Referee #2 Comments

This is a meaningful manuscript that provides a thorough review of ML approaches for SWT modeling and their evaluation metrics. I believe that the current scientific community has indeed developed a broad understanding of the integration of ML into stream temperature modeling. Hence, while the manuscript presents a comprehensive overview, incorporating more in-depth insights could enhance its appeal to readers and significantly increase its contribution to the field. The review covers a wealth of content, including recent articles and other reviews, but the sections are somewhat loosely structured, with key points relatively briefly mentioned.

AUTHOR RESPONSE: We thank the referee for their time and feedback, we believe the manuscript is stronger as a result. We address specific referee comments below. For reference, we separated some referee comments into a, b, etc., to provide a more organized response. Proposed new/edited text is in BLUE.

1. For instance, in the first section (Overview: SWT Model Types), the author provides a solid overview of statistical, physical, and ML models. However, a more detailed analysis of the comparative strengths and weaknesses of physical and ML models would strengthen the discussion. The models are presented in a nearly linear developmental order in this review, but it would be beneficial to mention some points, for example, [if] physical models perform well, why ML models are adopted[?].

AUTHOR RESPONSE: The referee makes a good point with regards to the question of "if physical models perform well, why are ML models being adopted?". We have expanded the section "Artificial Intelligence Models in SWT Modeling" to discuss this:

> "In the last decade, computing advances in AI have started to offer several advantages for using machine learning (ML) in hydrology that are comparable to physically based models (Cole et al., 2014; Zhu et al., 2019; Rehana and Rajesh, 2023). In contrast to traditional physically based models, the code underlying ML models are generally open-source and publicly available allowing for near real-time accessible advances and user feedback, whereas the source code for some physically based models may be inaccessible to the public due to being privately managed (MIKE suite of models) or the model software may be publicly available but take years to publish updates (USGS MODFLOW, Simunek's HYDRUS). One advantage that has made ML increasingly appealing includes its ability to learn directly from the data (i.e., data driven), which can be useful when the underlying physics are not fully understood or are considered too complex to model accurately.
>
> Additionally, ML models are more efficient in making predictions compared to the time-intensive solvers of physically based models. ML models can also handle the challenge of scalability, that is managing large datasets and seamlessly deploying across various computer platforms and applications (Rehana and Rajesh, 2023). Air2stream, a hybrid statistical-physically based SWT model (Toffolon and Piccolroaz, 2015; Piccolroaz et al., 2016), initially outperformed earlier ML models such as Gaussian Process Regression (Zhu et al., 2019). Though in the last few years, Air2stream has had its performance matched and even exceeded by recent neural networks models (Feigl et al., 2021; Rehana and Rajesh, 2023).
>
> Finally, with computer processing power improving and the emergent field of quantum computing, there is a strong belief amongst scientists, stakeholders and the public, that using ML and by extension AI, in science applications will drive innovation to the point where natural patterns and insights not currently apparent in physical modeling will be uncovered (Varadharajan et al., 2022). Thus, while physically based models are considered tried-and-true, thereby invaluable for their interpretability and grounding in established physics, ML models have the potential for growth – where they can be used to first complement and eventually lead as powerful tools for prediction, optimization, and understanding in increasingly complex and data-rich environments."

New citation:

Toffolon, M. and Piccolroaz, S., 2015. A hybrid model for river water temperature as a function of air temperature and discharge. *Environmental Research Letters*, *10*(11), p.114011.

2. How to gain the trust of traditional model users in ML methods? (This question is inherently challenging, as model users often have preferences based on their own familiarity with certain models and may exhibit biases against alternative approaches. However, it may be worthy to acknowledge this in the review.) This discussion could extend to the choice between different ML models as well, as conclusions favoring one model over another often depend on the specific context of the study. Many conclusions are applicable only under particular circumstances, so a generalization such as "a certain model is better suited to a particular type of problem" is more appropriate.

AUTHOR RESPONSE: We agree and appreciate the referee's feedback. We address this comment in our response to referee #1 for comment #3 (copied below) titled "Future Directions", where we discuss how researchers can work to present their ML models as trustworthy. For this, we propose adding a new 'Discussion' subsection, titled 'Future Directions of SWT Modeling', with the following:

The utility of ML in hydrologic modeling has come a long way, with interest seemingly growing exponentially (Nearing et al., 2021). With the novelty of ML, it is easy to get lost in the value of how well a model performs and ignore the science, but with several decades of ML-experience, we think it necessary to urge the scientific community to purposefully use ML address physically-meaningful questions and not just create ML for the sake of creating. Given this, Varadharajan et al. (2022) laid out an excellent discussion on opportunities for advancement of ML in water quality modeling, see section 3 of publication (Varadharajan et al., 2022). Here we highlight some of the questions from Varadharajan et al. (2022) that can be considered in the context of what the objectives of the SWT community should be in the ML era, namely: 1) How do we use physical knowledge (re: heat exchange equations, radiation influence) to improve models and process understanding? Rahmani et al. (2023) coupled NNs with the physical knowledge from SNTEMP, a one-dimensional stream temperature model that calculates the transfer of energy to or from a stream segment by either heat flux equations or advection, but found that even with SNTEMP, their flexible NNs exhibited substantial variance in prediction and needed to be constrained by further multi-dimensional assessments (Rahmani et al., 2023). In short, if our use of physics in machine learning makes our models worse, we must know why.

A second question that needs addressing is 2) How do we deal with predictive uncertainty in ML used for SWT modeling? According to Moriasi et al. (2007), uncertainty analysis is the process of quantifying the level of confidence in any given model output based on five guidelines: 1) the quality and amount of observations (data), 2) the lack of observations due to poor or limited field monitoring, 3) the lack of knowledge of physical processes or operational procedures (instrumentation), 4) the approximation of our mathematical equations, and 5) the robustness of model sensitivity analysis and calibration. For example, in rainfall-runoff modeling, researchers have proposed benchmarking to examine uncertainty predictions of ML rainfall-runoff modeling (Klotz et al., 2022). For stream temperature modeling, researchers have attempted to address the role of uncertainty in deep learning model (RGCN, LSTM) prediction using the Monte Carlo Dropout (Zwart, Oliver, et al., 2023) and a unimodal mixture density network approach (Zwart, Diaz, et al., 2023).

Other questions that SWT-ML studies should consider is 3) How do we make ML models generalize better, specifically with regards to ungaged basins? And 4) How can ML models be improved to predict extremes? As ML models advance to use satellite data, include more sensor networks and/or couple with climate models, there is a logical next step toward creating generalizable models that can account for extremes. In our review, only two papers by the same group (Rahmani et al., 2020, 2023) conducted a CONUS-scale approach towards SWT-ML modeling, omitting hydrologically important regions in the southwest (CA) and southeast (FL). Recently, a satellite remote sensing paper used RF to model monthly stream temperature across the CONUS and tested for temporal (walk-forward validation), unseen and 'true' ungaged regions (Philippus et al., 2024). We have also learned that ML models such as LSTMs, generally only make predictions within the bounds

of their training data (Kratzert et al., 2019), which is a limitation for predicting extremes. Thus, we strongly urge the community to work towards ML models that generalize better and/or are more robust towards predictions of extremes.

Finally, 5) How can we build ML models such that they are seen as trustworthy and interpretable by the hydrologic community? To answer this question, we must address a technical barrier (black-box issues, data limitations, model uncertainty) and a social barrier (i.e., educated skepticism of ML due to novelty, little understanding of computer science basics and/or coding experience). If we are to incorporate ML into more of the decision-making process, it makes sense that ML must be transparent and understandable to more than just computer scientists (Varadharajan et al., 2022). For example, Topp et al. (2023) recently used explainable AI to elucidate how ML architectures affected the SWT model's spatial and temporal dependencies, and how that in turn affected the model's accuracy. Addressing this technical barrier can also be done by improving access to data, which has seen remarkable progress thanks to web repositories such as NSF-funded CUAHSI's Hydro share (CUAHSI, 2024) and GitHub (GitHub, 2024). In the United States, data access to state and locally-based data remains limited, and should be addressed. In terms of the social barrier, education about ML and ML-use is key. Societal interest in ML has thankfully also lead to a plethora of educational resources and ML walk-through videos and tutorials in TensorFlow (Abadi et al., 2015), PyTorch (Abadi et al., 2015), and Google Colab (Bison, 2019). With how fast ML-use is evolving, short communication pieces (Capuchin et al., 2019) and opinion pieces (Kratzert et al., 2024) with clear examples about an ML-issue and practical solutions could also help make ML challenges more transparent and therefore accessible to the hydrologic community-at-large.

3a. Furthermore, the author may not clearly (separately) present the generalization capabilities of ML models in temporal and spatial contexts, which is crucial for data split. The model ability of generalization over time is particularly meaningful for climate change studies, where overfitting (common for ML studies) may lead to highly unreliable projections. Spatial generalization is useful for applying models to new regions or watersheds (ungauged stream/river/watershed).

AUTHOR RESPONSE: We agree. Referee #1 made a similar comment (ref #1, comment #1A) about overfitting and having ML undergo more testing and we propose to address both comments by adding: 1) a subsection under "section 2.4 SWT Predictions using ML" on overfitting/underfitting and 2) a diagram showing initial steps to mitigate overfitting. The new text is below:

Subsection 2.4.X Overfitting and Underfitting

When a model is too complex, i.e., has too many features or too many parameters relative to the number of observations, or is forced to overextend its capabilities, i.e., make predictions with insufficient training data, the model runs the risk of overfitting (Srivastava et al., 2014). An overfitting model fits the training data "too well", capturing noise and details that provide high accuracy on a training dataset, only to perform poorly once the model encounters "unseen" data in testing/validation (Xu and Liang, 2021). Scenarios where overfitting may be temporarily acceptable are those where: 1) model development is at its preliminary stages, where the interest is in a "proof of life" concept, 2) when the objective is to identify heavily-relied on features by the model, i.e., feature importance, or 3) in highly-controlled modeling environments where the expected data will be consistently similar to the training dataset. The latter is more likely in certain industrial applications and unlikely in the changing nature of hydrology.

In contrast, underfitting occurs when a model is too simple to capture any patterns in the data, which can also lead to terrible performance in training, testing and validation. Underfitting can occur with inadequate model features, poor model complexity or when regularization techniques, (e.g., L1 or L2 regularization), are over-used, making the model too rigid and unable to respond to changes in the data. Given the propensity of machine learning models to effectively learn the training data,

underfitting is less of an issue in ML whereas overfitting can be widespread. In the following diagram, we present an example workflow to transition away from overfitting and towards generalizability. We further encourage modelers to actively transition towards making more generalizable models, which are in theory, more capable of performing well across diverse scenarios and datasets, which will become increasingly important with the persistence of climate extremes.

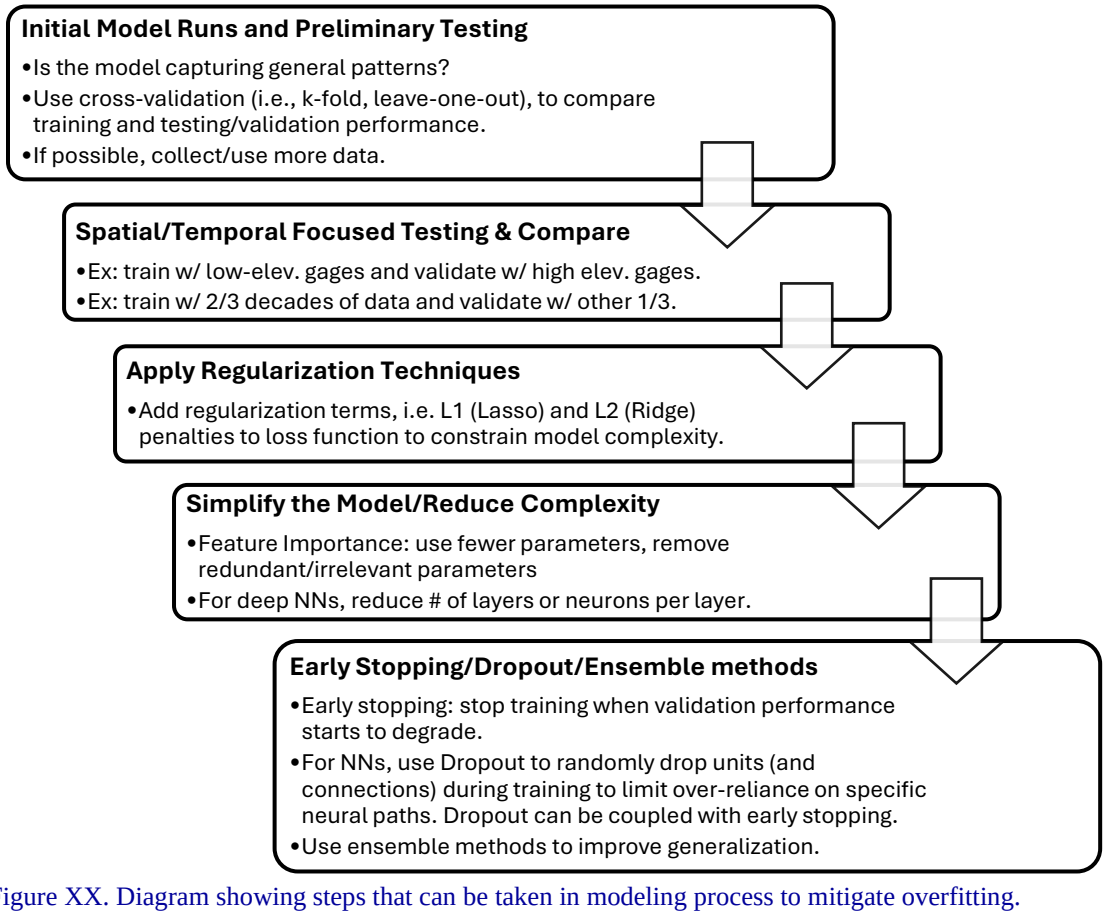

Figure XX. Diagram showing steps that can be taken in modeling process to mitigate overfitting.

With regards to generalization, we propose to address this comment and a similar one made by ref #1 (comm# 3) by adding a new Discussion subsection, titled 'Future Directions of SWT Modeling'. Below is our response in that section (full section is copied at comment #2) about generalization:

Another question that SWT-ML studies should consider is 2) How do we make ML models generalize better, specifically with regards to ungaged basins? And 3) How can ML models be improved to predict extremes? As ML models advance to use satellite data, incorporate more sensor networks and/or couple with climate models, there is a logical next step towards creating generalizable models. In our review, only two papers (Rahmani et al., 2020, 2023) conducted a CONUS-scale approach towards SWT-ML modeling, but omitted large parts of the southwest (CA) and southeast (FL), two hydrologically important regions. Recently, a satellite remote sensing RF was used to model monthly SWT across the CONUS and tested for temporal (walk-forward validation), unseen and 'true' ungaged regions, with the model architecture potentially generalizable due to it not being location-specific (Philippus et al., 2024). We have also learned that certain ML models such as LSTMs, can only predict within the bounds of their training data (Kratzert et al., 2019), which is a limitation for predicting extremes. Thus, we strongly urge the community to work towards ML models that generalize better and/or are more robust towards predictions of extremes.

3b. Additionally, the review does not systematically address the critical issue of model input selection, which is essential in ML modeling. Model inputs for SWT modeling may include hydrometeorological and physical parameters (or other attributes used in different studies), they play a role in model performance and should be discussed in this part.

AUTHOR RESPONSE: Thank you for pointing out this area in need of clarity. Referee #1, comment #4 had a similar question about model input, and we propose adding the paragraph below in response to both. Additionally, we want to note that we included in Supplementary Materials, Table S1, which contains some of the suggested data by the referee, such as: period considered, region examined, temporal resolution of SWT, spatial scale of study, and hydrometeorological parameters used for modeling.

This section will likely precede the "Local" and "Regional" subsections of 2.4 and be titled Model Inputs for ML-SWT:

> Using air temperature (AT) to better understand SWT has been considered since at least the 1960s, when Ward (1963) and Edinger et al. (1968) discussed the influence of air temperature on SWT. Since then, studies have used varying input variables (see Table S1), however, the model inputs of AT and SWT continue to be the most used in ML-modeling studies. In particular, studies have used AT from time periods outside of the known SWT record to improve model performance (Sahoo et al., 2009; Piotrowski et al., 2015; Graf et al., 2019). In addition to AT and SWT, flow discharge has been used to attempt to constrain SWT (Foreman et al., 2001; Tao et al., 2008; St-Hilaire et al., 2011; Grbić et al., 2013; Piotrowski et al., 2015; Graf et al., 2019; Qiu et al., 2020). Traditionally-used model inputs include precipitation (Cole et al., 2014; Jeong et al., 2016; Rozos, 2023), wind direction/speed (Hong and Bhamidimarri, 2012; Cole et al., 2014; Jeong et al., 2016; Kwak et al., 2016; Temizyurek and Dadaser-Celik, 2018; Abdi et al., 2021; Jiang et al., 2022), barometric pressure (Cole et al., 2014), landform attributes (Risley et al., 2003; DeWeber and Wagner, 2014; Topp et al., 2023; Souaissi et al., 2023), and many more (see Table S1).
>
> In the last few years, including the day-of-year as an input, DOY (Qiu et al., 2020; Heddam et al., 2022; Drainas et al., 2023; Rahmani et al., 2023) and humidity where available (Cole et al., 2014; Hong and Bhamidimarri, 2012; Kwak et al., 2016; Temizyurek and Dadaser-Celik, 2018; Abdi et al., 2021), have also shown to better capture the seasonal patterns of SWT (Qiu et al., 2020; Philippus et al., 2024). With improved access to remote sensing data, there has also been a notable increase of satellite products such as estimates of sky cover (Cole et al., 2014), solar radiation (Kwak et al., 2016; Topp et al., 2023; Majerska et al., 2024), sunshine per day (Drainas et al., 2023) and potential ET (Rozos, 2023; Topp et al., 2023). However, more research is needed to better understand the influence of newer model inputs on SWT (Zhu and Piotrowski, 2020).
>
> Most recently, SWT studies focused on the CONUS-scale have chosen to use as many model inputs as available, with Wade et al. (2023), a point-scale CONUS ML study using over 20 variables, while Rahmani et al. (2023) created a LSTM model and considered over 30 variables to simulate SWT. Despite the use of diverse data, the models performed only satisfactorily and were deemed not generalizable, leaving much room for improvement in CONUS-scale modeling of SWT. With the compilation of larger and larger datasets, feature importance in ML, that is the process of using techniques to assign a score to model input features based on how good the features are at predicting a target variable, can be an efficient way to improve data comprehension, model performance, and model interpretability, the latter of which can dually serve as a transparency marker of which features are driving predictions. Methods for measuring feature importance include using correlation criteria (Pearson's r, Spearman's rho), permutation feature importance (shuffling feature values, measuring decrease in model performance), linear regression feature importance (larger absolute values indicate greater importance), or if using CART/RF/gradient boosting, entropy impurity measurements can be insightful (Venkateswarlu and Anmala, 2023).

Moved from section 2.3.1, (original lines 246-253) to new section Model Inputs for ML-SWT:

> For example, one technique that can be used to improve ML model parameter selection is the

> *Least Absolute Shrinkage and Selection Operator (LASSO)*, a regression technique used for feature selection (Tibshirani, 1996). Research utilizing ML models for SWT frequency analysis at ungaged basins used the LASSO method to select explanatory variables for two ML models (Souaissi et al., 2023). The LASSO method consists of a shrinkage process where the method penalizes coefficients of regression variables by minimizing them to zero (Tibshirani, 1996). The number of coefficients set to zero depends on the adjustment parameter, which controls the severity of the penalty. Thus, the method can perform both feature selection and parameter estimation, an advantage when examining large datasets (Xu & Liang, 2021).

4. In the second section, the authors do an excellent job summarizing model evaluation metrics. However, considering that ML models are often optimized to achieve superior performance on these metrics, there is (always) a risk of overfitting. Thus, beyond focusing on metrics, the review should also highlight the importance of more rigorous evaluation to further assess generalization ability. For instance, if a SWT model is built to run climate change scenarios, additional testing and more rigorous designs are essential to evaluate the model's ability to generalize over time. For robust long-term predictions, the model is supposed to maintain robust predictive performance in completely unseen periods, rather than being limited to a specific temporal range.

AUTHOR RESPONSE: We agree. This comment has similar themes to our response to #3a regarding overfitting and highlighting the need for generalization, please see comment #3a for a full response.

For the comment regarding having ML undergo more rigorous testing, we propose adding the following discussion for more rigorous testing for MLs. We added a few sentences (blue is new) to the Discussion subsection titled “ML as Knowledge Discovery” where we urge for TUURTs (Temporal, Unseen, Ungaged Region Tests)’:

> Our review finds that ML studies examining SWT have been conducted from a computational perspective, one with a focus on comparing techniques and performance metrics as opposed to explaining the nature of SWT dynamics or influencing processes. While it is understandable that not every ML-SWT paper aims to explain physical processes, we think the SWT community should come together and agree on a baseline of tests that all ML-SWT models should undergo for model robustness and transferability. Along these lines, we urge consideration of TUURTs (temporal, unseen, ungaged region tests) for future ML-SWT models as a helpful step towards not only better modeling practices but also increased model transparency and robustness. For this, we clarify that testing for “unseen” cases means testing only within the developmental dataset, whereas testing for “ungaged” cases means testing for new sites that have not been previously seen by the model at all. Recent ML-SWT studies have only applied one or two of the tests, but not all three (Topp et al., 2023; Hani et al., 2023, Souassi et al., 2023). Siegel et al. (2023), a non-ML SWT paper, tested for ungaged and unseen data but did not perform a temporal test. A relatively new study, Philippus et al. (2024), appears to be the only published SWT-ML study that purposefully applied TUURTs with some success.

Overall, this review is informative and well-researched, and with more refined organization and deeper exploration of these key issues, it could make a substantial contribution to the field of SWT research.

AUTHOR RESPONSE: Thank you! This would certainly not be possible without the insightful feedback from referees.