

[134] J Padarian, B Minasny, and AB McBratney. Transfer learning to localise a continental soil vis-nir calibration model. *Geoderma*, 340:279–288, 2019.

[135] Zefang Shen, Leonardo Ramirez-Lopez, Thorsten Behrens, Lei Cui, Mingxi Zhang, Lewis Walden, Johanna Wetterlind, Zhou Shi, Kenneth A Sudduth, Philipp Baumann, et al. Deep transfer learning of global spectra for local soil carbon monitoring. *ISPRS Journal of Photogrammetry and Remote Sensing*, 188:190–200, 2022.

[136] Xuejun Guo, Yin Chen, Xiaofeng Liu, and Yue Zhao. Extraction of snow cover from high-resolution remote sensing imagery using deep learning on a small dataset. *Remote Sensing Letters*, 11(1):66–75, 2020.

[137] Jiwen Wang, Qiangqiang Yuan, Huanfeng Shen, Tingting Liu, Tongwen Li, Linwei Yue, Xiaogang Shi, and Liangpei Zhang. Estimating snow depth by combining satellite data and ground-based observations over alaska: A deep learning approach. *Journal of Hydrology*, 585:124828, 2020.

[138] Rui Xiong, Yi Zheng, Nengwang Chen, Qing Tian, Wei Liu, Feng Han, Shijie Jiang, Mengqian Lu, and Yan Zheng. Predicting dynamic riverine nitrogen export in unmonitored watersheds: Leveraging insights of ai from data-rich regions. *Environmental Science & Technology*, 56(14):10530–10542, 2022.

[139] Bernhard Nornes Lotsberg. Lstm models applied on hydrological time series. Master's thesis, University of Oslo, 2021.

[140] Sinno Jialin Pan and Qiang Yang. A Survey on Transfer Learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, October 2010.

[141] Shuteng Niu, Yongxin Liu, Jian Wang, and Houbing Song. A decade survey of transfer learning (2010–2020). *IEEE Transactions on Artificial Intelligence*, 1(2):151–166, 2020.

[142] Vishal M Patel, Raghuraman Gopalan, Ruonan Li, and Rama Chellappa. Visual domain adaptation: A survey of recent advances. *IEEE signal processing magazine*, 32(3):53–69, 2015.

[143] Gabriela Csurka. A comprehensive survey on domain adaptation for visual applications. *Domain adaptation in computer vision applications*, pages 1–35, 2017.

[144] Judy Hoffman, Dequan Wang, Fisher Yu, and Trevor Darrell. Fcns in the wild: Pixel-level adversarial and constraint-based adaptation. *arXiv preprint arXiv:1612.02649*, 2016.

[145] Konstantinos Bousmalis, Alex Irpan, Paul Wohlhart, Yunfei Bai, Matthew Kelcey, Mrinal Kalakrishnan, Laura Downs, Julian Ibarz, Peter Pastor, Kurt Konolige, et al. Using simulation and domain adaptation to improve efficiency of deep robotic grasping. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 4243–4250. IEEE, 2018.

[146] John Blitzer, Mark Dredze, and Fernando Pereira. Biographies, bollywood, boomboxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of the 45th annual meeting of the association of computational linguistics*, pages 440–447, 2007.

[147] Yongjie Shi, Xianghua Ying, and Jinfa Yang. Deep unsupervised domain adaptation with time series sensor data: A survey. *Sensors*, 22(15):5507, 2022.

[148] Ruizhi Zhou and Yanling Pan. Flooddan: Unsupervised flood forecasting based on adversarial domain adaptation. In *2022 IEEE 5th International Conference on Big Data and Artificial Intelligence (BDAI)*, pages 6–12. IEEE, 2022.

[149] Anuj Karpatne, Ramakrishnan Kannan, and Vipin Kumar. *Knowledge Guided Machine Learning: Accelerating Discovery Using Scientific Knowledge and Data*. CRC Press, 2022.

[150] Jared D. Willard, Xiaowei Jia, Shaoming Xu, Michael Steinbach, and Vipin Kumar. Integrating Scientific Knowledge with Machine Learning for Engineering and Environmental Systems. *ACM Comput. Surv.*, January 2022.

[151] Temoor Muther, Amirmasoud Kalantari Dahaghi, Fahad Iqbal Syed, and Vuong Van Pham. Physical laws meet machine intelligence: current developments and future directions. *Artificial Intelligence Review*, pages 1–67, 2022.

[152] Anuj Karpatne, Gowtham Atluri, James H Faghmous, Michael Steinbach, Arindam Banerjee, Auroop Ganguly, Shashi Shekhar, Nagiza Samatova, and Vipin Kumar. Theory-guided data science: A new paradigm for scientific discovery from data. *IEEE Transactions on knowledge and data engineering*, 29(10):2318–2331, 2017.

[153] Xiaowei Jia, Jared D Willard, Anuj Karpatne, Jordan S Read, Jacob A Zwart, Michael Steinbach, and Vipin Kumar. Physics-guided machine learning for scientific discovery: An application in simulating lake temperature profiles. *ACM/IMS Transactions on Data Science*, 2(3):1–26, 2021.

[154] Herath Mudiyanselage Viraj Vidura Herath, Jayashree Chadalawada, and Vladan Babovic. Hydrologically informed machine learning for rainfall–runoff modelling: towards distributed modelling. *Hydrology and Earth System Sciences*, 25(8):4373–4401, 2021.

[155] Pieter-Jan Hoedt, Frederik Kratzert, Daniel Klotz, Christina Halmich, Markus Holzleitner, Grey Nearing, Sepp Hochreiter, and Günter Klambauer. MC-LSTM: Mass-Conserving LSTM. *arXiv:2101.05186 [cs, stat]*, January 2021.

[156] Pravin Bhasme, Jenil Vagadiya, and Udit Bhatia. Enhancing predictive skills in physically-consistent way: Physics informed machine learning for hydrological processes. *Journal of Hydrology*, page 128618, 2022.

[157] Mario A Soriano, Helen G Siegel, Nicholaus P Johnson, Kristina M Gutchess, Boya Xiong, Yunpo Li, Cassandra J Clark, Desiree L Plata, Nicole C Deziel,

and James E Saiers. Assessment of groundwater well vulnerability to contamination through physics-informed machine learning. *Environmental Research Letters*, 16(8):084013, 2021.

[158] Brenda Ng, Vidya Samadi, Cheng Wang, and Jie Bao. Physics-informed deep learning for multiscale water cycle prediction. Technical report, Lawrence Livermore National Lab.(LLNL), Livermore, CA (United States . . . , 2021.

[159] Anuj Karpatne, William Watkins, Jordan Read, and Vipin Kumar. Physics-guided neural networks (pgnn): An application in lake temperature modeling. *arXiv preprint arXiv:1710.11431*, 2017.

[160] Paul C Hanson, Aviah B Stillman, Xiaowei Jia, Anuj Karpatne, Hilary A Dugan, Cayelan C Carey, Jemma Stachelek, Nicole K Ward, Yu Zhang, Jordan S Read, and others. Predicting lake surface water phosphorus dynamics using process-guided machine learning. *Ecological Modelling*, 430:109136, 2020.

[161] Nanzhe Wang, Dongxiao Zhang, Haibin Chang, and Heng Li. Deep learning of subsurface flow via theory-guided neural network. *Journal of Hydrology*, 584:124700, 2020.

[162] Arka Daw and Anuj Karpatne. Physics-aware Architecture of Neural Networks for Uncertainty Quantification: Application in Lake Temperature Modeling. In *FEED Workshop at Knowledge Discovery and Data Mining Conference (SIGKDD) 2019*. SIGKDD, 2019.

[163] NOAA. National water model. improving noaa's water prediction services. Technical report, National Oceanic and Atmospheric Administration, 2016.

[164] Steven L Markstrom. *P2S–coupled Simulation with the Precipitation-Runoff Modeling System (PRMS) and the Stream Temperature Network (SNTemp) Models*. US Department of the Interior, US Geological Survey, 2012.

[165] Julian Koch and Raphael Schneider. Long short-term memory networks enhance rainfall-runoff modelling at the national scale of denmark. *GEUS Bulletin*, 49, 2022.

[166] Navideh Noori, Latif Kalin, and Sabahattin Isik. Water quality prediction using swat-ann coupled approach. *Journal of Hydrology*, 590:125220, 2020.

[167] 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, 2021.

[168] Tianfang Xu and Feng Liang. Machine learning for hydrologic sciences: An introductory overview. *Wiley Interdisciplinary Reviews: Water*, 8(5):e1533, 2021.

[169] Satish Kumar Regonda, Dong-Jun Seo, Bill Lawrence, James D Brown, and Julie Demargne. Short-term ensemble streamflow forecasting using operationally-produced single-valued streamflow forecasts–a hydrologic model output statistics (hmos) approach. *Journal of Hydrology*, 497:80–96, 2013.

[170] Kyeungwoo Cho and Yeonjoo Kim. Improving streamflow prediction in the wrf-hydro model with lstm networks. *Journal of Hydrology*, 605:127297, 2022.

[171] P López López, JS Verkade, AH Weerts, and DP Solomatine. Alternative configurations of quantile regression for estimating predictive uncertainty in water level forecasts for the upper severn river: a comparison. *Hydrology and Earth System Sciences*, 18(9):3411–3428, 2014.

[172] Tianfang Xu and Albert J Valocchi. Data-driven methods to improve baseflow prediction of a regional groundwater model. *Computers & Geosciences*, 85:124–136, 2015.

[173] Riley C Hales, Robert B Sowby, Gustavious P Williams, E James Nelson, Daniel P Ames, Jonah B Dundas, and Josh Ogden. Saber: A model-agnostic postprocessor for bias correcting discharge from large hydrologic models. *Hydrology*, 9(7):113, 2022.

[174] Tao Yang, Fubao Sun, Pierre Gentine, Wenbin Liu, Hong Wang, Jiabo Yin, Muye Du, and Changming Liu. Evaluation and machine learning improvement of global hydrological model-based flood simulations. *Environmental Research Letters*, 14(11):114027, 2019.

[175] Jeffrey G Arnold, Raghavan Srinivasan, Ranjan S Muttiah, and Jimmy R Williams. Large area hydrologic modeling and assessment part i: model development 1. *JAWRA Journal of the American Water Resources Association*, 34(1):73–89, 1998.

[176] R Cibin, P Athira, KP Sudheer, and I Chaubey. Application of distributed hydrological models for predictions in ungauged basins: a method to quantify predictive uncertainty. *Hydrological Processes*, 28(4):2033–2045, 2014.

[177] Muhammad Waseem, Muhammad Ajmal, and Tae-Woong Kim. Ensemble hydrological prediction of streamflow percentile at ungauged basins in pakistan. *Journal of Hydrology*, 525:130–137, 2015.

[178] Tara Razavi and Paulin Coulibaly. Improving streamflow estimation in ungauged basins using a multi-modelling approach. *Hydrological Sciences Journal*, 61(15):2668–2679, 2016.

[179] Jonathan M Frame, Frederik Kratzert, Austin Raney, Mashrekur Rahman, Fernando R Salas, and Grey S Nearing. Post-processing the national water model with long short-term memory networks for streamflow predictions and model diagnostics. *JAWRA Journal of the American Water Resources Association*, 57(6):885–905, 2021.

[180] Ankush Khandelwal, Shaoming Xu, Xiang Li, Xiaowei Jia, Michael Stienbach, Christopher Duffy, John Nieber, and Vipin Kumar. Physics guided machine learning methods for hydrology. *arXiv preprint arXiv:2012.02854*, 2020.

[181] 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.

[182] Chaopeng Shen, Alison P Appling, Pierre Gentine, Toshiyuki Bandai, Hoshin Gupta, Alexandre Tartakovsky, Marco Baity-Jesi, Fabrizio Fenicia, Daniel Kifer, Li Li, et al. Differentiable modeling to unify machine learning and physical models and advance geosciences. *arXiv preprint arXiv:2301.04027*, 2023.

[183] Maximilian Gelbrecht, Alistair White, Sebastian Bathiany, and Niklas Boers. Differentiable programming for earth system modeling. *arXiv preprint arXiv:2208.13825*, 2022.

[184] Mohammed AlQuraishi and Peter K Sorger. Differentiable biology: using deep learning for biophysics-based and data-driven modeling of molecular mechanisms. *Nature methods*, 18(10):1169–1180, 2021.

[185] Sten Bergström. Development and application of a conceptual runoff model for scandinavian catchments. *Hydrology and Oceanography*, 1976.

[186] Dapeng Feng, Hylke Beck, Kathryn Lawson, and Chaopeng Shen. The suitability of differentiable, learnable hydrologic models for ungauged regions and climate change impact assessment. *Hydrology and Earth System Sciences Discussions*, pages 1–28, 2022.

[187] Latif Kalin, Sabahattin Isik, Jon E Schoonover, and B Graeme Lockaby. Predicting water quality in unmonitored watersheds using artificial neural networks. *Journal of environmental quality*, 39(4):1429–1440, 2010.

[188] Jin-Young Lee, Changhyun Choi, Doosun Kang, Byung Sik Kim, and Tae-Woong Kim. Estimating design floods at ungauged watersheds in south korea using machine learning models. *Water*, 12(11):3022, 2020.

[189] Aggrey Muhebwa, Sungwook Wi, Colin J Gleason, and Jay Taneja. Towards improved global river discharge prediction in ungauged basins using machine learning and satellite observations. *NeurIPS*, 2021.

[190] 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.

[191] Akhil Sanjay Potdar, Pierre-Emmanuel Kirstetter, Devon Woods, and Manabendra Saharia. Toward predicting flood event peak discharge in ungauged basins by learning universal hydrological behaviors with machine learning. *Journal of Hydrometeorology*, 22(11):2971–2982, 2021.

[192] Elaheh White. *Predicting Unimpaired Flow in Ungauged Basins:" Random Forests" Applied to California Streams*. University of California, Davis, 2017.

[193] 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, July 2021.

[194] Jingbo Hao and Yang Tao. Adversarially robust water quality assessment associated with power plants. *Energy Reports*, 8:37–45, 2022.

[195] Jianfeng Zhang, Yan Zhu, Xiaoping Zhang, Ming Ye, and Jinzhong Yang. Developing a long short-term memory (lstm) based model for predicting water table depth in agricultural areas. *Journal of hydrology*, 561:918–929, 2018.

[196] Hongxiang Fan, Mingliang Jiang, Ligang Xu, Hua Zhu, Junxiang Cheng, and Jiahu Jiang. Comparison of long short term memory networks and the hydrological model in runoff simulation. *Water*, 12(1):175, 2020.

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

[198] Colin Lea, Michael D Flynn, Rene Vidal, Austin Reiter, and Gregory D Hager. Temporal convolutional networks for action segmentation and detection. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 156–165, 2017.

[199] Kangling Lin, Sheng Sheng, Yanlai Zhou, Feng Liu, Zhiyu Li, Hua Chen, Chong-Yu Xu, Jie Chen, and Shenglian Guo. The exploration of a temporal convolutional network combined with encoder-decoder framework for runoff forecasting. *Hydrology Research*, 51(5):1136–1149, 2020.

[200] Yuanhao Xu, Caihong Hu, Qiang Wu, Zhichao Li, Shengqi Jian, and Youqian Chen. Application of temporal convolutional network for flood forecasting. *Hydrology Research*, 52(6):1455–1468, 2021.

[201] 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.

[202] International Rivers. Damming statistics, 10 2007.

[203] US Army Core of Engineers, 2020.

[204] John C Risley, Jim Constantz, Hedeff Essaid, and Stewart Rounds. Effects of upstream dams versus groundwater pumping on stream temperature under varying climate conditions. *Water Resources Research*, 46(6), 2010.

[205] M Lockhoff, O Zolina, C Simmer, and J Schulz. Evaluation of satellite-retrieved extreme precipitation over europe using gauge observations. *Journal of climate*, 27(2):607–623, 2014.

[206] Ashok Mishra, Tue Vu, Anoop Valiya Veettil, and Dara Entekhabi. Drought monitoring with soil moisture active passive (smap) measurements. *Journal of Hydrology*, 552:620–632, 2017.

[207] Yaoze Liu, Bernard A Engel, Dennis C Flanagan, Margaret W Gitau, Sara K McMillan, and Indrajeet Chaubey. A review on effectiveness of best management practices in improving hydrology and water quality: needs and opportunities. *Science of the Total Environment*, 601:580–593, 2017.

[208] Rosana Aguilera, David M Livingstone, Rafael Marcé, Eleanor Jennings, Jaume Piera, and Rita Adrian. Using dynamic factor analysis to show how sampling resolution and data gaps affect the recognition of patterns in limnological time series. *Inland Waters*, 6(3):284–294, 2016.

[209] Gary M Lovett, Douglas A Burns, Charles T Driscoll, Jennifer C Jenkins, Myron J Mitchell, Lindsey Rustad, James B Shanley, Gene E Likens, and Richard Haeuber. Who needs environmental monitoring? *Frontiers in Ecology and the Environment*, 5(5):253–260, 2007.

[210] Lynne Caughlan and Karen L Oakley. Cost considerations for long-term ecological monitoring. *Ecological indicators*, 1(2):123–134, 2001.

[211] Stephen J Dugdale, David M Hannah, and Iain A Malcolm. River temperature modelling: A review of process-based approaches and future directions. *Earth-Science Reviews*, 175:97–113, 2017.

[212] Claudio Paniconi and Mario Putti. Physically based modeling in catchment hydrology at 50: Survey and outlook. *Water Resources Research*, 51(9):7090–7129, 2015.

[213] Simone Fatichi, Enrique R Vivoni, Fred L Ogden, Valeriy Y Ivanov, Benjamin Mirus, David Gochis, Charles W Downer, Matteo Camporese, Jason H Davison, Brian Ebel, et al. An overview of current applications, challenges, and future trends in distributed process-based models in hydrology. *Journal of Hydrology*, 537:45–60, 2016.

[214] Adrien Gaudard, Love Råman Vinnå, Fabian Bärenbold, Martin Schmid, and Damien Bouffard. Toward an open access to high-frequency lake modeling and statistics data for scientists and practitioners–the case of swiss lakes using simstrat v2. 1. *Geoscientific Model Development*, 12(9), 2019.

[215] Matthew R Hipsey, Louise C Bruce, Casper Boon, Brendan Busch, Cayelan C Carey, David P Hamilton, Paul C Hanson, Jordan S Read, Eduardo De Sousa, Michael Weber, et al. A general lake model (glm 3.0) for linking with high-frequency sensor data from the global lake ecological observatory network (gleon). *Geoscientific Model Development*, 2019.

[216] Luke A Winslow, Gretchen JA Hansen, Jordan S Read, and Michael Notaro. Large-scale modeled contemporary and future water temperature estimates for 10774 midwestern us lakes. *Scientific data*, 4:170053, 2017.

[217] K Cuddington, M-J Fortin, LR Gerber, Alan Hastings, A Liebhold, M O'connor, and C Ray. Process-based models are required to manage ecological systems in a changing world. *Ecosphere*, 4(2):1–12, 2013.

[218] Craig R White and Dustin J Marshall. Should we care if models are phenomenological or mechanistic? *Trends in ecology & evolution*, 34(4):276–278, 2019.

[219] Stephanie E Hampton, Carly A Strasser, Joshua J Tewksbury, Wendy K Gram, Amber E Budden, Archer L Batcheller, Clifford S Duke, and John H Porter. Big data and the future of ecology. *Frontiers in Ecology and the Environment*, 11(3):156–162, 2013.

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

[221] Markus Reichstein, Gustau Camps-Valls, Bjorn Stevens, Martin Jung, Joachim Denzler, Nuno Carvalhais, et al. Deep learning and process understanding for data-driven earth system science. *Nature*, 566(7743):195, 2019.

[222] Halil Ibrahim Erdal and Onur Karakurt. Advancing monthly streamflow prediction accuracy of cart models using ensemble learning paradigms. *Journal of Hydrology*, 477:119–128, 2013.

[223] Chaopeng Shen. A transdisciplinary review of deep learning research and its relevance for water resources scientists. *Water Resources Research*, 54(11):8558–8593, 2018.

[224] Hristos Tyralis, Georgia Papacharalampous, and Andreas Langousis. A brief review of random forests for water scientists and practitioners and their recent history in water resources. *Water*, 11(5):910, 2019.

[225] Anuj Karpatne, Imme Ebert-Uphoff, Sai Ravela, Hassan Ali Babaie, and Vipin Kumar. Machine learning for the geosciences: Challenges and opportunities. *IEEE Transactions on Knowledge and Data Engineering*, 2018.

[226] Xiaowei Jia, Jared D Willard, Anuj Karpatne, Jordan Read, Jacob Zwart, Michael Steinbach, and Vipin Kumar. Physics guided rnns for modeling dynamical systems: A case study in simulating lake temperature profiles. In *Proceedings of the 2019 SIAM International Conference on Data Mining*, pages 558–566. SIAM, 2019.

[227] Jordan S Read, Xiaowei Jia, Jared Willard, Alison P Appling, Jacob A Zwart, Samantha K Oliver, Anuj Karpatne, Gretchen JA Hansen, Paul C Hanson,

William Watkins, et al. Process-guided deep learning predictions of lake water temperature. *Water Resources Research*, 2019.

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

[229] Jared D Willard, Xiaowei Jia, Shaoming Xu, Michael Steinbach, and Vipin Kumar. Integrating physics-based modeling with machine learning: A survey. *arXiv preprint arXiv:2003.04919*, 2020.

[230] K Kashinath, M Mustafa, A Albert, JL Wu, C Jiang, S Esmaeilzadeh, K Azizzadenesheli, R Wang, A Chattopadhyay, A Singh, et al. Physics-informed machine learning: case studies for weather and climate modelling. *Philosophical Transactions of the Royal Society A*, 379(2194):20200093, 2021.

[231] Jordan S Read, Luke A Winslow, Gretchen JA Hansen, Jamon Van Den Hoek, Paul C Hanson, Louise C Bruce, and Corey D Markfort. Simulating 2368 temperate lakes reveals weak coherence in stratification phenology. *Ecological Modelling*, 291:142–150, 2014.

[232] Emily H Stanley, Sarah M Collins, Noah R Lottig, Samantha K Oliver, Katherine E Webster, Kendra S Cheruvelil, and Patricia A Soranno. Biases in lake water quality sampling and implications for macroscale research. *Limnology and Oceanography*, 64(4):1572–1585, 2019.

[233] Stephen B Baines, Katherine E Webster, Timothy K Kratz, Stephen R Carpenter, and John J Magnuson. Synchronous behavior of temperature, calcium, and chlorophyll in lakes of northern wisconsin. *Ecology*, 81(3):815–825, 2000.

[234] Martin Erlandsson, Ishi Buffam, Jens Fölster, Hjalmar Laudon, Johan Temnerud, Gesa A Weyhenmeyer, and Kevin Bishop. Thirty-five years of synchrony in the organic matter concentrations of swedish rivers explained by variation in flow and sulphate. *Global Change Biology*, 14(5):1191–1198, 2008.

[235] Barbara J Benson, John D Lenters&, dagger, John J Magnuson, Maryan Stubbs, Dagger, Peter J Dillon&, sect, Robert E Hecky, and Richard C Lathrop. Regional

coherence of climatic and lake thermal variables of four lake districts in the upper great lakes region of north america. *Freshwater Biology*, 43(3):517–527, 2000.

[236] Eville Gorham and Farrell M Boyce. Influence of lake surface area and depth upon thermal stratification and the depth of the summer thermocline. *Journal of Great Lakes Research*, 15(2):233–245, 1989.

[237] HG Stefan, M Hondzo, X Fang, JG Eaton, and JH McCormick. Simulated long term temperature and dissolved oxygen characteristics of lakes in the north-central united states and associated fish habitat limits. *Limnology and Oceanography*, 41(5):1124–1135, 1996.

[238] Jordan S Read and Kevin C Rose. Physical responses of small temperate lakes to variation in dissolved organic carbon concentrations. *Limnology and Oceanography*, 58(3):921–931, 2013.

[239] Kevin C Rose, Luke A Winslow, Jordan S Read, and Gretchen JA Hansen. Climate-induced warming of lakes can be either amplified or suppressed by trends in water clarity. *Limnology and Oceanography Letters*, 1(1):44–53, 2016.

[240] David M Livingstone. A change of climate provokes a change of paradigm: taking leave of two tacit assumptions about physical lake forcing. *International Review of Hydrobiology*, 93(4-5):404–414, 2008.

[241] Murugesu Sivapalan, K Takeuchi, SW Franks, VK Gupta, H Karambiri, V Lakshmi, X Liang, JJ McDonnell, EM 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.

[242] Thorsten Wagener, Murugesu Sivapalan, Peter Troch, and Ross Woods. Catchment classification and hydrologic similarity. *Geography compass*, 1(4):901–931, 2007.

[243] Luis Samaniego, Rohini Kumar, and Sabine Attinger. Multiscale parameter regionalization of a grid-based hydrologic model at the mesoscale. *Water Resources Research*, 46(5), 2010.

[244] Stacey A Archfield, Martyn Clark, Berit Arheimer, Lauren E Hay, Hilary McMillan, Julie E Kiang, Jan Seibert, Kirsti Hakala, Andrew Bock, Thorsten Wagener, et al. Accelerating advances in continental domain hydrologic modeling. *Water Resources Research*, 51(12):10078–10091, 2015.

[245] Sinno Jialin Pan, Qiang Yang, et al. A survey on transfer learning. *TKDE*, 2010.

[246] Aydin Kaya, Ali Seydi Keceli, Cagatay Catal, Hamdi Yalin Yalic, Huseyin Temucin, and Bedir Tekinerdogan. Analysis of transfer learning for deep neural network based plant classification models. *Computers and electronics in agriculture*, 158:20–29, 2019.

[247] Jun Ma, Jack CP Cheng, Changqing Lin, Yi Tan, and Jingcheng Zhang. Improving air quality prediction accuracy at larger temporal resolutions using deep learning and transfer learning techniques. *Atmospheric Environment*, 214:116885, 2019.

[248] Xing-peng Liu, Guang-quan Zhang, Jie Lu, and Ji-quan Zhang. Risk assessment using transfer learning for grassland fires. *Agricultural and forest meteorology*, 269:102–111, 2019.

[249] Joaquin Vanschoren. Meta-learning: A survey. *arXiv preprint arXiv:1810.03548*, 2018.

[250] Joaquin Vanschoren. Meta-learning. In *Automated Machine Learning*, pages 35–61. Springer, Cham, 2019.

[251] Pavel Brazdil. *Metalearning: Concepts and Systems*, pages 1–10. Springer Berlin Heidelberg, 2009.

[252] Wei Ying, Yu Zhang, Junzhou Huang, and Qiang Yang. Transfer learning via learning to transfer. In *International Conference on Machine Learning*, pages 5072–5081, 2018.

[253] R Core Team. R: A language and environment for statistical computing, 2013.

[254] Robert G Wetzel and Gene E Likens. The heat budget of lakes. In *Limnological Analyses*, pages 45–56. Springer, 2000.

[255] Gabriel Fink, Martin Schmid, Bernd Wahl, Thomas Wolf, and Alfred Wüest. Heat flux modifications related to climate-induced warming of large european lakes. *Water Resources Research*, 50(3):2072–2085, 2014.

[256] Yafang Zhong, Michael Notaro, Stephen J Vavrus, and Michael J Foster. Recent accelerated warming of the laurentian great lakes: Physical drivers. *Limnology and Oceanography*, 61(5):1762–1786, 2016.

[257] Jorge Sola and Joaquin Sevilla. Importance of input data normalization for the application of neural networks to complex industrial problems. *IEEE Transactions on nuclear science*, 44(3):1464–1468, 1997.

[258] Jerome H Friedman. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232, 2001.

[259] Ciro Castiello, Giovanna Castellano, and Anna Maria Fanelli. Meta-data: Characterization of input features for meta-learning. In *International Conference on Modeling Decisions for Artificial Intelligence*, pages 457–468. Springer, 2005.

[260] Isabelle Guyon, Jason Weston, Stephen Barnhill, and Vladimir Vapnik. Gene selection for cancer classification using support vector machines. *Machine learning*, 46(1-3):389–422, 2002.

[261] Lars Buitinck, Gilles Louppe, Mathieu Blondel, Fabian Pedregosa, Andreas Mueller, Olivier Grisel, Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort, Jaques Grobler, et al. Api design for machine learning software: experiences from the scikit-learn project. *arXiv preprint arXiv:1309.0238*, 2013.

[262] Jared D Willard, Jordan S Read, Alison P Appling, and Samantha K Oliver. Data release: Predicting water temperature dynamics of unmonitored lakes with meta transfer learning. U.S. Geological Survey - ScienceBase, 2020.

[263] Kenneth E Mitchell, Dag Lohmann, Paul R Houser, Eric F Wood, John C Schaake, Alan Robock, Brian A Cosgrove, Justin Sheffield, Qingyun Duan, Lifeng Luo, et al. The multi-institution north american land data assimilation system (nldas):

Utilizing multiple gcip products and partners in a continental distributed hydrological modeling system. *Journal of Geophysical Research: Atmospheres*, 109(D7), 2004.

[264] Gabriele Zenobi and Padraig Cunningham. Using diversity in preparing ensembles of classifiers based on different feature subsets to minimize generalization error. In *European Conference on Machine Learning*, pages 576–587. Springer, 2001.

[265] Ludmila I Kuncheva and Christopher J Whitaker. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine learning*, 51(2):181–207, 2003.

[266] Anders Krogh and Jesper Vedelsby. Neural network ensembles, cross validation, and active learning. In *Advances in neural information processing systems*, pages 231–238, 1995.

[267] Emily K Read et al. Water quality data for national-scale aquatic research: The water quality portal. *Water Resources Research*, 2017.

[268] Matthew R Hipsey, David P Hamilton, Paul C Hanson, Cayelan C Carey, Janaine Z Coletti, Jordan S Read, Bas W Ibelings, Fiona J Valesini, and Justin D Brookes. Predicting the resilience and recovery of aquatic systems: A framework for model evolution within environmental observatories. *Water Resources Research*, 51(9):7023–7043, 2015.

[269] Naoki Mizukami, Martyn P Clark, Andrew J Newman, Andrew W Wood, Ethan D Gutmann, Bart Nijssen, Oldrich Rakovec, and Luis Samaniego. Towards seamless large-domain parameter estimation for hydrologic models. *Water Resources Research*, 53(9):8020–8040, 2017.

[270] John J Magnuson, Larry B Crowder, and Patricia A Medvick. Temperature as an ecological resource. *American Zoologist*, 19(1):331–343, 1979.

[271] Michael L Jones, Brian J Shuter, Yingming Zhao, and Jason D Stockwell. Forecasting effects of climate change on great lakes fisheries: models that link habitat supply to population dynamics can help. *Canadian Journal of Fisheries and Aquatic Sciences*, 63(2):457–468, 2006.

[272] Gretchen JA Hansen, Jordan S Read, Jonathan F Hansen, and Luke A Winslow. Projected shifts in fish species dominance in wisconsin lakes under climate change. *Global change biology*, 23(4):1463–1476, 2017.

[273] Margaret A Cook, Carey W King, F Todd Davidson, and Michael E Webber. Assessing the impacts of droughts and heat waves at thermoelectric power plants in the united states using integrated regression, thermodynamic, and climate models. *Energy Reports*, 1:193–203, 2015.

[274] Ali A Ghorbani and Kiarash Owrangh. Stacked generalization in neural networks: generalization on statistically neutral problems. In *IJCNN'01. International Joint Conference on Neural Networks. Proceedings (Cat. No. 01CH37222)*, volume 3, pages 1715–1720. IEEE, 2001.

[275] Blake A Schaeffer, John Iiames, John Dwyer, Erin Urquhart, Wilson Salls, Jennifer Rover, and Bridget Seegers. An initial validation of landsat 5 and 7 derived surface water temperature for us lakes, reservoirs, and estuaries. *International Journal of Remote Sensing*, 39(22):7789–7805, 2018.

[276] Peter A. Raymond, Jens Hartmann, Ronny Lauerwald, Sebastian Sobek, Cory McDonald, Mark Hoover, David Butman, Robert Striegl, Emilio Mayorga, Christoph Humborg, Pirkko Kortelainen, Hans Dürr, Michel Meybeck, Philippe Ciais, and Peter Guth. Global carbon dioxide emissions from inland waters. *Nature*, 503(7476):355–359, November 2013.

[277] Xing Fang, Heinz G Stefan, John G Eaton, J.Howard McCormick, and Shoeb R Alam. Simulation of thermal/dissolved oxygen habitat for fishes in lakes under different climate scenarios. *Ecological Modelling*, 172(1):13–37, February 2004.

[278] Quinten Vanhellemont. Automated water surface temperature retrieval from Landsat 8/TIRS. *Remote Sensing of Environment*, 237:111518, February 2020.

[279] Alexander Y Sun and Bridget R Scanlon. How can Big Data and machine learning benefit environment and water management: a survey of methods, applications, and future directions. *Environmental Research Letters*, 14(7):073001, July 2019.

[280] Muhammed Sit, Bekir Z. Demiray, Zhongrun Xiang, Gregory J. Ewing, Yusuf Sermet, and Ibrahim Demir. A comprehensive review of deep learning applications in hydrology and water resources. *Water Science and Technology*, 82(12):2635–2670, December 2020.

[281] 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, December 2019.

[282] Dmitrii Mironov, Erdmann Heise, Ekaterina Kourzeneva, Bodo Ritter, Natalia Schneider, and Arkady Terzhevik. Implementation of the lake parameterisation scheme FLake into the numerical weather prediction model COSMO. *Boreal Environment Research*, 15:13, 2010.

[283] Muñoz Sabater. ERA5-Land hourly data from 1981 to present. *Copernicus Climate Change Service (C3S) Climate Data Store (CDS)*, 2019.

[284] Richard B Moore, Lucinda D McKay, Alan H Rea, Timothy R Bondelid, Curtis V Price, Thomas G Dewald, Craig M Johnston, and others. User's guide for the National Hydrography Dataset plus (NHDPlus) High Resolution. Technical Report 2019-1096, United States Geological Survey, 2019.

[285] Russ Rew and Glenn Davis. NetCDF: an interface for scientific data access. *IEEE computer graphics and applications*, 10(4):76–82, 1990.

[286] Luke A Winslow, Scott Chamberlain, Alison P Appling, and Jordan S Read. sbtools: A Package Connecting R to Cloud-based Data for Collaborative Online Research. *R Journal*, 8(1), 2016.

[287] Sapna Sharma, Steven C Walker, and Donald A Jackson. Empirical modelling of lake water-temperature relationships: a comparison of approaches. *Freshwater biology*, 53(5):897–911, 2008.

[288] Patricia A Soranno, Linda C Bacon, Michael Beauchene, Karen E Bednar, Edward G Bissell, Claire K Boudreau, Marvin G Boyer, Mary T Bremigan, Stephen R

Carpenter, Jamie W Carr, et al. Lagos-ne: a multi-scaled geospatial and temporal database of lake ecological context and water quality for thousands of us lakes. *GigaScience*, 6(12):gix101, 2017.

[289] Emanuel Dutra, Victor M Stepanenko, Gianpaolo Balsamo, Pedro Viterbo, Pedro Miranda, Dmitrii Mironov, and Christoph Schär. An offline study of the impact of lakes on the performance of the ECMWF surface scheme. *Boreal Environment Research*, 10:100–112, 2010.

[290] Ufficio Generale Spacio, Aero e Meteorologia, Gospodarki Wodnej, Agenzia Regionale per la Protezione, Ambientale dell Emilia-Romagna, Servizio Idro Meteo, Centro Italiano Ricerche, and Amt für GeoInformationswesen. Parameterization of lakes in numerical weather prediction. Description of a lake model. Technical report, German Weather Service, Offenbach am Main, Germany, 2008.

[291] Xiaowei Jia, Anuj Karpatne, Jared D Willard, Michael Steinbach, Jordan Read, Paul C Hanson, Hilary A Dugan, and Vipin Kumar. Physics guided recurrent neural networks for modeling dynamical systems: Application to monitoring water temperature and quality in lakes. *arXiv preprint arXiv:1810.02880*, 2018.

[292] B. J. Kreakie, S. D. Shivers, J. W. Hollister, and W. B. Milstead. Predictive Model of Lake Photic Zone Temperature Across the Conterminous United States. *Frontiers in Environmental Science*, 9:707874, October 2021.

[293] John E. Edinger, David W. Duttweiler, and John C. Geyer. The Response of Water Temperatures to Meteorological Conditions. *Water Resources Research*, 4(5):1137–1143, October 1968.

[294] S. Piccolroaz, N. C. Healey, J. D. Lenters, S. G. Schladow, S. J. Hook, G. B. Sahoo, and M. Toffolon. On the predictability of lake surface temperature using air temperature in a changing climate: A case study for Lake Tahoe (U.S.A.): On the predictability of lake surface temperature. *Limnology and Oceanography*, 63(1):243–261, January 2018.

[295] Martin Schmid and Jordan Read. Heat Budget of Lakes. In *Reference Module in Earth Systems and Environmental Sciences*. Elsevier, 2021.

[296] R. Iestyn Woolway, Ian D. Jones, Stephen C. Maberly, Jon R. French, David M. Livingstone, Donald T. Monteith, Gavin L. Simpson, Stephen J. Thackeray, Mikkel R. Andersen, Richard W. Battarbee, Curtis L. DeGasperi, Christopher D. Evans, Elvira de Eyto, Heidrun Feuchtmayr, David P. Hamilton, Martin Kernan, Jan Krokowski, Alon Rimmer, Kevin C. Rose, James A. Rusak, David B. Ryves, Daniel R. Scott, Ewan M. Shilland, Robyn L. Smyth, Peter A. Staehr, Rhian Thomas, Susan Waldron, and Gesa A. Weyhenmeyer. Diel Surface Temperature Range Scales with Lake Size. *PLOS ONE*, 11(3):e0152466, March 2016.

[297] J. Sola and J. Sevilla. Importance of input data normalization for the application of neural networks to complex industrial problems. *IEEE Transactions on Nuclear Science*, 44(3):1464–1468, June 1997.

[298] Kenneth E Mitchell, Dag Lohmann, Paul R Houser, Eric F Wood, John C Schaake, Alan Robock, Brian A Cosgrove, Justin Sheffield, Qingyun Duan, Lifeng Luo, and others. The multi-institution North American Land Data Assimilation System (NLDAS): Utilizing multiple GCIP products and partners in a continental distributed hydrological modeling system. *Journal of Geophysical Research: Atmospheres*, 109(D7), 2004.

[299] Edzer Pebesma. Simple Features for R: Standardized Support for Spatial Vector Data. *The R Journal*, 10(1):439–446, 2018.

[300] David Blodgett and Luke Winslow. ncdfgeom: 'NetCDF' Geometry and Time Series, 2019.

[301] Jeffrey Hollister. *Access Elevation Data from Various APIs*. The Comprehensive R Archive Network, 2021.

[302] Tom G Farr, Paul A Rosen, Edward Caro, Robert Crippen, Riley Duren, Scott Hensley, Michael Kobrick, Mimi Paller, Ernesto Rodriguez, Ladislav Roth, and others. The shuttle radar topography mission. *Reviews of geophysics*, 45(2), 2007.

[303] Jordan S Read, Jacob A. Zwart, Holly Kundel, Hayley R Corson-Dosch, Gretchen J.A. Hansen, Kelsey Vitense, Alison P. Appling, Samantha K. Oliver,

and Lindsay Platt. Data release: Process-based predictions of lake water temperature in the Midwest US, 2021.

[304] Blake A. Schaeffer, John Iiames, John Dwyer, Erin Urquhart, Wilson Salls, Jennifer Rover, and Bridget Seegers. An initial validation of Landsat 5 and 7 derived surface water temperature for U.S. lakes, reservoirs, and estuaries. *International Journal of Remote Sensing*, 39(22):7789–7805, November 2018.

[305] Robert M Hirsch and Laura A De Cicco. User guide to Exploration and Graphics for RivEr Trends (EGRET) and dataRetrieval: R packages for hydrologic data. Technical report, US Geological Survey, 2015.

[306] Andrew Estabrooks, Taeho Jo, and Nathalie Japkowicz. A Multiple Resampling Method for Learning from Imbalanced Data Sets. *Computational Intelligence*, 20(1):18–36, February 2004.

[307] Nathalie Japkowicz and Shaju Stephen. The class imbalance problem: A systematic study1. *Intelligent Data Analysis*, 6(5):429–449, November 2002.

[308] Tingting Pan, Junhong Zhao, Wei Wu, and Jie Yang. Learning imbalanced datasets based on SMOTE and Gaussian distribution. *Information Sciences*, 512:1214–1233, 2020.

[309] Robert Tibshirani, Jerome Friedman, and Trevor Hastie. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction.* Springer Series in Statistics. Springer, 2nd edition, 2009.

[310] S. Lloyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2):129–137, March 1982.

[311] Xiaowei Jia, Jared D Willard, Anuj Karpatne, Jordan Read, Jacob Zwart, Michael Steinbach, and Vipin Kumar. Physics guided RNNs for modeling dynamical systems: A case study in simulating lake temperature profiles. In *Proceedings of the 2019 SIAM International Conference on Data Mining*, pages 558–566. SIAM, 2019.

[312] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. *arXiv:1412.6980 [cs]*, January 2017.

[313] Alan K Betts, Daniel Reid, and Caitlin Crossett. Evaluation of the FLake model in ERA5 for Lake Champlain. *Frontiers in Environmental Science*, 8:250, 2020.

[314] Lori A Sprague, Gretchen P Oelsner, and Denise M Argue. Challenges with secondary use of multi-source water-quality data in the United States. *Water research*, 110:252–261, 2017.

[315] Hans Hersbach and DJEN Dee. ERA5 reanalysis is in production. *ECMWF newsletter*, 147(7):5–6, 2016.

[316] Kendra Spence Cheruvelil, Patricia A Soranno, Ian M McCullough, Katherine E Webster, Lauren K Rodriguez, and Nicole J Smith. LAGOS-US LOCUS v1. 0: Data module of location, identifiers, and physical characteristics of lakes and their watersheds in the conterminous US. *Limnology and Oceanography Letters*, 6(5):270–292, 2021.

[317] Timothy T Wynne, Richard P Stumpf, Michelle C Tomlinson, Gary L Fahnenstiel, Julianne Dyble, David J Schwab, and Sonia Joseph Joshi. Evolution of a cyanobacterial bloom forecast system in western Lake Erie: development and initial evaluation. *Journal of Great Lakes Research*, 39:90–99, 2013.

[318] Desmond Tromans. Temperature and pressure dependent solubility of oxygen in water: a thermodynamic analysis. *Hydrometallurgy*, 48(3):327–342, 1998.

[319] David Deslauriers, Steven R Chipps, James E Breck, James A Rice, and Charles P Madenjian. Fish bioenergetics 4.0: an R-based modeling application. *Fisheries*, 42(11):586–596, 2017.

[320] Katja Dörnhöfer and Natascha Oppelt. Remote sensing for lake research and monitoring – Recent advances. *Ecological Indicators*, 64:105–122, May 2016.

[321] Glynn C. Hulley, Simon J. Hook, and Philipp Schneider. Optimized split-window coefficients for deriving surface temperatures from inland water bodies. *Remote Sensing of Environment*, 115(12):3758–3769, December 2011.

[322] Isabella A. Oleksy, Jill S. Baron, and Whitney S. Beck. Nutrients and warming alter mountain lake benthic algal structure and function. *Freshwater Science*, pages 000–000, February 2021.

[323] Sherry L. Larkin and Charles M. Adams. Economic Consequences of Harmful Algal Blooms: Literature Summary. *EDIS*, 2013(11), December 2013.

[324] Noah R Lottig, Pang-Ning Tan, Tyler Wagner, Kendra Spence Cheruvelil, Patricia A Soranno, Emily H Stanley, Caren E Scott, Craig A Stow, and Shuai Yuan. Macroscale patterns of synchrony identify complex relationships among spatial and temporal ecosystem drivers. *Ecosphere*, 8(12):e02024, 2017.

[325] Maciej Bartosiewicz, Anna Przytulska, Jean-François Lapierre, Isabelle Laurion, Moritz F. Lehmann, and Roxane Maranger. Hot tops, cold bottoms: Synergistic climate warming and shielding effects increase carbon burial in lakes. *Limnology and Oceanography Letters*, 4(5):132–144, October 2019.

[326] Raquel Mendonça, Roger A Müller, David Clow, Charles Verpoorter, Peter Raymond, Lars J Tranvik, and Sebastian Sobek. Organic carbon burial in global lakes and reservoirs. *Nature communications*, 8(1):1–7, 2017.

[327] Cory P McDonald, Edward G Stets, Robert G Striegl, and David Butman. Inorganic carbon loading as a primary driver of dissolved carbon dioxide concentrations in the lakes and reservoirs of the contiguous United States. *Global Biogeochemical Cycles*, 27(2):285–295, 2013.

[328] Gretchen J.A. Hansen, Stephen R. Carpenter, Jereme W. Gaeta, Joseph M. Hennessy, and M. Jake Vander Zanden. Predicting walleye recruitment as a tool for prioritizing management actions. *Canadian Journal of Fisheries and Aquatic Sciences*, 72(5):661–672, May 2015.

[329] Danielle L. Massie, Gretchen J.A. Hansen, Yan Li, Greg G. Sass, and Tyler Wagner. Do lake-specific characteristics mediate the temporal relationship between walleye growth and warming water temperatures? *Canadian Journal of Fisheries and Aquatic Sciences*, 78(7):913–923, July 2021.

[330] Benjamin M. Kraemer, Orlane Anneville, Sudeep Chandra, Margaret Dix, Esko Kuusisto, David M. Livingstone, Alon Rimmer, S. Geoffrey Schladow, Eugene Silow, Lewis M. Sitoki, Rashid Tamatamah, Yvonne Vadeboncoeur, and Peter B. McIntyre. Morphometry and average temperature affect lake stratification responses to climate change: LAKE STRATIFICATION RESPONSES TO CLIMATE. *Geophysical Research Letters*, 42(12):4981–4988, June 2015.

[331] Luke A. Winslow, Jordan S. Read, Gretchen J. A. Hansen, and Paul C. Hanson. Small lakes show muted climate change signal in deepwater temperatures. *Geophysical Research Letters*, 42(2):355–361, January 2015.

[332] R Iestyn Woolway and Stephen C Maberly. Climate velocity in inland standing waters. *Nature Climate Change*, 10(12):1124–1129, 2020.

[333] R Iestyn Woolway, Benjamin M Kraemer, John D Lenters, Christopher J Merchant, Catherine M O'Reilly, and Sapna Sharma. Global lake responses to climate change. *Nature Reviews Earth & Environment*, 1(8):388–403, 2020.

[334] Catherine M. O'Reilly, Sapna Sharma, Derek K. Gray, Stephanie E. Hampton, Jordan S. Read, Rex J. Rowley, Philipp Schneider, John D. Lenters, Peter B. McIntyre, Benjamin M. Kraemer, Gesa A. Weyhenmeyer, Dietmar Straile, Bo Dong, Rita Adrian, Mathew G. Allan, Orlane Anneville, Lauri Arvola, Jay Austin, John L. Bailey, Jill S. Baron, Justin D. Brookes, Elvira de Eyto, Martin T. Dokulil, David P. Hamilton, Karl Havens, Amy L. Hetherington, Scott N. Higgins, Simon Hook, Lyubov R. Izmest'eva, Klaus D. Joehnk, Kulli Kangur, Peter Kasprzak, Michio Kumagai, Esko Kuusisto, George Leshkevich, David M. Livingstone, Sally MacIntyre, Linda May, John M. Melack, Doerthe C. Mueller-Navarra, Mikhail Naumenko, Peeter Noges, Tiina Noges, Ryan P. North, Pierre-Denis Plisnier, Anna Rigosi, Alon Rimmer, Michela Rogora, Lars G. Rudstam, James A. Rusak, Nico Salmaso, Nihar R. Samal, Daniel E. Schindler, S. Geoffrey Schladow, Martin Schmid, Silke R. Schmidt, Eugene Silow, M. Evren Soylu, Katrin Teubner, Piet Verburg, Ari Voutilainen, Andrew Watkinson, Craig E. Williamson, and Guoqing Zhang. Rapid and highly variable warming of lake surface waters around the

globe: GLOBAL LAKE SURFACE WARMING. *Geophysical Research Letters*, 42(24):10,773–10,781, December 2015.

[335] Luke A. Winslow, Gretchen J.A. Hansen, Jordan S Read, and Michael Notaro. Large-scale modeled contemporary and future water temperature estimates for 10774 Midwestern U.S. Lakes. *Scientific Data*, 4(1):170053, December 2017.

[336] Benjamin M Kraemer, Sudeep Chandra, Anthony I Dell, Margaret Dix, Esko Kuusisto, David M Livingstone, S Geoffrey Schladow, Eugene Silow, Lewis M Sitoki, Rashid Tamatamah, and others. Global patterns in lake ecosystem responses to warming based on the temperature dependence of metabolism. *Global Change Biology*, 23(5):1881–1890, 2017.

[337] Daniel P. Gillis, Charles K. Minns, and Brian J. Shuter. Predicting open-water thermal regimes of temperate North American lakes. *Canadian Journal of Fisheries and Aquatic Sciences*, pages cjfas–2020–0140, January 2021.

[338] Salim Heddam, Mariusz Ptak, and Senlin Zhu. Modelling of daily lake surface water temperature from air temperature: Extremely randomized trees (ERT) versus Air2Water, MARS, M5Tree, RF and MLPNN. *Journal of Hydrology*, 588:125130, 2020.

[339] Sebastiano Piccolroaz. Prediction of lake surface temperature using the air2water model: guidelines, challenges, and future perspectives. *Advances in Oceanography and Limnology*, 7(1), 2016.

[340] Marco Toffolon, Sebastiano Piccolroaz, Bruno Majone, Anna-Maria Soja, Frank Peeters, Martin Schmid, and Alfred Wüest. Prediction of surface temperature in lakes with different morphology using air temperature. *Limnology and Oceanography*, 59(6):2185–2202, 2014.

[341] Jemma Stachelek, Lauren K Rodriguez, Jessica Diaz Vazquez, Arika Hawkins, Ellie Phillips, Allie Shoffner, Ian M McCullough, Katelyn King, Jake Namovich, Lindsie A Egedy, Maggie Haite, Patrick J Hanly, Katherine E Webster, Kendra Spence Cheruvelil, and Patricia A Soranno. LAGOS-US DEPTH v1.0:

Data module of observed maximum and mean lake depths for a subset of lakes in the conterminous U.S. *EDI Data Portal*, December 2021.

[342] Ekaterina Kourzeneva. External data for lake parameterization in Numerical Weather Prediction and climate modeling. *Boreal Environment Research*, 15:165–177, 2010.

[343] Arthur R Cooper, Dana M Infante, Wesley M Daniel, Kevin E Wehrly, Lizhu Wang, and Travis O Brenden. Assessment of dam effects on streams and fish assemblages of the conterminous USA. *Science of the Total Environment*, 586:879–889, 2017.

[344] Andrea Ostroff, Daniel Wieferich, Arthur Cooper, and Dana Infante. Data release: National Anthropogenic Barrier Dataset (NABD) 2012 U.S. Geological Survey data release, 2013.

[345] Amina I Pollard, Stephanie E Hampton, and Dina M Leech. The promise and potential of continental-scale limnology using the US Environmental Protection Agency's National Lakes Assessment. *Limnology and Oceanography Bulletin*, 27(2):36–41, 2018.

[346] Mathis Loïc Messager, Bernhard Lehner, Günther Grill, Irena Nedeva, and Oliver Schmitt. Estimating the volume and age of water stored in global lakes using a geo-statistical approach. *Nature communications*, 7(1):1–11, 2016.

[347] Michael F Meyer, Stephanie G Labou, Alli N Cramer, Matthew R Brousil, and Bradley T Luff. The global lake area, climate, and population dataset. *Scientific data*, 7(1):1–12, 2020.

[348] JR Brett. Environmental factors, part i. temperature. *Marine ecology*, 50(0), 1970.

[349] Naoki Mizukami, Martyn P Clark, Andrew J Newman, Andrew W Wood, Ethan D Gutmann, Bart Nijssen, Oldrich Rakovec, and Luis Samaniego. Towards seamless large-domain parameter estimation for hydrologic models. *Water Resources Research*, 53(9):8020–8040, 2017.

[350] Christa D. Peters-Lidard, Martyn Clark, Luis Samaniego, Niko E. C. Verhoest, Tim van Emmerik, Remko Uijlenhoet, Kevin Achieng, Trenton E. Franz, and Ross Woods. Scaling, similarity, and the fourth paradigm for hydrology. *Hydrology and Earth System Sciences*, 21(7):3701–3713, July 2017.

[351] Jared D Willard, David Barajas-Solano, Guzel Tartakovsky, and Alexandre M Tartakovsky. Water resources time series prediction in unmonitored sites: A survey of machine learning techniquese. *arXiv:XXX.XXX [cs]*, 2023.

[352] 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, 2019.

[353] Charuleka Varadharajan, Valerie C Hendrix, Danielle S Christianson, Madison Burrus, Catherine Wong, Susan S Hubbard, and Deborah A Agarwal. BASIN-3D: A brokering framework to integrate diverse environmental data. *Computers & Geosciences*, 159:105024, 2022.

[354] James A Falcone. GAGES-II: Geospatial attributes of gages for evaluating streamflow. Technical report, US Geological Survey, 2011.

[355] PE Thornton, MM Thornton, BW Mayer, Y Wei, R Devarakonda, RS Vose, and RB Cook. Daymet: daily surface weather data on a 1-km grid for North America, version 3. ORNL DAAC, Oak Ridge, Tennessee, USA. In *USDA-NASS, 2019. 2017 Census of Agriculture, Summary and State Data, Geographic Area Series, Part 51, AC-17-A-51.* Oak Ridge National Laboratory Distributed Active Archive Center, 2016.

[356] J Brian Gray. Introduction to regression modeling. *Journal of Quality Technology*, 38(4):376, 2006.

[357] Tianqi Chen and Carlos Guestrin. XGBoost: A Scalable Tree Boosting System. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794, August 2016.

[358] K Smith. The prediction of river water temperatures/prédiction des températures des eaux de rivière. *Hydrological Sciences Journal*, 26(1):19–32, 1981.

[359] AP Mackey and AD Berrie. The prediction of water temperatures in chalk streams from air temperatures. *Hydrobiologia*, 210(3):183–189, 1991.

[360] RJ Caldwell, S Gangopadhyay, J Bountry, Y Lai, and MM Elsner. Statistical modeling of daily and subdaily stream temperatures: Application to the methow river basin, washington. *Water Resources Research*, 49(7):4346–4361, 2013.

[361] Moritz Feigl, Katharina Lebiedzinski, Mathew Herrnegger, and Karsten Schulz. Machine-learning methods for stream water temperature prediction. *Hydrology and Earth System Sciences*, 25(5):2951–2977, May 2021.

[362] Kevin E Wehrly, Travis O Brenden, and Lizhu Wang. A comparison of statistical approaches for predicting stream temperatures across heterogeneous landscapes 1. *JAWRA Journal of the American Water Resources Association*, 45(4):986–997, 2009.

[363] Anik Daigle, André St-Hilaire, Daniel Peters, and Donald Baird. Multivariate modelling of water temperature in the okanagan watershed. *Canadian water resources journal*, 35(3):237–258, 2010.

[364] Senlin Zhu and Adam P Piotrowski. River/stream water temperature forecasting using artificial intelligence models: a systematic review. *Acta Geophysica*, 68:1433–1442, 2020.

[365] Omer Sagi and Lior Rokach. Ensemble learning: A survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 8(4):e1249, 2018.

[366] BW Webb, PD Clack, and DE Walling. Water–air temperature relationships in a devon river system and the role of flow. *Hydrological processes*, 17(15):3069–3084, 2003.

[367] Sepp Hochreiter and Jürgen Schmidhuber. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780, November 1997.

[368] Kevin Faust, Sudarshan Bala, Randy Van Ommeren, Alessia Portante, Raniah Al Qawahmed, Ugljesa Djuric, and Phedias Diamandis. Intelligent feature engineering and ontological mapping of brain tumour histomorphologies by deep learning. *Nature Machine Intelligence*, 1(7):316–321, 2019.

[369] Daniel Gibert, Jordi Planes, Carles Mateu, and Quan Le. Fusing feature engineering and deep learning: A case study for malware classification. *Expert Systems with Applications*, 207:117957, 2022.

[370] Jared D Willard, Jordan S Read, Simon Topp, Gretchen JA Hansen, and Vipin Kumar. Daily surface temperatures for 185,549 lakes in the conterminous United States estimated using deep learning (1980–2020). *Limnology and Oceanography Letters*, 2022.

[371] Xiang Li, Ankush Khandelwal, Xiaowei Jia, Kelly Cutler, Rahul Ghosh, Arvind Renganathan, Shaoming Xu, Kshitij Tayal, John Nieber, Christopher Duffy, et al. Regionalization in a global hydrologic deep learning model: from physical descriptors to random vectors. *Water Resources Research*, 58(8):e2021WR031794, 2022.

[372] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

[373] Richard Liaw, Eric Liang, Robert Nishihara, Philipp Moritz, Joseph E Gonzalez, and Ion Stoica. Tune: A research platform for distributed model selection and training. *arXiv preprint arXiv:1807.05118*, 2018.

[374] Yuan Yao, Lorenzo Rosasco, and Andrea Caponnetto. On early stopping in gradient descent learning. *Constructive Approximation*, 26(2):289–315, 2007.

[375] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256, 2010.

[376] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[377] Mudasir A Ganaie, Minghui Hu, AK Malik, M Tanveer, and PN Suganthan. Ensemble deep learning: A review. *Engineering Applications of Artificial Intelligence*, 115:105151, 2022.

[378] Ulrike Grömping. Variable importance in regression models. *Wiley interdisciplinary reviews: Computational statistics*, 7(2):137–152, 2015.

[379] Christoph Molnar. *Interpretable Machine Learning*, chapter 8.5 Permutation Feature Importance. Independently published, 2022.

[380] Jangho Park, Juliane Müller, Bhavna Arora, Boris Faybishenko, Gilberto Pastorello, Charuleka Varadharajan, Reetik Sahu, and Deborah Agarwal. Long-term missing value imputation for time series data using deep neural networks. *Neural Computing and Applications*, pages 1–21, 2022.

[381] Farshid Rahmani, Kathryn Lawson, Wenyu Ouyang, Alison Appling, Samantha Oliver, and Chaopeng Shen. Exploring the exceptional performance of a deep learning stream temperature model and the value of streamflow data. *Environmental Research Letters*, 16(2):024025, 2021.

[382] Rujian Qiu, Yuankun Wang, Bruce Rhoads, Dong Wang, Wenjie Qiu, Yuwei Tao, and Jichun Wu. River water temperature forecasting using a deep learning method. *Journal of Hydrology*, 595:126016, 2021.

[383] EC Evans, Glen R McGregor, and Geoffrey E Petts. River energy budgets with special reference to river bed processes. *Hydrological processes*, 12(4):575–595, 1998.

[384] A Story, RD Moore, and JS Macdonald. Stream temperatures in two shaded reaches below cutblocks and logging roads: downstream cooling linked to subsurface hydrology. *Canadian Journal of Forest Research*, 33(8):1383–1396, 2003.

[385] Christina Tague, Michael Farrell, Gordon Grant, Sarah Lewis, and Serge Rey. Hydrogeologic controls on summer stream temperatures in the mckenzie river

basin, oregon. *Hydrological Processes: An International Journal*, 21(24):3288–3300, 2007.

[386] Barbara K Burkholder, Gordon E Grant, Roy Haggerty, Tarang Khangaonkar, and Peter J Wampler. Influence of hyporheic flow and geomorphology on temperature of a large, gravel-bed river, clackamas river, oregon, usa. *Hydrological Processes: An International Journal*, 22(7):941–953, 2008.

[387] Moloud Abdar, Farhad Pourpanah, Sadiq Hussain, Dana Rezazadegan, Li Liu, Mohammad Ghavamzadeh, Paul Fieguth, Xiaochun Cao, Abbas Khosravi, U Rajendra Acharya, et al. A review of uncertainty quantification in deep learning: Techniques, applications and challenges. *Information Fusion*, 76:243–297, 2021.

[388] Christoph Molnar. *Interpretable machine learning*. Lulu. com, 2020.

[389] Fabio Ciulla, Jared Willard, Helen Weierbach, and Charuleka Varadharajan. Interpretable classification of the contiguous united states river catchments using network science methods. In *AGU Fall Meeting Abstracts*, volume 2022, 2022.

[390] Maruan Al-Shedivat, Liam Li, Eric Xing, and Ameet Talwalkar. On data efficiency of meta-learning. In *International Conference on Artificial Intelligence and Statistics*, pages 1369–1377. PMLR, 2021.

[391] Jared D Willard, Jordan S Read, Simon N Topp, Gretchen J. A. Hansen, and Kumar, Vipin. Data release: Daily surface temperatures for 185,000 lakes in the Contiguous United States estimated using deep learning (1980-2020) (in review), 2021.

[392] Siyang Yao, Cheng Chen, Mengnan He, Zhen Cui, Kangle Mo, Ruonan Pang, and Qiuwen Chen. Land use as an important indicator for water quality prediction in a region under rapid urbanization. *Ecological Indicators*, 146:109768, 2023.

[393] Somya Sharma, Rahul Ghosh, Arvind Renganathan, Xiang Li, Snigdhansu Chatterjee, John Nieber, Christopher Duffy, and Vipin Kumar. Probabilistic inverse modeling: An application in hydrology. *arXiv preprint arXiv:2210.06213*, 2022.

[394] John C Schaake, Victor I Koren, Qing-Yun Duan, Kenneth Mitchell, and Fei Chen. Simple water balance model for estimating runoff at different spatial and temporal scales. *Journal of Geophysical Research: Atmospheres*, 101(D3):7461–7475, 1996.

[395] Feng Zhou, Youpeng Xu, Ying Chen, C-Y Xu, Yuqin Gao, and Jinkang Du. Hydrological response to urbanization at different spatio-temporal scales simulated by coupling of clue-s and the swat model in the yangtze river delta region. *Journal of Hydrology*, 485:113–125, 2013.

[396] Ye Zhu, Wen Wang, Vijay P Singh, and Yi Liu. Combined use of meteorological drought indices at multi-time scales for improving hydrological drought detection. *Science of the Total Environment*, 571:1058–1068, 2016.

[397] David Labat. Recent advances in wavelet analyses: Part 1. a review of concepts. *Journal of Hydrology*, 314(1-4):275–288, 2005.

[398] Vahid Nourani, Aida Hosseini Baghanam, Jan Adamowski, and Ozgur Kisi. Applications of hybrid wavelet–artificial intelligence models in hydrology: a review. *Journal of Hydrology*, 514:358–377, 2014.

[399] Vahid Nourani, Nasrin Nezamdoost, Maryam Samadi, and Farnaz Daneshvar Vousoughi. Wavelet-based trend analysis of hydrological processes at different timescales. *Journal of Water and Climate Change*, 6(3):414–435, 2015.

[400] John Quilty and Jan Adamowski. Addressing the incorrect usage of wavelet-based hydrological and water resources forecasting models for real-world applications with best practices and a new forecasting framework. *Journal of hydrology*, 563:336–353, 2018.

[401] Min Cheng, Qian Xu, LV Jianming, Wenyin Liu, Qing Li, and Jianping Wang. Ms-lstm: A multi-scale lstm model for bgp anomaly detection. In *2016 IEEE 24th International Conference on Network Protocols (ICNP)*, pages 1–6. IEEE, 2016.

[402] Junyoung Chung, Sungjin Ahn, and Yoshua Bengio. Hierarchical multiscale recurrent neural networks. *arXiv preprint arXiv:1609.01704*, 2016.

[403] Xiaohan Chen, Beike Zhang, and Dong Gao. Bearing fault diagnosis base on multi-scale cnn and lstm model. *Journal of Intelligent Manufacturing*, 32(4):971–987, 2021.

[404] Jan Seibert and JJ McDonnell. Gauging the ungauged basin: relative value of soft and hard data. *Journal of hydrologic engineering*, 20(1):A4014004, 2015.

[405] Tim Van Emmerik, Gert Mulder, Dirk Eilander, Marijn Piet, and Hubert Savenije. Predicting the ungauged basin: model validation and realism assessment. *Frontiers in Earth Science*, 3:62, 2015.

[406] Dara Entekhabi, Eni G Njoku, Peggy E O'Neill, Kent H Kellogg, Wade T Crow, Wendy N Edelstein, Jared K Entin, Shawn D Goodman, Thomas J Jackson, Joel Johnson, et al. The soil moisture active passive (smap) mission. *Proceedings of the IEEE*, 98(5):704–716, 2010.

[407] Petra Döll, Hannes Müller Schmied, Carina Schuh, Felix T Portmann, and Annette Eicker. Global-scale assessment of groundwater depletion and related groundwater abstractions: Combining hydrological modeling with information from well observations and grace satellites. *Water Resources Research*, 50(7):5698–5720, 2014.

[408] Zarine Kakalia, Charuleka Varadharajan, Erek Alper, Eoin L Brodie, Madison Burrus, Rosemary WH Carroll, Danielle S Christianson, Wenming Dong, Valerie C Hendrix, Matthew Henderson, et al. The colorado east river community observatory data collection. *Hydrological Processes*, 35(6):e14243, 2021.

[409] John J Magnuson, Timothy K Kratz, Barbara J Benson, et al. *Long-term dynamics of lakes in the landscape: long-term ecological research on north temperate lakes*. Oxford University Press on Demand, 2006.

[410] QiZhi He, David Barajas-Solano, Guzel Tartakovsky, and Alexandre M Tartakovsky. Physics-informed neural networks for multiphysics data assimilation with application to subsurface transport. *Advances in Water Resources*, 141:103610, 2020.

[411] Yu Zhang and Qiang Yang. An overview of multi-task learning. *National Science Review*, 5(1):30–43, 2018.

[412] Jeffrey Michael Sadler, Alison Paige Appling, Jordan S Read, Samantha Kay Oliver, Xiaowei Jia, Jacob Aaron Zwart, and Vipin Kumar. Multi-task deep learning of daily streamflow and water temperature. *Water Resources Research*, 58(4):e2021WR030138, 2022.

[413] Jutta Thielen, John Schaake, Robert Hartman, and Roberto Buizza. Aims, challenges and progress of the hydrological ensemble prediction experiment (hepex) following the third hepex workshop held in stresa 27 to 29 june 2007. *Atmospheric Science Letters*, 9(2):29–35, 2008.

[414] Magali Troin, Richard Arsenault, Andrew W Wood, François Brissette, and Jean-Luc Martel. Generating ensemble streamflow forecasts: A review of methods and approaches over the past 40 years, 2021.

[415] Y He, F Wetterhall, HL Cloke, Florian Pappenberger, M Wilson, J Freer, and G McGregor. Tracking the uncertainty in flood alerts driven by grand ensemble weather predictions. *Meteorological Applications: A journal of forecasting, practical applications, training techniques and modelling*, 16(1):91–101, 2009.

[416] Jan Seibert and Keith J Beven. Gauging the ungauged basin: how many discharge measurements are needed? *Hydrology and Earth System Sciences*, 13(6):883–892, 2009.

[417] Tadhg N Moore, Jorrit P Mesman, Robert Ladwig, Johannes Feldbauer, Freya Olsson, Rachel M Pilla, Tom Shatwell, Jason J Venkiteswaran, Austin D Delany, Hilary Dugan, et al. Lakeensemblr: An r package that facilitates ensemble modelling of lakes. *Environmental Modelling & Software*, 143:105101, 2021.

[418] Timothy DelSole, Jyothi Nattala, and Michael K Tippett. Skill improvement from increased ensemble size and model diversity. *Geophysical Research Letters*, 41(20):7331–7342, 2014.

[419] Jason Brownlee. *Better deep learning: train faster, reduce overfitting, and make better predictions*. Machine Learning Mastery, 2018.

[420] Lars Kai Hansen and Peter Salamon. Neural network ensembles. *IEEE transactions on pattern analysis and machine intelligence*, 12(10):993–1001, 1990.

[421] Leo Breiman. Bagging predictors. *Machine learning*, 24:123–140, 1996.

[422] Zachary C Lipton. The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue*, 16(3):31–57, 2018.

[423] Alejandro Barredo Arrieta, Natalia Diaz-Rodriguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador Garcia, Sergio Gil-Lopez, Daniel Molina, Richard Benjamins, et al. Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information fusion*, 58:82–115, 2020.

[424] Hakan Başağaoğlu, Debaditya Chakraborty, Cesar Do Lago, Lilianna Gutierrez, Mehmet Arif Şahinli, Marcio Giacomoni, Chad Furl, Ali Mirchi, Daniel Moriasi, and Sema Sevinç Şengor. A review on interpretable and explainable artificial intelligence in hydroclimatic applications. *Water*, 14(8):1230, 2022.

[425] Antonios Mamalakis, Elizabeth A Barnes, and Imme Ebert-Uphoff. Carefully choose the baseline: Lessons learned from applying xai attribution methods for regression tasks in geoscience. *Artificial Intelligence for the Earth Systems*, 2(1):e220058, 2023.

[426] Benjamin A Toms, Elizabeth A Barnes, and Imme Ebert-Uphoff. Physically interpretable neural networks for the geosciences: Applications to earth system variability. *Journal of Advances in Modeling Earth Systems*, 12(9):e2019MS002002, 2020.

[427] Imme Ebert-Uphoff and Kyle Hilburn. Evaluation, tuning, and interpretation of neural networks for working with images in meteorological applications. *Bulletin of the American Meteorological Society*, 101(12):E2149–E2170, 2020.

[428] Antonios Mamalakis, Elizabeth A Barnes, and Imme Ebert-Uphoff. Investigating the fidelity of explainable artificial intelligence methods for applications of

convolutional neural networks in geoscience. *Artificial Intelligence for the Earth Systems*, 1(4):e220012, 2022.

[429] Chaopeng Shen, Eric Laloy, Amin Elshorbagy, Adrian Albert, Jerad Bales, Fi-John Chang, Sangram Ganguly, Kuo-Lin Hsu, Daniel Kifer, Zheng Fang, et al. Hess opinions: Incubating deep-learning-powered hydrologic science advances as a community. *Hydrology and Earth System Sciences*, 22(11):5639–5656, 2018.

[430] Frederik Kratzert, Mathew Herrnegger, Daniel Klotz, Sepp Hochreiter, and Günter Klambauer. Neuralhydrology–interpreting lstms in hydrology. In *Explainable AI: Interpreting, explaining and visualizing deep learning*, pages 347–362. Springer, 2019.

[431] Zhitao Ying, Dylan Bourgeois, Jiaxuan You, Marinka Zitnik, and Jure Leskovec. Gnnexplainer: Generating explanations for graph neural networks. *Advances in neural information processing systems*, 32, 2019.

[432] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International conference on machine learning*, pages 3319–3328. PMLR, 2017.

[433] Pieter-Jan Kindermans, Sara Hooker, Julius Adebayo, Maximilian Alber, Kristof T Schütt, Sven Dähne, Dumitru Erhan, and Been Kim. The (un) reliability of saliency methods. *Explainable AI: Interpreting, explaining and visualizing deep learning*, pages 267–280, 2019.

[434] Ann-Kathrin Dombrowski, Christopher J Anders, Klaus-Robert Müller, and Pan Kessel. Towards robust explanations for deep neural networks. *Pattern Recognition*, 121:108194, 2022.

[435] Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature machine intelligence*, 1(5):206–215, 2019.

[436] Antonios Mamalakis, Imme Ebert-Uphoff, and Elizabeth A Barnes. Neural network attribution methods for problems in geoscience: A novel synthetic benchmark dataset. *Environmental Data Science*, 1:e8, 2022.