Advancing Climate Science through Machine Learning-Driven Prediction of Ocean Heat Streams

By Saman Khadivar

The Imperative for Predictive Oceanography in Climate Science

The world's oceans are the primary flywheel of the global climate system. Their immense volume and thermal capacity allow them to absorb, store, and transport vast quantities of heat, governing weather patterns, stabilizing temperatures, and sequestering a significant portion of anthropogenic carbon dioxide emissions. A critical mechanism in this process is the biological carbon pump (BCP), where microscopic phytoplankton absorb atmospheric CO2 and, upon dying, sink into the deep ocean as "marine snow," transferring an estimated 10 billion tonnes of carbon annually—an amount comparable to yearly fossil fuel emissions.[1, 2] Understanding and predicting the dynamics of this system, particularly the movement of heat and the efficiency of carbon transport, is therefore fundamental to modern climate science. However, the ability to accurately forecast oceanic phenomena is increasingly challenged by the very changes it seeks to model. Extreme events like marine heatwaves (MHWs) are becoming more frequent and intense, with profound and often devastating impacts on marine ecosystems, fisheries, and coastal communities.[3] The prediction of these events is essential for developing effective mitigation and adaptation strategies to manage their significant social and ecological risks.[3, 4]

The pursuit of predictive oceanography is fraught with significant, deeply-rooted challenges that limit the efficacy of traditional methods. These challenges are not merely technical but also systemic, stemming from the nature of data collection in the marine environment and the computational demands of physics-based modeling.

First, oceanographic data is fundamentally sparse, noisy, and often inconsistent. The ocean is a physically "noisy" place, making the identification of clear patterns from observational data incredibly challenging.[5, 6, 7] This problem is exacerbated by data sparsity; vast regions of the ocean, particularly in the polar seas during winter, remain critically under-sampled.[6, 8] Furthermore, a historical lack of standardized measurement protocols across different research groups and projects has introduced significant variations into the data record.[6, 9] These methodological differences can mask or even create false trends, making it difficult to draw definitive conclusions about the drivers of critical processes. For instance, researchers have been unable to conclusively determine the role of temperature in driving the efficiency of the BCP

precisely because these data variations obscure any potential physical signal.[6] This points to a systemic issue within the oceanographic community, where historically "insular" disciplines have used distinct vocabularies and methods to describe and quantify data, hindering large-scale synthesis.[9, 10]

Second, all ocean data are inherently uncertain. A fundamental distinction exists between the error of an estimate (the difference from a true value, which can never be truly known) and the uncertainty (a characterization of the possible error).[10] Inadequate treatment of this uncertainty persists throughout the research community, decreasing confidence in many datasets. This is particularly evident in the process of calibrating and validating (cal/val) satellite measurements with in-situ observations. A "representation error" can arise because a pointwise in-situ measurement of sea surface temperature may not agree with a satellite measurement that represents an average over a larger ground footprint, complicating the process of creating a cohesive and reliable data product.[10] Seminal community documents on ocean observing systems have historically lacked recommendations for quantifying and communicating these uncertainties, highlighting a systemic gap.[10]

Third, traditional ocean forecasting models, while powerful, face significant computational limitations. These models are built on a data assimilation (DA) framework, which combines a physical model—a complex algorithm for solving a set of partial differential equations (PDEs)—with the ingestion of observational data to produce a forecast.[11] While these physics-based models can produce high-resolution outputs, they are computationally intensive.[7, 12, 13] This high computational cost limits their speed and the feasibility of running large forecast ensembles, which are crucial for capturing the range of possible outcomes. Moreover, the chaotic nature of the climate system means that the accuracy of these models degrades significantly for forecasts with lead times beyond approximately one week.[14]

Finally, the enhancement of the global ocean observing system, which is necessary to feed these models with better data, is constrained by systemic issues of "flat funding and limited cooperation among present and potential users".[14] This creates a difficult cycle: models need more and better data to improve, but the infrastructure to collect that data is under-resourced.[15] It is clear that while technology is a key part of the solution, its success is causally linked to the health of the data ecosystem that supports it. Any advanced modeling effort must therefore be developed in concert with community-wide efforts toward data standardization, improved data sharing, and coordinated observing strategies.[8]

In this context, the rapid acceleration of artificial intelligence (AI) and machine learning (ML) applications in Earth system science offers a transformative opportunity.[16, 11] ML presents a fundamentally different, "bottom-up" approach to modeling. Instead of starting with the laws of physics, ML models learn empirical relationships directly from vast quantities of data.[17] This data-driven paradigm has the potential to create faster and more efficient forecasting tools that can complement, and in some cases even surpass, the capabilities of traditional physics-based models, heralding a new era of predictive oceanography.[3]

The Scientific Machine Learning (SciML) Paradigm for Ocean Forecasting

The application of machine learning to complex scientific domains like oceanography requires a more rigorous framework than that used for standard commercial applications. This has given rise to the field of Scientific Machine Learning (SciML), which seeks to integrate domain knowledge with data-driven methods to create models that are not only predictive but also robust, interpretable, and physically consistent.[16] SciML moves beyond the "black-box" paradigm, aiming to build tools that augment scientific discovery. According to a framework defined by the U.S. Department of Energy, SciML is characterized by six key elements: (i) the incorporation of domain knowledge, such as physical principles and constraints; (ii) interpretability, where model predictions can be supported by causal explanations; (iii) robustness and reliability, essential for high-stakes decisions; (iv) data-intensiveness, involving the effective use of high-dimensional and uncertain data; (v) the enhancement of traditional modeling and simulation, for example, through model acceleration; and (vi) support for intelligent automation in the scientific workflow.[16, 18, 19] This paradigm provides a critical lens through which to evaluate and guide the development of ML for ocean forecasting.

The evolution of ML architectures applied to weather and ocean prediction demonstrates a clear progression towards models that better embody SciML principles by more effectively representing the underlying physics and geometry of the Earth system. This is not a random walk through different algorithms but a directed search for architectures that can overcome the limitations of their predecessors.

- **Convolutional Neural Networks (CNNs):** As one of the first serious endeavors in this space, CNNs were adapted from their success in image processing. They use convolutional filters to learn local spatial features and patterns from gridded data. CNNs have been used to emulate weather forecast models and have even

been successful in creating coupled atmosphere-ocean emulators that can produce stable, thousand-year-long climate simulations competitive with traditional climate models.[11] In a study of the Mediterranean Sea, CNNs were among the ML techniques shown to predict sea surface temperature (SST) and marine heatwave occurrences with reasonable accuracy.[3]

- **Graph Neural Networks (GNNs):** Recognizing that a flat grid is an imperfect representation of a spherical planet, researchers turned to GNNs. Models like GraphCast, a leading emulator for medium-range weather forecasts, represent the Earth as a graph of interconnected points. This is a more natural and flexible topology for global-scale phenomena, allowing the model to learn relationships that are not constrained by a rigid grid structure.[11]
- **Transformers:** Originally developed for natural language processing, Transformer architectures have proven revolutionary due to their "attention mechanism," which allows them to weigh the importance of different inputs when making a prediction. This has made them exceptionally powerful for modeling long-range dependencies in sequential data. Transformers now form the backbone of some of the most advanced atmospheric emulators, including Pangu-Weather and FuXi.[11]
- **Fourier Neural Operators (FNOs):** FNOs represent a significant conceptual leap. They are designed to learn operators in the frequency domain using Fourier transforms, which makes them inherently well-suited to learning the solutions of partial differential equations (PDEs) that govern physical systems. This approach allows them to be resolution-independent, or "mesh-independent".[11] Variants like the Spherical FNO (SFNO) explicitly incorporate the Earth's spherical geometry and symmetries, further embedding physical domain knowledge directly into the model architecture. The use of SFNOs has recently been extended to coupled atmosphere-ocean modeling for seasonal prediction, demonstrating their power and flexibility.[11]

This architectural evolution highlights a core tenet of SciML: the deliberate embedding of domain knowledge. The progression from generic grid-based CNNs to more physically and geometrically informed architectures like GNNs and FNOs shows a field maturing from applying general-purpose ML to developing highly specialized scientific tools.

A key opportunity within the SciML paradigm is the creation of hybrid models that integrate ML with traditional data assimilation (DA) frameworks. There is a strong conceptual correspondence between the two fields, as both seek to produce an optimal state estimate from a combination of a model and observations.[11] This

synergy allows for several powerful integration strategies. ML models can be used as computationally efficient "surrogate models" to replace specific, expensive components of a larger physical model, such as the parameterization of subgrid-scale (SGS) turbulent oceanic processes.[11] This can dramatically accelerate the overall simulation without sacrificing critical physical detail. More broadly, ML can be embedded directly within the DA workflow to improve the process of ingesting and weighting observational data, potentially leading to more accurate and reliable forecasts.[11, 20] This hybrid approach leverages the respective strengths of both paradigms: the rigorous physical grounding of traditional models and the ability of ML to learn complex patterns and relationships from data.

A Deep Dive into PyTorch-Based Time-Series Forecasting Architectures

The development of powerful and accessible deep learning frameworks is a key enabler for the application of SciML to oceanography. PyTorch has emerged as a leading framework, favored by the research community for its flexibility and intuitive design. Building on this foundation, a new generation of high-level libraries has appeared, designed to streamline the process of building, training, and evaluating complex forecasting models. These libraries signal a maturation of time-series forecasting as a distinct sub-discipline of machine learning, where the focus is shifting from reinventing fundamental components to leveraging production-ready tools that codify best practices.

The pytorch-forecasting package is a prime example of this trend. It is a PyTorch-based library built on top of pytorch-lightning, a tool that further abstracts away boilerplate code for training on different hardware (CPUs, single or multiple GPUs) and automates tasks like logging.[21, 22] The value proposition of a library like pytorch-forecasting is immense for a research team. It provides a high-level API that handles many of the most tedious and error-prone aspects of a real-world forecasting project. This includes a specialized TimeSeriesDataSet class that manages variable transformations, the handling of missing values, randomized subsampling for training, and the use of multiple history lengths—all common requirements when working with complex, real-world data like oceanographic measurements.[21, 22] By providing a base model class with built-in training loops, TensorBoard logging, and generic visualizations (e.g., actuals vs. predictions), the library allows researchers to bypass boilerplate engineering and focus their efforts on the unique scientific challenges of their problem: data curation, feature engineering, and novel model architecture design.

This ecosystem provides out-of-the-box access to a range of state-of-the-art neural network architectures that have been enhanced for real-world deployment and, critically for SciML, often come with in-built interpretation capabilities. Key models available within the library include:

- **Temporal Fusion Transformers (TFT):** This architecture, based on the powerful Transformer model, is specifically designed for interpretable multi-horizon time series forecasting.[23, 24] It uses attention mechanisms to learn long-term dependencies while incorporating specialized components to select relevant features and produce prediction intervals.[23] Its proven ability to outperform established baselines like Amazon's DeepAR by 36-69% in benchmark studies makes it a compelling choice for generating reliable, long-range ocean forecasts.[21]
- **N-BEATS and N-HiTS:** The N-BEATS (Neural Basis Expansion Analysis for Interpretable Time Series Forecasting) architecture introduced a novel deep learning approach that proved highly successful, outperforming all other methods in the prestigious M4 forecasting competition.[21, 25, 26, 27, 28] It works by decomposing the time series into a set of interpretable basis functions.[28] The successor model, N-HiTS (Neural Hierarchical Interpolation for Time Series Forecasting), improves upon this by supporting covariates (external variables) and demonstrating consistently better performance, particularly for long-horizon forecasting, which is a key requirement for climate-scale predictions.[21]
- **DeepAR:** This model, developed by Amazon, is a popular and robust algorithm for probabilistic forecasting using autoregressive recurrent networks.[21, 29, 30, 31, 32] It does not predict a single value but rather the parameters of a probability distribution (e.g., mean and standard deviation of a Gaussian), from which future values can be sampled.[29] This makes it an excellent baseline for any project that requires not just a point forecast but also a quantifiable estimate of uncertainty.

Beyond the models themselves, the pytorch-forecasting ecosystem provides essential tools for the complete research lifecycle. It integrates with optuna for sophisticated hyperparameter tuning, a critical step for achieving optimal model performance.[21] It also provides specialized loss functions, such as the MQF2 (Multivariate Quantile Loss), which are necessary for training models to produce accurate multivariate quantile forecasts—a sophisticated form of probabilistic forecasting.[21, 22] The availability of these production-grade tools dramatically lowers the barrier to entry for conducting high-quality deep learning research, democratizing access to powerful techniques that were previously the domain of highly specialized teams.

Case Study: The FuXi-Ocean Model for High-Resolution, Sub-Daily Prediction

The FuXi-Ocean model represents a landmark achievement in the application of SciML to oceanography, showcasing how a purpose-built deep learning architecture can overcome the core limitations of both traditional numerical models and more generic data-driven approaches. It stands as the first data-driven global ocean forecasting model to achieve six-hourly temporal resolution at an eddy-resolving 1/12° spatial resolution, with predictions extending to a depth of 1500 meters.[5, 33] This capability directly addresses two major challenges: the immense computational expense of traditional physics-based models and the tendency of previous deep learning models to operate at a coarse daily resolution, where they struggle with the accumulation of errors over sequential predictions.[5, 33]

The architecture of FuXi-Ocean is a masterclass in SciML design, integrating domain knowledge at multiple levels to create a robust and powerful forecasting system. Its design is autoregressive, meaning it predicts a future state based on a sequence of previous states, a standard approach for time-series forecasting.[5] However, it incorporates several key innovations to tailor this approach to the specific physics of the ocean.

First, the model employs a **context-aware feature extraction** module. This is not a generic encoder; it is designed to be sensitive to the spatiotemporal context of the ocean. The pipeline uses a shared encoder with convolutional layers for efficient patch embedding from the input data. Critically, this is augmented by a "prior information network" that processes external spatial data (like geographic coordinates and bathymetry) and temporal information (like diurnal and seasonal cycles).[5] The features from this prior network are then used to modulate the weights of the main encoder.[5] This mechanism, described by the equation $F_t = \text{Norm}(\text{Conv}(X_t, W \odot F_S))$, where W are learnable weights and F_S are the contextual features, allows the model to learn region-specific patterns.[5, 6] It can, for example, become more sensitive to the dynamics of boundary currents or the seasonal changes in the mixed layer depth in different parts of the globe, embedding geographical and physical knowledge directly into the feature learning process.[5]

Second, the predictive core of the model uses **stacked attention blocks**, leveraging the power of the Transformer architecture that has proven so effective in other large-scale modeling efforts.[34] This allows the model to capture complex, long-range spatiotemporal dependencies in the data.

The most significant innovation, however, is the **Mixture-of-Time (MoT) module**. This component is a direct and intelligent architectural solution to the well-known problem of error accumulation in autoregressive forecasting. In a naive autoregressive model, a small error in the prediction at time t+1 becomes part of the input for predicting time t+2, leading to a cascade where errors compound over the forecast horizon. The MoT module mitigates this by adaptively integrating predictions from multiple temporal contexts.[5] It learns a "variable-specific reliability," meaning it can assess how trustworthy a prediction is for a given variable (e.g., temperature vs. current) over a given time step.[5, 13, 25] This allows the model to dynamically weigh information from different forecast horizons, preventing it from blindly trusting its most recent, potentially error-prone prediction. It is a sophisticated, learned error-correction mechanism that demonstrates a deep understanding of both the failure modes of deep learning and the physical nature of the forecasting problem.

Validated against both reanalysis and observational datasets, FuXi-Ocean has demonstrated superior performance compared to traditional numerical forecasting models at sub-daily intervals.[5, 31] Its success provides a powerful proof of concept, showing that data-driven models can be a viable, and in some cases superior, alternative for operational oceanography. The implications are significant, with potential applications ranging from optimizing shipping routes and managing marine resources to enhancing climate research and environmental monitoring.[35, 31]

Data Sources, Performance Evaluation, and Visualization

A successful machine learning project is built on three pillars: high-quality data for training, rigorous metrics for evaluation, and insightful visualizations for analysis. For the domain of ocean heat stream prediction, a wealth of public data and established evaluation practices are available to guide development.

Publicly Available Oceanographic Datasets

The primary data resource for training and validating ocean models is the **World Ocean Database (WOD)**, curated by the U.S. National Centers for Environmental Information (NCEI). It is the world's largest collection of uniformly formatted, quality-controlled, and publicly available ocean profile data.[6, 36] The WOD is an invaluable asset for machine learning because it aggregates over 20,000 separate datasets into a single, cohesive resource.[6, 11] Its temporal coverage is remarkable, spanning from Captain Cook's voyage in 1772 to the modern era of autonomous Argo floats, making it uniquely suited for studying long-term climate signals.[6] The database contains profiles for a comprehensive set of variables essential for heat stream prediction, including Temperature, Salinity, Oxygen, and Nutrients.[6] Data can

be accessed through the flexible **WODselect** web interface, which allows users to query by geographic area, date range, and variable, and download custom subsets in user-friendly formats like CSV and netCDF.[6, 14]

Another critical dataset, particularly for sea surface phenomena, is the **Optimum Interpolation Sea Surface Temperature (OISST)** dataset from NOAA. This is a high-resolution (0.25° grid) daily dataset that is widely used as a benchmark and training source for SST prediction models.[4, 37, 38, 39]

Performance Metrics for Model Evaluation

To quantitatively assess the performance of a predictive model, a standard set of statistical indicators is used across the field. These metrics provide an objective measure of the model's accuracy and skill.

- **Root Mean Squared Error (RMSE):** This is the most common metric for regression tasks like temperature prediction. It measures the square root of the average of the squared differences between predicted and actual values, and is reported in the units of the variable (e.g., °C). A lower RMSE indicates a better fit. Studies have shown that ML models can significantly outperform traditional baselines; for instance, one study on stream temperature prediction found that ML models achieved a mean RMSE of 0.55°C, compared to 1.55°C for linear regression and 0.98°C for the air2stream model.[40] For global SST prediction, deep learning models have achieved RMSE values ranging from 0.27°C to 0.65°C for 10-day forecasts.[41]
- **Mean Absolute Error (MAE):** This metric measures the average of the absolute differences between predicted and actual values. It is less sensitive to large outliers than RMSE.[42, 18, 24]
- **Determination Coefficient (R2):** This metric indicates the proportion of the variance in the dependent variable that is predictable from the independent variable(s). It provides a measure of how well the model explains the variability of the data.[42]
- **Nash–Sutcliffe Efficiency (NSE):** Often used in hydrology and climate modeling, the NSE is a normalized statistic that compares the residual variance of the model to the variance of the observed data. A value of 1 corresponds to a perfect match, while a value of 0 indicates the model is only as good as the mean of the observed data.[42]

The following table synthesizes reported performance metrics from the literature, providing a clear comparison of different modeling approaches.

Model Type	Key Metric	Reported Value (°C)	Lead Time	Dataset Used	Source
Linear Regression	Mean RMSE	1.55	Daily	Austrian Catchments	[40]
air2stream	Mean RMSE	0.98	Daily	Austrian Catchments	[40]
FNN, XGBoost	Mean RMSE	0.55	Daily	Austrian Catchments	[40]
MR-EDLSTM	RMSE	0.2712 - 0.6487	1-10 days	NOAA OISST V2	[41]
MR-EDConv LSTM	RMSE	0.3195 - 0.6722	1-10 days	NOAA OISST V2	[41]
LFS (Numerical Model)	RMSE	0.522 - 0.647	1-7 days	LICOM	[41]

Visualization Strategies for Insight and Communication

Beyond quantitative metrics, effective visualization is crucial for understanding model behavior, diagnosing problems, and communicating results. A comprehensive visualization suite should include:

- **Spatial Error Maps:** Plotting heatmaps of the prediction error (Predicted - True) across the geographic domain. These maps are essential for identifying regional biases, such as whether the model performs worse in high-latitude regions or along complex coastlines.[41]
- **Time-Series Metric Plots:** Plotting key metrics like area-averaged RMSE over the course of a long evaluation period. This can reveal if a model's performance degrades over time or if it has seasonal dependencies (e.g., performing worse during summer months).[41]
- **Error Distribution Histograms:** Creating histograms of the prediction error provides insight into the model's bias and variance. An ideal model would have errors centered at zero with a tight, Gaussian distribution. Heavy tails in the distribution would indicate the model makes occasional very large errors.[41]
- **Predicted vs. Actual Scatter Plots:** A simple scatter plot of predicted values against true values is a powerful diagnostic tool. A perfect model would show all points lying on the y=x line. Deviations from this line can reveal systematic over-

or under-prediction.

Strategic Recommendations for Implementation

Developing a robust, state-of-the-art predictive model for ocean heat streams requires a cohesive strategy that integrates data management, phased model development, and a rigorous evaluation framework. The following recommendations provide an actionable roadmap for such a project.

Data Strategy: Curation and Feature Engineering

The foundation of any successful machine learning model is the data it is trained on. Given the known challenges of oceanographic data, a meticulous data strategy is paramount.

1. **Establish a Robust Data Pipeline:** The project should begin by building a data ingestion and processing pipeline centered on authoritative public datasets. The **World Ocean Database (WOD)** should be the primary source for subsurface data due to its comprehensive nature and quality control.[6, 36] For surface phenomena, the **NOAA OISST** dataset provides a high-quality, gridded product suitable for benchmarking.[4, 37]
2. **Prioritize Quality Control:** A dedicated phase of the project must focus on data cleaning and quality control. This involves implementing protocols to handle missing values, flag outliers, and harmonize data from different sources to mitigate the known issues of noise and inconsistent measurement protocols that can plague historical ocean data.[5, 6, 9, 10]
3. **Implement Informed Feature Engineering:** The selection of input features should be guided by prior research. Studies have shown that a minimal set of highly predictive features includes satellite-observable quantities like Sea Surface Height (SSH), Sea Surface Temperature (SST), and surface wind stress components. Critically, **geographic information**—such as latitude/longitude coordinates or derived quantities like the Coriolis parameter—is an essential input for enabling the model to learn location-dependent physics.[17]

Modeling Strategy: A Phased, SciML-Informed Approach

A phased approach to model development allows for rapid initial progress while building towards a state-of-the-art solution.

1. **Phase 1 - Baselining:** The project should commence by establishing strong performance baselines. Leveraging a high-level library like pytorch-forecasting will enable the rapid deployment of well-understood and robust models such as DeepAR or standard LSTM networks.[21, 22] This phase will validate the data

pipeline and provide the initial metrics against which all future, more complex models will be compared.

2. **Phase 2 - Advanced Architectures:** Once baselines are established, the project should progress to more sophisticated, off-the-shelf architectures available in the ecosystem. Models like the **Temporal Fusion Transformer (TFT)** or **N-HiTS** should be implemented. These models offer superior performance on complex time-series data and provide greater interpretability, aligning with the principles of SciML.[21, 23, 25]
3. **Phase 3 - Custom Architecture Development:** To achieve breakthrough performance, the final phase should focus on developing a custom PyTorch architecture. This model should be inspired by the design principles of leading models like **FuXi-Ocean**.[5] The key objective will be to embed domain knowledge directly into the architecture. This includes designing a context-aware feature encoder that can utilize static information like bathymetry and developing a core predictive module, analogous to the Mixture-of-Time (MoT) concept, specifically designed to handle multi-scale temporal dynamics and mitigate the problem of cumulative error in long-range forecasting.

Evaluation and Validation Framework

A rigorous evaluation framework is necessary to track progress, diagnose model weaknesses, and ensure the final product is reliable.

1. **Implement a Comprehensive Metrics Suite:** The evaluation framework must track the standard performance metrics defined in Section 5, including RMSE, MAE, R^2, and NSE.[40, 42] These should be calculated at various forecast lead times to understand how model skill degrades over time.
2. **Utilize an Insight-Oriented Visualization Dashboard:** The project should maintain a dashboard of the key visualizations outlined in Section 5.3. Spatial error maps, time-series metric plots, and error distribution histograms are not merely for reporting; they are essential diagnostic tools for gaining a deep understanding of model behavior and identifying systematic biases.[41]
3. **Benchmark Against Established Systems:** Where possible, model performance should be compared not only against internal baselines but also against the forecasts produced by established, operational physics-based models (such as the LFS system referenced in one study).[41] This provides a real-world measure of the data-driven model's skill.

Final Synthesis: A Roadmap to a Predictive Heat Stream Model

In conclusion, the path to developing a cutting-edge predictive model for ocean heat streams is a tripartite strategy. It begins with a **data-centric foundation** that

prioritizes the curation of high-quality, multi-modal data and informed feature engineering. It proceeds with a **phased modeling strategy** that leverages high-level libraries for rapid baselining before progressing to the development of domain-specific custom architectures that embody SciML principles. Finally, the entire process must be underpinned by a **rigorous evaluation framework** that uses a combination of quantitative metrics and insightful visualizations to provide a deep, actionable understanding of model performance. By following this roadmap, a research organization can efficiently navigate the complexities of data-driven oceanography and build a tool with the potential to significantly advance climate science.

References

- Baker, N., et al. (2019). *Basic Research Needs for Scientific Machine Learning*. U.S. Department of Energy, Office of Science.
- Bonino, G., et al. (2024). Machine learning methods to predict sea surface temperature and marine heatwave occurrence: a case study of the Mediterranean Sea. *Ocean Science, 20*(2), 417-432.
- Huang, Q., et al. (2025). FuXi-Ocean: A Global Ocean Forecasting System with Sub-Daily Resolution. *arXiv preprint arXiv:2506.03210*.
- Lim, B., et al. (2021). Temporal Fusion Transformers for Interpretable Multi-horizon Time Series Forecasting. *International Journal of Forecasting, 37*(4), 1748-1764.
- Oreshkin, B. N., et al. (2020). N-BEATS: Neural basis expansion analysis for interpretable time series forecasting. *International Conference on Learning Representations*.
- Salinas, D., et al. (2020). DeepAR: Probabilistic forecasting with autoregressive recurrent networks. *International Journal of Forecasting, 36*(3), 1181-1191.
- Weyn, J. A., et al. (2019). Can Machines Learn to Predict Weather? Using Deep Learning to Predict Gridded 500-hPa Geopotential Height From Historical Weather Data. *Journal of Advances in Modeling Earth Systems, 11*(9), 2680-2693.
- Zhu, J., et al. (2023). Short-Term Prediction of Global Sea Surface Temperature Using Deep Learning Networks. *Journal of Marine Science and Engineering, 11*(7), 1352.