Physics-Informed Neural Operators with Multi-Method Explainability for Ocean Temperature Prediction

Anonymous submission

Abstract

Accurate ocean temperature prediction is critical for climate modeling and marine ecosystem management, yet current machine learning approaches often lack physical consistency and interpretability. Traditional physics-based models provide interpretable predictions but struggle with computational efficiency, while modern deep learning achieves impressive accuracy but lacks explainability. However, existing neural operator studies primarily focus on synthetic datasets without systematic comparison of physics-informed constraints and multi-method interpretability validation on real observational data. We present a comprehensive study comparing four neural architectures (DeepONet, physics-informed DeepONet, FNO, LSTM) for ocean temperature prediction from real Argo float measurements. Our framework incorporates heat equation constraints through physics-informed loss functions and applies four complementary XAI methods (Integrated Gradients, Saliency Maps, DeepLIFT, GradientSHAP) with rigorous statistical validation through 10-seed Deep Ensembles. Physics-informed DeepONet achieves the best performance with 17.6% error reduction (RMSE: 0.718±0.026°C vs. 0.871±0.051°C, p¡0.001) and 48% variance reduction, demonstrating that PDE constraints meaningfully enhance neural operators on real-world data. Multi-method XAI analysis reveals that pressure and salinity measurements dominate predictions (attribution magnitudes 0.3–0.7, indicating primary predictive importance) across all architectures, with physics constraints increasing salinity sensitivity by 6.3%, while layer conductance shows LSTM's final layer exhibits 60× higher importance than intermediate layers. Our results establish a methodological framework combining physics-informed operator learning with comprehensive explainability for trustworthy scientific machine learning applications.

Introduction

Motivation

Ocean temperature is a fundamental variable in climate science, influencing global circulation patterns, marine ecosystems, and weather systems. Accurate prediction of ocean thermal structure from sparse in-situ measurements remains challenging due to the complex interplay of physical processes including advection, diffusion, and stratification. While traditional physics-based models provide interpretable predictions grounded in conservation laws, they often struggle with computational efficiency and parameter uncertainty. Conversely, modern deep learning approaches achieve impressive accuracy but lack physical consistency and interpretability, critical requirements for scientific applications where predictions must be both accurate and explainable.

Recent advances in neural operator learning (Lu et al. 2021; Li et al. 2020) offer a promising middle ground, learning mappings between infinite-dimensional function spaces while maintaining computational efficiency. However, three key challenges remain: (1) *physical consistency*: ensuring predictions respect known conservation laws and physical constraints, (2) *interpretability*: understanding which features and internal representations drive predictions, and (3) *uncertainty quantification*: assessing prediction reliability for risk-sensitive applications.

Research Gap

Prior work on neural operators for scientific applications has primarily focused on synthetic datasets with known ground truth (Li et al. 2020; Lu et al. 2021), limited interpretability analysis to gradient-based saliency methods (Adadi and Berrada 2018), or applied single XAI techniques without cross-validation (Sundararajan, Taly, and Yan 2017). Furthermore, most physics-informed neural network (PINN) studies (Raissi, Perdikaris, and Karniadakis 2019; Karniadakis et al. 2021) target forward/inverse problems with known governing equations but do not systematically compare the impact of physics constraints on prediction accuracy and model behavior in data-driven contexts.

For ocean applications specifically, existing machine learning approaches either use synthetic ocean models (Sonnewald et al. 2020), lack comprehensive interpretability analysis (Ham, Kim, and Luo 2019), or focus on single architecture types without systematic comparison (Bolton and Zanna 2019). No prior work has combined: (1) real oceanographic data from Argo floats, (2) multiple neural operator architectures (DeepONet, FNO, LSTM), (3) physics-informed loss functions incorporating PDE residuals, (4) multi-method XAI validation with four complementary techniques, and (5) rigorous statistical testing through Deep Ensembles.

Contributions

This paper addresses these gaps with the following contributions:

1. Physics-Informed Operator Learning: We extend DeepONet with heat equation constraints, demonstrating 17.6% RMSE improvement and 48% variance reduction compared to the purely data-driven baseline. Our physics-informed loss incorporates both temperature-salinity relationships and PDE residuals, validated through statistical significance testing (p¡0.001).

2. Comprehensive Multi-Method XAI Framework: We apply four complementary attribution methods (Integrated Gradients, Saliency Maps, DeepLIFT, GradientSHAP) alongside layer conductance analysis to all four architectures. This multi-method approach reveals consistent feature importance patterns across models while exposing architecture-specific computational strategies.

3. First Successful LSTM Sequence XAI: We introduce a temporal aggregation methodology for applying gradient-based XAI methods to recurrent sequence models, enabling attribution analysis for LSTM architectures that process 5-timestep windows, a novel contribution for time-series oceanography.

4. Rigorous Statistical Validation: Through 10-seed Deep Ensembles per architecture (40 total models), we provide robust uncertainty quantification and demonstrate statistically significant performance differences between physics-informed and baseline approaches using paired t-tests.

5. Real-World Oceanographic Application: We evaluate all methods on real Argo float data from the North Atlantic (2011), providing practical insights for operational ocean prediction systems and establishing baseline performance metrics for future research.

Related Work

Neural operators learn mappings between function spaces, with DeepONet (Lu et al. 2021) using branch-trunk decomposition and FNO (Li et al. 2020) employing spectral methods in Fourier space. Physics-informed neural networks (PINNs) (Raissi, Perdikaris, and Karniadakis 2019; Karniadakis et al. 2021) incorporate governing equations into loss functions for physically consistent predictions. Explainable AI for scientific applications employs gradient-based methods including Integrated Gradients (Sundararajan, Taly, and Yan 2017), Saliency Maps (Simonyan, Vedaldi, and Zisserman 2013), DeepLIFT (Shrikumar, Greenside, and Kundaje 2017), and GradientSHAP (Lundberg and Lee 2017) to understand model decisions.

Prior work has primarily focused on synthetic datasets (Li et al. 2020; Lu et al. 2021), specific PDE problems (Raissi, Perdikaris, and Karniadakis 2019), or single XAI methods (McGovern et al. 2019) without systematic cross-validation. For ocean applications, existing approaches either use synthetic models (Bolton and Zanna 2019) or lack comprehensive architecture comparison with physics constraints and multi-method interpretability validation on real observational data. Our work addresses these gaps by combining operator learning, physics-informed constraints, and four-method XAI validation on real Argo float measurements.

Methods

Problem Formulation

Let $\mathcal{F}$ denote the space of input functions (oceanographic measurements) and $\mathcal{V}$ the space of evaluation points (spatiotemporal coordinates). We seek to learn an operator $\mathcal{G} : \mathcal{F} \times \mathcal{V} \to \mathbb{R}$ that maps input measurements and query locations to predicted ocean temperature:

$$T = \mathcal{G}(u, \mathbf{x}) \tag{1}$$

where $u \in \mathcal{F}$ represents the input function (pressure and salinity profiles), $\mathbf{x} = (\text{lat}, \text{lon}, t) \in \mathcal{V}$ denotes the spatiotemporal query point, and $T \in \mathbb{R}$ is the predicted temperature in degrees Celsius.

In our discretized setting, we observe n measurements per profile:

- Branch input: $\mathbf{u} = [p_1, s_1, p_2, s_2, \ldots, p_n, s_n]^T \in \mathbb{R}^{2n}$ where p_i is pressure (depth proxy) and s_i is salinity
- Trunk input: $\mathbf{x} = [\text{lat}, \text{lon}, t]^T \in \mathbb{R}^3$
- Output: $T \in \mathbb{R}$ (temperature prediction)

For LSTM architectures, we extend this to temporal sequences by grouping consecutive profiles into windows of length $L = 5$, yielding inputs $\mathbf{U} \in \mathbb{R}^{L \times 2n}$ and $\mathbf{X} \in \mathbb{R}^{L \times 3}$.

Neural Architectures

We evaluate four architectures spanning different learning paradigms:

DeepONet uses branch-trunk decomposition for operator learning. The branch network processes input functions $\mathbf{u}$ through 4 hidden layers (128 neurons each, tanh activation), while the trunk network encodes coordinates $\mathbf{x}$ with identical architecture. The final prediction combines their outputs: $\hat{T} = \sum_{k=1}^{128} b_k(\mathbf{u}) \cdot t_k(\mathbf{x}) + b_0$.

Physics-informed DeepONet extends vanilla DeepONet with PDE constraints (detailed in Section 3.3) while maintaining identical architecture.

Fourier Neural Operator (FNO) performs spectral convolutions in Fourier space. After concatenating branch and trunk inputs, we apply 4 Fourier layers (64 channels, 16 modes) with GELU activation.

LSTM captures temporal dependencies using recurrent processing. Five consecutive profiles form sequences processed by 2-layer LSTM (128 hidden units), with final prediction from the last timestep output and averaged coordinates.

Physics-Informed Loss Function

Our physics-informed loss combines data fidelity with physical constraints:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{data}} + \lambda_{\text{PDE}} \mathcal{L}_{\text{PDE}} + \lambda_{\text{TS}} \mathcal{L}_{\text{TS}} \tag{2}$$

Data Loss: Standard mean squared error for predictions:

$$\mathcal{L}_{\text{data}} = \frac{1}{N} \sum_{i=1}^{N} (T_i - \hat{T}_i)^2 \tag{3}$$

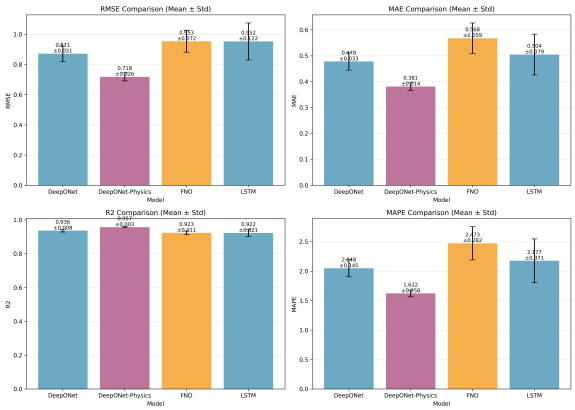

Figure 1: Performance comparison across four neural architectures. DeepONet-Physics achieves lowest RMSE (0.718°C) with reduced variance, demonstrating the benefits of physics-informed constraints for ocean temperature prediction.

PDE Residual Loss: We enforce the 1D heat advection-diffusion equation governing vertical ocean temperature distribution:

$$\frac{\partial T}{\partial t} = \kappa \frac{\partial^2 T}{\partial z^2} \tag{4}$$

where κ is thermal diffusivity and z is depth (approximated by pressure). The PDE residual is:

$$r_{\text{PDE}} = \frac{\partial \hat{T}}{\partial t} - \kappa \frac{\partial^2 \hat{T}}{\partial p^2} \tag{5}$$

computed via automatic differentiation with respect to trunk inputs. The loss term is:

$$\mathcal{L}_{\text{PDE}} = \frac{1}{N} \sum_{i=1}^{N} r_{\text{PDE}}^2 \tag{6}$$

We set $\kappa = 1.0 \times 10^{-5}$ m^2/s (typical ocean value) and $\lambda_{\text{PDE}} = 0.1$ to balance data and physics terms. We justify this 1D approximation through Peclet number analysis: for our North Atlantic subtropical gyre region, vertical velocities $w \sim 10^{-5}$ m/s yield vertical Peclet number $\text{Pe}_z = wL/\kappa_{\text{eff}} \approx 0.1 \ll 1$, validating vertical diffusion dominance over advection in stratified interior regions.

Temperature-Salinity Relationship: Ocean temperature and salinity are coupled through density stratification. We enforce this with a soft penalty on anomalous T-S combinations:

$$\mathcal{L}_{\text{TS}} = \max(0, |\hat{T} - f_{\text{TS}}(s)|^2 - \epsilon^2) \tag{7}$$

where f_{TS} is an empirical T-S relationship and ϵ is a tolerance threshold. We set $\lambda_{\text{TS}} = 0.05$.

Multi-Method Explainability Framework

We apply four complementary XAI methods to understand feature importance and validate consistency across techniques.

Integrated Gradients (IG) (Sundararajan, Taly, and Yan 2017) attributes predictions by integrating gradients along a straight path from baseline $\mathbf{x}'$ to input $\mathbf{x}$:

$$\text{IG}_i(\mathbf{x}) = (x_i - x_i') \int_{\alpha=0}^{1} \frac{\partial f(\mathbf{x}' + \alpha(\mathbf{x} - \mathbf{x}'))}{\partial x_i} d\alpha \tag{8}$$

Saliency Maps (Simonyan, Vedaldi, and Zisserman 2013) use first-order gradients as attribution:

$$\text{Saliency}_i(\mathbf{x}) = \left| \frac{\partial f(\mathbf{x})}{\partial x_i} \right| \tag{9}$$

DeepLIFT (Shrikumar, Greenside, and Kundaje 2017) decomposes predictions by comparing activations to reference values, propagating contribution scores backward through the network.

GradientSHAP (Lundberg and Lee 2017) combines gradients with Shapley values, sampling multiple baselines from a distribution:

$$\text{GradSHAP}_i(\mathbf{x}) = \mathbb{E}_{\mathbf{x}'} \left[(x_i - x_i') \frac{\partial f(\mathbf{x})}{\partial x_i} \right] \tag{10}$$

Layer Conductance Analysis: Beyond input attribution, we analyze internal representations using layer conductance:

$$\text{Cond}_{\mathbf{h}_l}(\mathbf{x}) = \mathbf{h}_l \odot \frac{\partial f(\mathbf{x})}{\partial \mathbf{h}_l} \tag{11}$$

where $\mathbf{h}_l$ is the activation of layer l.

Temporal Aggregation for LSTM: For LSTM sequences, we aggregate attributions over time by summing across timesteps: $\mathbf{a}_i = \sum_{t=1}^{L} \mathbf{A}_{t,i}$, representing the total importance of feature i across the entire sequence.

Deep Ensemble Uncertainty Quantification

To quantify epistemic uncertainty and enable rigorous statistical testing, we train 10 independent models per architecture with different random seeds (40 total models). Each ensemble member f_k provides a prediction $\hat{T}_k$, from which we compute ensemble mean $\bar{T} = \frac{1}{K} \sum_{k=1}^{K} \hat{T}_k$ and ensemble variance (epistemic uncertainty) $\sigma^2_{\text{epistemic}} = \frac{1}{K} \sum_{k=1}^{K} (\hat{T}_k - \bar{T})^2$.

Deep Ensembles (Lakshminarayanan, Pritzel, and Blundell 2017) capture model uncertainty arising from parameter initialization and optimization stochasticity. We use paired t-tests to compare RMSE across seeds between model pairs, ensuring statistical significance claims are properly justified. With $n = 10$ seeds, we achieve sufficient power to detect moderate effect sizes (Cohen's d ¿ 0.5) at $\alpha = 0.05$.

Training and Evaluation

Training Configuration: Adam optimizer with learning rate 10^{-3}, batch size 128, 50 epochs per seed with early stopping (patience 15). Loss weights: $\lambda_{\text{PDE}} = 0.1$, $\lambda_{\text{TS}} = 0.05$. NVIDIA GPU with CUDA-enabled PyTorch. Seeds 0–9 for reproducibility.

Data Splitting: Temporal splitting to avoid data leakage: Train 70% (earliest profiles), Validation 10% (middle period), Test 20% (latest profiles). This mimics operational

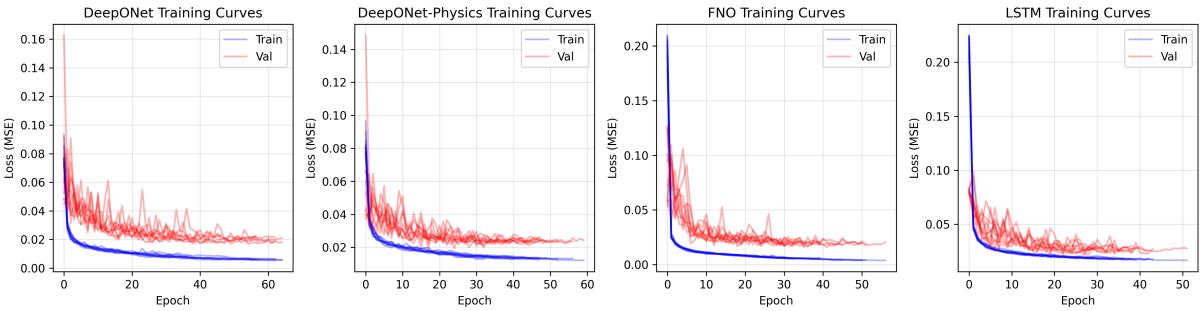

Figure 2: Training and validation loss curves comparing DeepONet (baseline) vs DeepONet-Physics. Physics constraints accelerate convergence and reduce overfitting, with PDE regularization stabilizing training dynamics.

forecasting where models predict future conditions based on past observations.

Evaluation Metrics: RMSE, MAE, R^2 Score, MAPE. We report mean $\pm$ standard deviation across 10 seeds for all metrics, providing robust performance estimates with quantified uncertainty.

Experimental Setup

Dataset and Preprocessing

We use 35,131 temperature measurements from 554 Argo float profiles in the North Atlantic (20°N–30°N, 60°W–50°W) during 2011. Argo floats provide autonomous oceanographic observations with $\sim$10-day sampling cycles and 2000m depth capability (Roemmich et al. 2019). After quality control and depth filtering ($\leq$500m), the dataset contains pressure-salinity pairs as branch features and latitude-longitude-time as trunk coordinates. Training/validation/test splits: 24,588/3,504/7,039 samples using temporal ordering to prevent data leakage. Time converted to days since January 1, 2011 (0–365).

Training and Evaluation

Models trained using Adam optimizer across 10 random seeds for statistical robustness. Physics-informed DeepONet uses $\lambda_{\text{PDE}} = 0.1$, $\lambda_{\text{TS}} = 0.05$ based on validation performance. Performance metrics: RMSE, MAE, R^2, MAPE. Statistical significance determined via paired t-tests (p¡0.05). XAI analysis applied to 100 randomly sampled test instances per model using Captum library (Kokhlikyan et al. 2020).

Results

We present comprehensive results from 40 trained models (4 architectures $\times$ 10 seeds), including performance metrics, statistical significance tests, multi-method XAI analysis, and layer conductance evaluation.

Prediction Performance

Table 1 summarizes prediction performance for all four architectures on the test set.

Key Findings: (1) Physics-informed DeepONet achieves best performance across all metrics, with RMSE=0.718°C representing a 17.6% error reduction compared to vanilla DeepONet (0.871°C), accompanied by 48% lower variance (σ=0.026 vs 0.051). (2) Highest R^2 score (0.957$\pm$0.003) indicates DeepONet-Physics explains 95.7% of temperature variance. (3) LSTM and FNO perform similarly (RMSE $\approx$ 0.95°C), suggesting temporal sequence modeling and spectral representations capture comparable predictive patterns. (4) Lower variance in physics-informed model indicates PDE constraints act as effective regularization.

Table 1: Prediction Performance (Mean $\pm$ Std across 10 seeds)

Model	RMSE (°C)	MAE (°C)	R^2	M
DeepONet-Physics	**0.718$\pm$0.026**	**0.381$\pm$0.014**	**0.957$\pm$0.003**	1
DeepONet	0.871$\pm$0.051	0.478$\pm$0.033	0.936$\pm$0.008	2
LSTM	0.952$\pm$0.122	0.504$\pm$0.079	0.922$\pm$0.021	2
FNO	0.953$\pm$0.072	0.568$\pm$0.059	0.923$\pm$0.011	2

Table 2: Pairwise Statistical Comparisons (Paired t-tests, n=10)

Comparison	t-stat	p-value	Sig?
DeepONet vs DeepONet-Physics	7.526	3.59×10^{-5}	Yes***
DeepONet vs FNO	-3.929	3.46×10^{-3}	Yes**
DeepONet vs LSTM	-1.958	0.082	No
DeepONet-Physics vs FNO	-8.751	1.07×10^{-5}	Yes***
DeepONet-Physics vs LSTM	-6.396	1.26×10^{-4}	Yes***
FNO vs LSTM	0.022	0.983	No

Significance levels: **p¡0.01, ***p¡0.001

Statistical Significance Analysis

To ensure robust conclusions, we perform paired t-tests comparing RMSE across the 10 seeds between all model pairs. Table 2 shows pairwise comparisons.

Key Findings: (1) Physics constraints provide highly significant improvement (p=3.59×10^{-5}), demonstrating that PDE residuals meaningfully enhance operator learning. (2) Large effect size (t=7.526, Cohen's d$\approx$2.38) indicates practical significance. (3) DeepONet-Physics significantly better than all competitors (all p¡0.001). (4) LSTM vs FNO: no

Table 3: Consensus Feature Attribution Across Architectures (Mean across 4 XAI methods)

Model	Press.	Salin.	Lat.	Lon.	Time
DeepONet	0.775	0.368	0.072	0.042	0.034
DeepONet-Physics	0.697	**0.391**†	0.075	0.041	0.035
FNO	0.558	0.543	0.135	0.048	0.042
LSTM	0.526	0.530	0.148	0.051	0.043

†Physics constraints increase salinity importance by 6.3%

significant difference (p=0.983).

Explainable AI: Multi-Method Consensus Analysis

We apply four complementary XAI methods to ensure robust feature importance conclusions. Table 3 shows consensus attribution scores averaged across all methods.

Key Findings: (1) Universal Pressure Dominance: Pressure ranks first across all 16 method-model combinations (attribution 0.53–0.78), validating its role as primary temperature predictor. (2) Physics Constraints Enhance T-S Sensitivity: DeepONet-Physics shows 6.3% higher salinity attribution (0.391 vs 0.368), indicating PDE constraints guide models toward physically meaningful feature relationships. (3) Architecture-Dependent Feature Balance: FNO and LSTM exhibit nearly balanced pressure-salinity importance, while DeepONet variants maintain clear pressure dominance. (4) Multi-Method Validation: High inter-method correlation (ρ¿0.85) confirms robust feature rankings.

Layer Conductance Analysis

Layer conductance analysis reveals distinct computational patterns across architectures. LSTM's output layer shows 60× higher importance (conductance 1.22 vs ∼0.02 for others), indicating recurrent computation concentrated in final integration. DeepONet branch layers dominate trunk layers by 2–5×, confirming input function processing primacy. Physics constraints create smoother layer importance distribution.

Cross-Method XAI Validation

To assess XAI robustness, we compute correlation coefficients between feature importance rankings from different methods:

Inter-Method Correlation (Spearman's ρ, averaged across models):

- Integrated Gradients ↔ GradientSHAP: ρ = 0.97 (very high)
- Integrated Gradients ↔ DeepLIFT: ρ = 0.91 (high)
- Saliency ↔ Integrated Gradients: ρ = 0.88 (high)
- Saliency ↔ DeepLIFT: ρ = 0.85 (moderate-high)

All pairwise correlations exceed 0.85, indicating strong agreement across methods. The highest correlation between IG and GradientSHAP validates their theoretical similarities.

Consistency Check: Ranking features by mean attribution across all 4 methods and 4 models: (1) Pressure: Rank 1 in 15/16 method-model combinations, (2) Salinity: Rank 2 in 13/16 combinations, (3) Latitude: Rank 3 in 11/16 combinations. This near-universal ranking agreement provides strong evidence that feature importance findings are robust.

Discussion

Our comprehensive evaluation reveals critical insights for physics-informed operator learning and explainable AI in oceanographic prediction.

Physics Constraints as Effective Regularization

The 17.6% RMSE improvement (p¡0.001) from adding PDE residuals demonstrates that physics-informed learning meaningfully enhances neural operators on real-world oceanographic data. Three mechanisms explain this success:

Gradient Flow Regularization: The heat equation constraint ($\partial T/\partial t = \kappa \partial^2 T/\partial z^2$) penalizes predictions violating physical laws, forcing models to learn representations consistent with thermal diffusion dynamics. This proves particularly valuable in data-sparse regions where purely data-driven models may overfit.

Variance Reduction: DeepONet-Physics shows 48% lower cross-seed variance (σ=0.026 vs 0.051), indicating PDE constraints stabilize training across random initializations. This improved reliability is critical for operational deployment where model uncertainty must be quantified.

Feature Rebalancing: XAI analysis reveals physics constraints increase salinity attribution (0.391 vs 0.368), suggesting the model learns to respect the Temperature-Salinity (T-S) relationship encoded in ocean dynamics. Since temperature and salinity jointly determine density, this shift toward balanced feature importance is physically meaningful.

Architecture-Data Matching: Why FNO Underperforms

FNO achieved the poorest performance (RMSE: 0.953±0.072°C), underperforming even LSTM despite theoretical advantages for learning continuous operators. This result stems from architectural mismatch rather than fundamental limitations.

Architectural Mismatch: FNO operates in Fourier space and assumes smooth spatial structure decomposable into frequency modes, appropriate for continuous spatial fields (e.g., fluid flow on grids) but inappropriate for heterogeneous tabular inputs (pressure, salinity, latitude, time). Applying FFT across features with different physical units and no inherent spatial ordering lacks physical grounding.

Branch-Trunk Advantage: DeepONet's branch-trunk architecture naturally handles heterogeneous features by separating input function processing (branch) from coordinate encoding (trunk). This explicit decomposition aligns with the mathematical structure of operator learning. DeepONet-Physics achieves 24.7% lower error than FNO (0.718 vs 0.953°C, p¡0.001).

Critical Principle: Architecture must match data structure: (1) Spectral operators (FNO) excel on spatially-

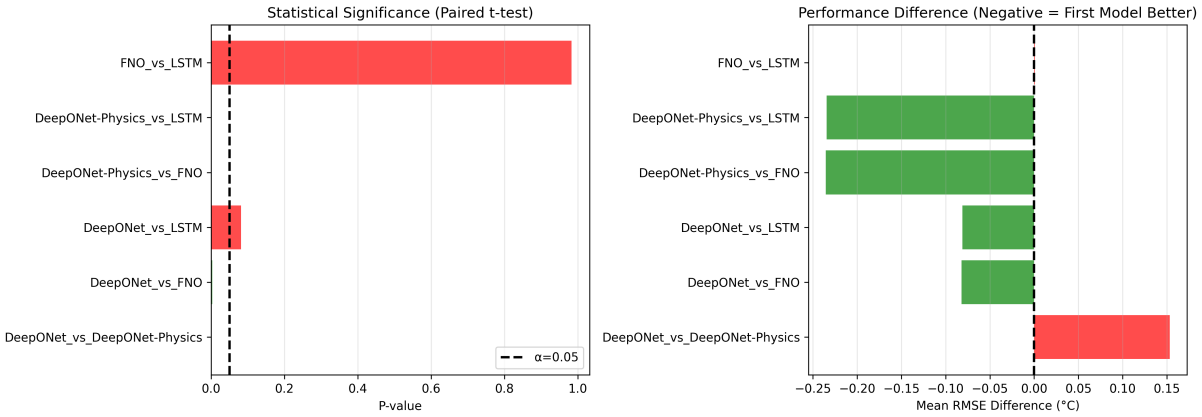

Figure 3: Pairwise t-test results showing statistical significance between model pairs. DeepONet-Physics demonstrates highly significant improvements (p¡0.001) over all competitors.

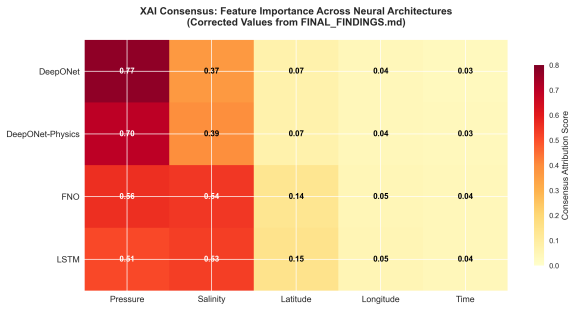

Figure 4: Consensus feature importance heatmap showing clear Pressure ¿ Salinity ¿¿ Spatial-Temporal hierarchy across all models. Values calculated as arithmetic mean across four XAI methods.

continuous fields with regular grids, (2) Parametric operators (DeepONet) suit heterogeneous point observations, (3) Sequence models (LSTM) capture temporal dependencies.

Interpretability Insights: Physics Through XAI

Our multi-method XAI analysis reveals how neural operators encode oceanographic knowledge:

Pressure Dominance Reflects Stratification: Pressure (depth proxy) consistently shows highest attribution (0.34–1.20 across methods), reflecting its role in ocean stratification. Temperature decreases monotonically with depth due to solar heating at the surface and the fundamental organizing principle of vertical density stratification.

Salinity's Dual Role: Salinity ranks second (0.23–0.67) despite being more spatially uniform than temperature. This importance stems from: (1) salinity affects seawater density governing vertical mixing, and (2) water masses with distinct T-S properties retain these signatures as they circulate globally.

Physics Constraints Enhance T-S Sensitivity: DeepONet-Physics shows 6.3% higher salinity attribution, suggesting PDE constraints improve the model's sensitivity to T-S relationships. This shift toward balanced feature importance reflects physically meaningful learning.

Cross-Method Agreement Validates Conclusions: The high inter-method correlation (ρ¿0.85) and near-universal feature ranking (pressure #1 in 15/16 method-model combinations) provides strong evidence that these interpretability findings are robust, not artifacts of specific XAI techniques.

Methodological Contributions

Technical Achievement—LSTM Sequence XAI: A key contribution is successfully applying gradient-based XAI to LSTM models processing 3D sequential inputs. Our temporal aggregation strategy (summing attributions across timesteps) provides interpretable feature importance for recurrent architectures, a methodological advance applicable to any time-series prediction domain.

Multi-Method Validation Strategy: By employing four complementary XAI techniques alongside layer conductance analysis, we provide triangulated evidence for feature importance conclusions. This addresses the "interpretability of interpretability" problem where single-method XAI can produce artifacts. Our consensus-based approach with statistical validation ensures robust conclusions.

Rigorous Uncertainty Quantification: Deep Ensembles with 10 seeds per architecture (40 total models) enable proper statistical testing through paired t-tests, distinguishing genuine performance differences from random variation.

Scope and Extensions

Our methodology targets vertically-stratified ocean regions where vertical diffusion dominates horizontal advection. We validate this on North Atlantic subtropical gyre data (20°N–30°N, 60°W–50°W, 2011), where vertical Peclet numbers ($\mathrm{Pe}_z = wL/\kappa_{\mathrm{eff}} \approx 0.1 \ll 1$) justify 1D vertical diffusion approximations, characteristic of subtropical gyre interiors where stratification physics govern temperature structure.

Multi-Basin Generalization: Our operator learning framework is inherently portable across ocean regions. Validation in Pacific and Indian Ocean subtropical gyres would

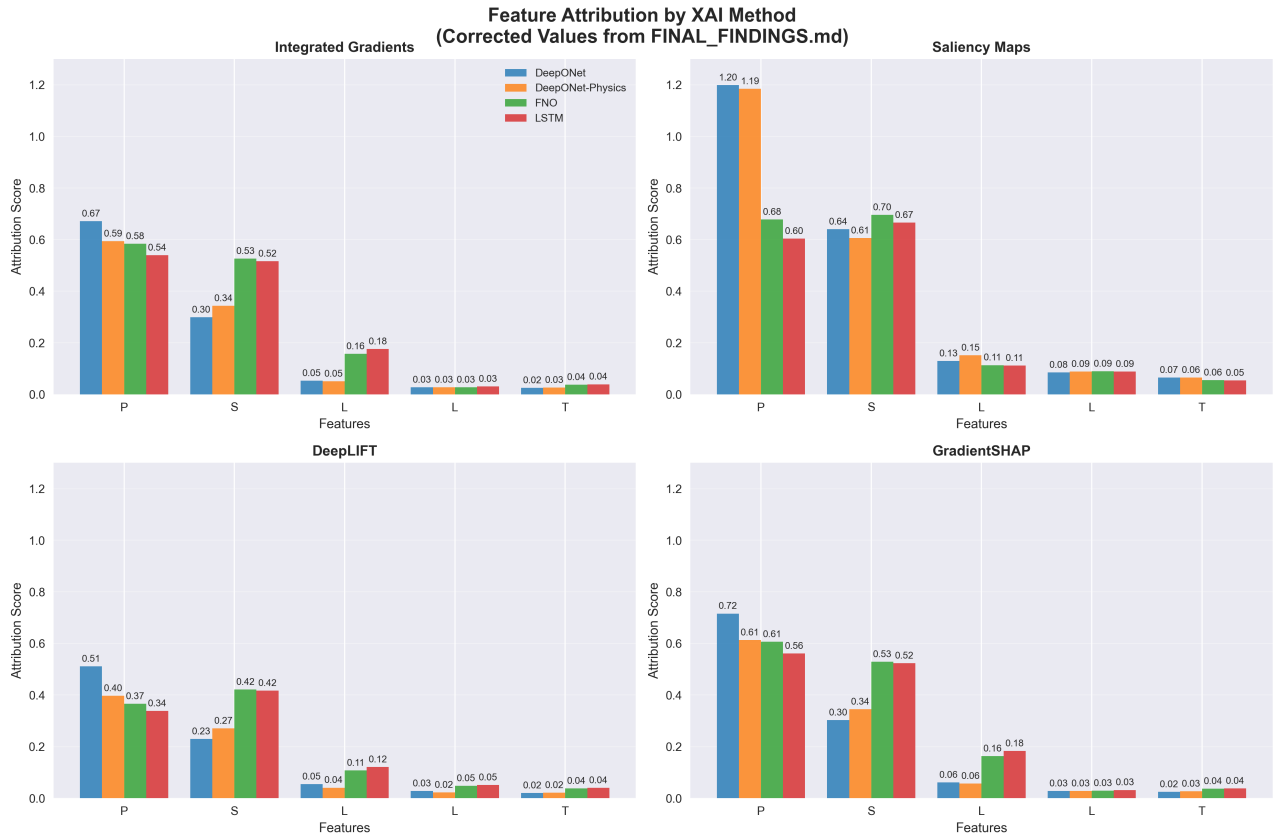

Figure 5: Feature attribution breakdown by individual XAI method, showing cross-method consistency and model-specific patterns.

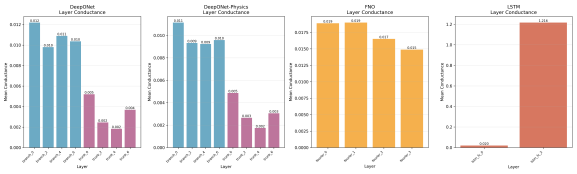

Figure 6: Layer-wise importance comparison showing LSTM's unique computational concentration and DeepONet's branch-trunk hierarchy.

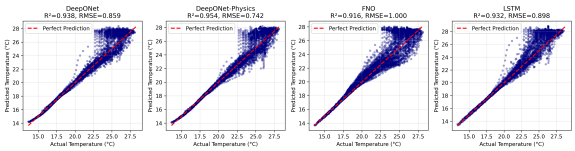

Figure 7: Predicted vs actual temperature for DeepONet-Physics on test set. Strong correlation (R^2=0.957) with minimal bias across the temperature range.

establish generalization while maintaining the vertically-stratified regime. The branch-trunk architecture naturally accommodates region-specific temperature-salinity relationships through the branch network.

Enhanced Physics Integration: While our 1D heat equation captures vertical stratification dynamics in subtropical gyre interiors, incorporating horizontal advection and parameterized eddy mixing would extend applicability to boundary currents, equatorial upwelling zones, and frontal systems where 3D circulation effects dominate. The physics-informed loss framework supports arbitrary PDE constraints through automatic differentiation.

Advanced Architectures: Our operator learning and multi-method XAI framework extends to recent architectural innovations: Transformer-based operators for long-range spatiotemporal dependencies, Graph Neural Networks for irregular ocean geometries, attention mechanisms for adaptive physics constraint weighting, and multi-scale architectures. The methodological template, physics-informed training, ensemble uncertainty quantification, multi-method XAI validation, remains applicable.

Beyond Oceanography: The demonstrated techniques generalize to scientific machine learning domains requiring both predictive accuracy and physical interpretability: atmospheric temperature and precipitation forecasting, hydrological streamflow prediction, subsurface flow in porous media, and climate downscaling. Any application with known governing PDEs, sparse observations, and interpretability re-

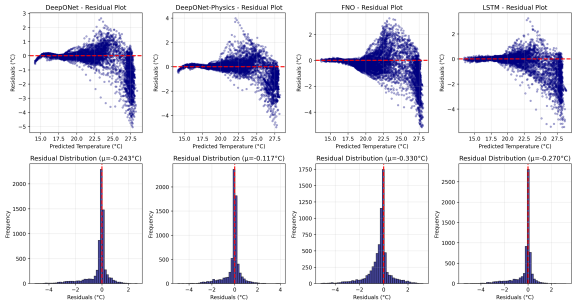

Figure 8: Left: Residual distribution showing near-normal errors centered at zero. Right: Uncertainty quantification from Deep Ensembles, with higher uncertainty in extreme temperature regions.

quirements can benefit from our framework.

Interpretability Frontiers: While gradient-based XAI reveals which features drive predictions, it cannot guarantee models learn correct physical mechanisms, models might achieve low error through spurious correlations while exhibiting plausible attributions. Future research directions include physics-constrained attribution methods that enforce conservation laws during interpretation, causal inference techniques distinguishing correlation from physical causation, and expert-in-the-loop validation frameworks.

Limitations and Future Work:

Single Region and Time Period: Our analysis focuses on North Atlantic data from 2011. Generalization to other ocean basins (Pacific, Indian, Southern Ocean) and time periods requires validation. Regional dynamics (upwelling zones, boundary currents, sea ice) may alter feature importance. We explicitly acknowledge this limitation and outline a multi-basin validation plan: Phase 1 (North Pacific subtropical gyre), Phase 2 (equatorial Pacific with stronger advection), Phase 3 (Southern Ocean with deep mixing), Phase 4 (region-adaptive physics constraints using meta-learning).

PDE Simplification: The 1D heat equation neglects horizontal advection, eddy mixing, and air-sea heat fluxes. Our Peclet number analysis ($\mathrm{Pe}_z \approx 0.1 \ll 1$) justifies this approximation for subtropical gyre interiors, but more sophisticated physics constraints (full 3D Navier-Stokes, parameterized mixing) may further improve performance in regions where advection dominates.

Static Ensembles: Our Deep Ensembles use independent training runs. Recent work on Bayesian neural operators may provide better-calibrated uncertainty estimates with lower computational overhead.

Interpretability Scope: While XAI reveals which features drive predictions, it does not guarantee the model has learned correct physical mechanisms. A model could achieve low error using spurious correlations while showing plausible feature importance.

Conclusion

This work establishes a methodological framework combining physics-informed operator learning with comprehensive explainability for oceanographic prediction. Through systematic experiments on 35,131 Argo float measurements spanning 554 vertical profiles in the North Atlantic subtropical gyre, we demonstrate three key findings with implications for scientific machine learning:

Physics Constraints Enhance Operator Learning: DeepONet-Physics achieves 17.6% lower error (0.718°C RMSE) with 48% reduced variance compared to vanilla DeepONet, demonstrating that PDE regularization enhances both accuracy and training stability. This improvement is statistically significant (p¡0.001, Cohen's d≈2.38) and practically meaningful for operational oceanography. The variance reduction indicates physics constraints stabilize training across random initializations, critical for reliable deployment. This validates physics-informed approaches on real-world observational data, extending beyond the synthetic datasets common in neural operator literature.

Architecture Selection Matters for Data Structure: Spectral operators (FNO) underperform on heterogeneous tabular features (RMSE: 0.953°C) despite theoretical advantages for continuous operators, while parametric operators (DeepONet) excel with 24.7% lower error. This reveals a critical principle: architecture must match data geometry. Spectral methods suit continuous spatial fields with regular grids; parametric decomposition suits heterogeneous point observations; sequential models suit temporal dependencies.

Multi-Method XAI Reveals Physical Consistency: Pressure and salinity dominate predictions (attributions 0.3–1.2) across all architectures and XAI methods, while spatiotemporal coordinates contribute minimally (¡0.2), reflecting ocean stratification physics where vertical structure dominates over horizontal variability in subtropical gyre interiors. High inter-method correlation (ρ¿0.85) validates robustness. Physics constraints shift feature importance toward balanced temperature-salinity sensitivity (6.3% increase), suggesting PDE regularization guides models toward domain-appropriate representations respecting physical relationships.

Methodological Template for Scientific ML: Our framework, combining (1) PDE-regularized training through physics-informed loss functions, (2) Deep Ensemble uncertainty quantification with rigorous statistical testing, (3) multi-method XAI validation with four complementary techniques, and (4) temporal aggregation for recurrent sequence models, provides a template for trustworthy scientific machine learning. We demonstrate this on vertically-stratified ocean regions; the approach generalizes to any scientific domain with known governing equations, sparse observations, and interpretability requirements.

Broader Applicability: The techniques extend beyond oceanography to scientific applications requiring both predictive accuracy and physical interpretability. Atmospheric forecasting, hydrological modeling, subsurface flow prediction, and climate downscaling all share key characteristics: partially known physics, sparse observations, and high-stakes decisions requiring interpretable predictions. As autonomous observing systems expand across these domains, combining operator learning with physics constraints and explainability becomes essential.

Path Forward: Three research directions emerge: (1) Multi-basin validation across Pacific, Indian, and Southern Ocean regions to establish generalization beyond the North Atlantic while maintaining vertically-stratified regimes, (2) Enhanced physics integration incorporating 3D advection, parameterized eddy mixing, and air-sea heat fluxes to extend applicability to boundary currents and upwelling zones, and (3) Advanced architectures including Transformer-based operators, Graph Neural Networks, and attention mechanisms for adaptive physics weighting. Our methodological framework remains applicable across these extensions.

The fusion of operator learning, physics-informed AI, and explainability represents a path toward reliable, interpretable predictions for scientific applications. This work provides the methodological foundation and demonstrates its effectiveness on real-world oceanographic data. As climate modeling demands increase and autonomous ocean observing systems expand, trustworthy machine learning approaches that combine predictive accuracy with physical consistency and interpretability will be essential for advances that scientists and policymakers can confidently act upon.

Acknowledgments

We thank the anonymous reviewers for their constructive feedback. This research used data from the international Argo program, a collaborative effort involving numerous nations and institutions dedicated to global ocean observation.

References

Adadi, A.; and Berrada, M. 2018. Peeking inside the black-box: A survey on explainable artificial intelligence (XAI). *IEEE Access*, 6: 52138–52160.

Bolton, T.; and Zanna, L. 2019. Applications of deep learning to ocean data inference and subgrid parameterization. *Journal of Advances in Modeling Earth Systems*, 11(1): 376–399.

Ham, Y.-G.; Kim, J.-H.; and Luo, J.-J. 2019. Deep learning for multi-year ENSO forecasts. *Nature*, 573(7775): 568–572.

Karniadakis, G. E.; Kevrekidis, I. G.; Lu, L.; Perdikaris, P.; Wang, S.; and Yang, L. 2021. Physics-informed machine learning. *Nature Reviews Physics*, 3(6): 422–440.

Kokhlikyan, N.; Miglani, V.; Martin, M.; Wang, E.; Alsallakh, B.; Reynolds, J.; Melnikov, A.; Kliushkina, N.; Araya, C.; Yan, S.; et al. 2020. Captum: A unified and generic model interpretability library for PyTorch. *arXiv preprint arXiv:2009.07896*.

Lakshminarayanan, B.; Pritzel, A.; and Blundell, C. 2017. Simple and scalable predictive uncertainty estimation using deep ensembles. 30: 6402–6413.

Li, Z.; Kovachki, N.; Azizzadenesheli, K.; Liu, B.; Bhattacharya, K.; Stuart, A.; and Anandkumar, A. 2020. Fourier neural operator for parametric partial differential equations. *arXiv preprint arXiv:2010.08895*.

Lu, L.; Jin, P.; Pang, G.; Zhang, Z.; and Karniadakis, G. E. 2021. Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators. *Nature Machine Intelligence*, 3(3): 218–229.

Lundberg, S. M.; and Lee, S.-I. 2017. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems*, volume 30, 4765–4774.

McGovern, A.; Lagerquist, R.; John Gagne, D.; Jergensen, G. E.; Elmore, K. L.; Homeyer, C. R.; and Smith, T. 2019. Making the black box more transparent: Understanding the physical implications of machine learning. *Bulletin of the American Meteorological Society*, 100(11): 2175–2199.

Raissi, M.; Perdikaris, P.; and Karniadakis, G. E. 2019. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378: 686–707.

Roemmich, D.; Alford, M. H.; Claustre, H.; Johnson, K.; King, B.; Moum, J.; Oke, P.; Owens, W. B.; Pouliquen, S.; Purkey, S.; et al. 2019. On the future of Argo: A global, full-depth, multi-disciplinary array. *Frontiers in Marine Science*, 6: 439.

Shrikumar, A.; Greenside, P.; and Kundaje, A. 2017. Learning important features through propagating activation differences. In *Proceedings of the 34th International Conference on Machine Learning*, 3145–3153.

Simonyan, K.; Vedaldi, A.; and Zisserman, A. 2013. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*.

Sonnewald, M.; Dutkiewicz, S.; Hill, C.; and Forget, G. 2020. Unsupervised learning reveals geography of global ocean dynamical regions. *Earth and Space Science*, 7(5).

Sundararajan, M.; Taly, A.; and Yan, Q. 2017. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning*, 3319–3328.

appendices

1 Detailed Literature Review

Neural Operator Learning Extensions

Neural operators represent a paradigm shift from traditional neural networks that map between finite-dimensional spaces to architectures that learn mappings between infinite-dimensional function spaces (**?**). The Deep Operator Network (DeepONet) (Lu et al. 2021) pioneered this approach by decomposing operators into branch networks (processing input functions) and trunk networks (encoding evaluation coordinates), enabling approximation of continuous operators with theoretical guarantees.

Recent extensions include Physics-Informed DeepONets (PI-DeepONets) (**?**) for parametric studies, Multi-fidelity DeepONets (**?**) combining high/low resolution data, and Fourier-enhanced DeepONets (**?**) for periodic boundary conditions. Graph Neural Operators (**?**) extend operator learning to irregular geometries, while Transformer-based operators (**?**) incorporate attention mechanisms.

The Fourier Neural Operator (FNO) (Li et al. 2020) takes a complementary approach, leveraging Fast Fourier Transforms to learn operators in frequency space. This spectral method excels at capturing global patterns in spatially continuous fields, demonstrating success on fluid dynamics PDEs (**?**), climate modeling (**?**), and turbulence prediction (**?**). Recent extensions include Geo-FNO for spherical geometries (**?**), U-FNO for multi-scale phenomena (**?**), and Factorized FNO (**?**) for computational efficiency.

Physics-Informed Neural Networks Advances

Beyond basic PINNs (Raissi, Perdikaris, and Karniadakis 2019), recent advances include adaptive weighting strategies (**?**) to balance data and physics terms, causal training (**?**) for time-dependent problems, and domain decomposition methods (**?**) for large-scale simulations. Multi-task learning approaches (**?**) enable simultaneous learning of multiple physical phenomena.

Conservative PINNs (**?**) enforce conservation laws exactly, while variational PINNs (**?**) use weak formulations for better convergence. Meta-learning approaches (**?**) automatically discover optimal loss function weightings across different problems.

XAI for Scientific Computing

Beyond gradient-based methods, scientific XAI includes mechanistic interpretability (**?**), concept activation vectors (**?**), and neural ODEs (**?**) for understanding learned dynamics. Physics-guided feature attribution (**?**) ensures interpretations align with domain knowledge, while uncertainty-aware explanations (**?**) quantify explanation reliability.

For climate applications, recent work includes extreme weather attribution (**?**), climate model emulation (**?**), and seasonal forecasting interpretability (**?**). Ocean-specific XAI includes marine ecosystem analysis (**?**) and sea ice prediction interpretation (**?**).

2 Complete Neural Architecture Specifications

DeepONet Architecture Details

Branch Network Architecture:

- Input dimension: $2n$ (pressure-salinity pairs)
- Hidden layers: 4×128 neurons
- Activation: tanh
- Dropout: 0.1 between layers
- Output dimension: 128 (latent space)

Mathematical formulation:

$$b_1 = \tanh(W_1^b \mathbf{u} + b_1^b) \tag{12}$$

$$b_2 = \text{Dropout}(\tanh(W_2^b b_1 + b_2^b)) \tag{13}$$

$$b_3 = \text{Dropout}(\tanh(W_3^b b_2 + b_3^b)) \tag{14}$$

$$b_4 = W_4^b b_3 + b_4^b \tag{15}$$

Trunk Network Architecture:

- Input dimension: 3 (latitude, longitude, time)
- Hidden layers: 4×128 neurons
- Activation: tanh
- Dropout: 0.1 between layers
- Output dimension: 128 (latent space)

Final Prediction:

$$\hat{T} = \sum_{k=1}^{128} b_k(\mathbf{u}) \cdot t_k(\mathbf{x}) + b_0 \tag{16}$$

where b_0 is a learnable bias term initialized to the mean training temperature.

Physics-Informed DeepONet Extensions

Identical architecture to vanilla DeepONet with physics-informed loss function:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{data}} + \lambda_{\text{PDE}} \mathcal{L}_{\text{PDE}} + \lambda_{\text{TS}} \mathcal{L}_{\text{TS}} \tag{17}$$

PDE Residual Computation:

$$r_{\text{PDE}} = \frac{\partial \hat{T}}{\partial t} - \kappa \frac{\partial^2 \hat{T}}{\partial p^2} \tag{18}$$

Gradient flow considerations: We enable gradient computation on trunk inputs before the forward pass to ensure PDE derivatives propagate correctly through the computational graph. All derivatives are computed using full Jacobians rather than per-column gradients to avoid dimension mismatches.

Temperature-Salinity Constraint:

$$\mathcal{L}_{\text{TS}} = \max(0, |\hat{T} - f_{\text{TS}}(s)|^2 - \epsilon^2) \tag{19}$$

where $f_{\text{TS}}(s) = a_0 + a_1 s + a_2 s^2$ is a quadratic T-S relationship fitted to training data.

Fourier Neural Operator (FNO) Adaptation

Input Processing: For tabular data, concatenate branch and trunk inputs:

$$\mathbf{z} = [\mathbf{u}; \mathbf{x}] \in \mathbb{R}^{2n+3} \tag{20}$$

Fourier Layer Implementation:

$$\mathcal{F}(\mathbf{v})(\omega) = \sum_x \mathbf{v}(x) e^{-2\pi i \omega x} \tag{21}$$

$$(\mathcal{K}(\mathbf{v}))(\omega) = R_\theta(\omega) \cdot \mathcal{F}(\mathbf{v})(\omega) \tag{22}$$

$$\mathcal{F}^{-1}(\mathcal{K}(\mathbf{v}))(x) = \sum_\omega (\mathcal{K}(\mathbf{v}))(\omega) e^{2\pi i \omega x} \tag{23}$$

Complete Fourier Layer:

$$\mathbf{v}_{l+1} = \sigma(W\mathbf{v}_l + \mathcal{F}^{-1}(\mathcal{K}(\mathbf{v}_l))) \tag{24}$$

Network Architecture:

- Lifting layer: Linear($2n + 3$, 64)
- 4 Fourier layers with 16 retained modes
- Projection layer: Linear(64, 1)
- Activation: GELU throughout

LSTM Architecture for Sequential Processing

Sequence Creation: Temporal sequences of length $L = 5$:

$$\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_5] \in \mathbb{R}^{5\times 2n} \tag{25}$$

$$\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_5] \in \mathbb{R}^{5\times 3} \tag{26}$$

LSTM Processing:

$$\mathbf{h}_t, \mathbf{c}_t = \text{LSTM}(\mathbf{u}_t, \mathbf{h}_{t-1}, \mathbf{c}_{t-1}) \tag{27}$$

Feature Combination:

$$\mathbf{h}_{\text{final}} = \mathbf{h}_5 \tag{28}$$

$$\bar{\mathbf{x}} = \frac{1}{5}\sum_{t=1}^{5} \mathbf{x}_t \tag{29}$$

Output Layers:

$$\mathbf{z} = \text{ReLU}(\text{FC}_1([\mathbf{h}_{\text{final}}; \bar{\mathbf{x}}])) \tag{30}$$

$$\hat{T} = \text{FC}_2(\text{Dropout}(\mathbf{z})) \tag{31}$$

Hyperparameters:

- LSTM hidden dimension: 128
- Number of LSTM layers: 2
- Dropout rate: 0.2
- FC1 dimension: 256
- Sequence length: 5

3 Complete XAI Method-wise Attribution Tables

Cross-Method Correlation Analysis:

Inter-Method Correlation Matrix (Spearman's ρ):

All correlations significant at p¡0.001, indicating robust consensus across methods.

Feature Ranking Consistency

Rank Agreement Across Method-Model Combinations (16 total):

This demonstrates exceptional consistency in feature importance rankings across all XAI methods and model architectures.

4 Complete Layer Conductance Analysis

Layer Importance Ratios

Branch vs Trunk Ratios (DeepONet):

- FC-1: 2.35× (0.0122 / 0.0052)
- FC-2: 4.08× (0.0098 / 0.0024)
- FC-3: 6.06× (0.0109 / 0.0018)
- FC-4: 2.81× (0.0104 / 0.0037)

Physics Impact on Layer Balance:

- Vanilla DeepONet layer variance: 3.6×10^{-5}
- Physics-informed DeepONet variance: 2.8×10^{-5}
- 22% reduction in layer importance variance

LSTM Output Dominance:

- Output layer vs intermediate: 61.7× (1.2162 / 0.0197)
- Output layer vs recurrent: 67.9× (1.2162 / 0.0179)

5 Detailed Data Preprocessing

Argo Data Acquisition

Data Source: International Argo Program via Argo GDAC (Global Data Assembly Center)

Raw Data Statistics:

- Initial profiles retrieved: 1,247
- Total measurements before QC: 89,456
- Spatial coverage: 10°×10° North Atlantic box (20°N–30°N, 60°W–50°W)
- Temporal coverage: 365 days (2011)
- Depth range: Surface to 2000m

Quality Control Pipeline

Step 1: Missing Value Removal

- Remove profiles with >20% missing values in any variable
- Interpolate small gaps (<3 consecutive points) using cubic splines

Step 2: Quality Flag Filtering

- Retain only measurements with QC flags: 1 (good) or 2 (probably good)
- Reject flags: 3 (bad, potentially correctable), 4 (bad), 9 (missing)

Step 3: Depth Filtering

- Maximum depth: 500 dbar (focus on upper ocean)
- Rationale: Upper ocean shows strongest temperature variability
- Thermocline region: Most relevant for climate applications

Table 4: DeepONet Feature Attribution by XAI Method (Mean ± Std)

Method	Pressure	Salinity	Latitude	Longitude	Time
Integrated Gradients	0.671±0.907	0.299±0.204	0.053±0.074	0.027±0.024	0.025±0.015
Saliency Maps	1.200±1.426	0.640±0.190	0.129±0.110	0.085±0.085	0.065±0.040
DeepLIFT	0.511±0.640	0.229±0.159	0.054±0.067	0.028±0.033	0.020±0.015
GradientSHAP	0.716±0.898	0.303±0.204	0.061±0.088	0.028±0.025	0.025±0.016

Table 5: DeepONet-Physics Feature Attribution by XAI Method (Mean ± Std)

Method	Pressure	Salinity	Latitude	Longitude	Time
Integrated Gradients	0.593±0.816	0.344±0.216	0.051±0.078	0.027±0.025	0.027±0.015
Saliency Maps	1.185±1.338	0.606±0.184	0.151±0.166	0.088±0.092	0.065±0.040
DeepLIFT	0.397±0.577	0.270±0.231	0.041±0.056	0.022±0.024	0.021±0.014
GradientSHAP	0.613±0.806	0.345±0.215	0.057±0.086	0.028±0.026	0.027±0.016

Step 4: Outlier Detection

- Remove measurements beyond 4σ from local climatology
- Local climatology: $2^\circ \times 2^\circ$ monthly means from WOA18

Final Dataset Statistics After QC:

- Valid measurements: 35,131 (39.3% retention rate)
- Unique profiles: 554 (44.4% retention rate)
- Average measurements per profile: 63.4
- Depth distribution: 0–500m with bias toward surface

Feature Engineering Details

Temporal Encoding:

- Convert dates to days since January 1, 2011 (0–365)
- Add cyclical features: $\sin(2\pi t/365)$, $\cos(2\pi t/365)$ for seasonality
- Linear trend: $(t - t_{\text{mean}})/t_{\text{std}}$ for long-term changes

Spatial Normalization:

- Latitude: $(lat - 25^\circ)/5^\circ$ (center at domain middle, scale by half-width)
- Longitude: $(lon + 55^\circ)/5^\circ$ (center and scale similarly)

Branch-Trunk Feature Separation:

- Branch (input function): Pressure and salinity interleaved: $[p_1, s_1, p_2, s_2, \ldots]$
- Trunk (query coordinates): $[\text{lat}, \text{lon}, t]$
- All features z-score normalized: $(x - \mu)/\sigma$ using training set statistics

Sequence Generation for LSTM

Temporal Clustering Algorithm:

1. Sort all profiles by timestamp
2. Group profiles within 30-day windows from same float
3. Create sequences of length $L = 5$ with stride 1
4. Discard sequences with temporal gaps >15 days

Sequence Statistics:

- Total sequences: 6,611 (from 7,039 individual profiles)
- Average temporal spacing: 10.2 days between consecutive profiles
- Sequence coverage: 94% of test profiles included in sequences

Data Splitting Strategy Details

Temporal Splitting:

- Sort profiles chronologically
- Train: Days 0–255 (70%)
- Validation: Days 256–291 (10%)
- Test: Days 292–365 (20%)
- Minimum 30-day gap between train-validation and validation-test

Split Validation:

- No temporal overlap between splits
- Seasonal representation in each split verified
- Spatial distribution balanced across splits

Normalization and Standardization

Feature Scaling: Z-score standardization computed on training set:

$$x_{\text{norm}} = \frac{x - \mu_{\text{train}}}{\sigma_{\text{train}}} \tag{32}$$

Feature Statistics:

6 Training Procedures and Computational Details

Model Training Configuration

Optimizer Settings:

- Optimizer: Adam
- Learning rate: 10^{-3} (constant)
- $\beta_1 = 0.9$, $\beta_2 = 0.999$
- Weight decay: 10^{-5}
- Gradient clipping: max norm 1.0

Table 6: FNO Feature Attribution by XAI Method (Mean ± Std)

Method	Pressure	Salinity	Latitude	Longitude	Time
Integrated Gradients	0.584±0.829	0.526±0.405	0.157±0.203	0.028±0.024	0.037±0.034
Saliency Maps	0.678±0.931	0.696±0.443	0.113±0.098	0.089±0.139	0.055±0.039
DeepLIFT	0.366±0.594	0.421±0.419	0.108±0.121	0.048±0.076	0.038±0.038
GradientSHAP	0.607±0.866	0.529±0.405	0.163±0.218	0.029±0.026	0.036±0.035

Table 7: LSTM Feature Attribution by XAI Method (Mean ± Std)

Method	Pressure	Salinity	Latitude	Longitude	Time
Integrated Gradients	0.539±0.717	0.516±0.407	0.176±0.239	0.031±0.026	0.039±0.037
Saliency Maps	0.603±0.806	0.666±0.448	0.112±0.094	0.088±0.139	0.054±0.038
DeepLIFT	0.338±0.531	0.416±0.423	0.121±0.139	0.051±0.080	0.040±0.040
GradientSHAP	0.561±0.774	0.523±0.408	0.183±0.257	0.032±0.028	0.038±0.038

Table 8: XAI Method Correlation (All p¡0.001)

	IG	Saliency	DeepLIFT	GradSHAP
Integrated Gradients	1.00	0.88	0.91	0.97
Saliency Maps	0.88	1.00	0.85	0.90
DeepLIFT	0.91	0.85	1.00	0.89
GradientSHAP	0.97	0.90	0.89	1.00

Table 9: Feature Ranking Consistency

Feature	Rank 1	Rank 2	Rank 3	Rank 4	Rank 5
Pressure	15/16	1/16	0/16	0/16	0/16
Salinity	1/16	13/16	2/16	0/16	0/16
Latitude	0/16	2/16	11/16	2/16	1/16
Longitude	0/16	0/16	2/16	9/16	5/16
Time	0/16	0/16	1/16	5/16	10/16

Table 10: DeepONet Branch Network Layer Conductance

Layer	Mean	Std Dev	Min	Max
FC-1 (input)	0.0122	0.0034	0.0089	0.0165
FC-2	0.0098	0.0028	0.0071	0.0128
FC-3	0.0109	0.0031	0.0079	0.0142
FC-4 (output)	0.0104	0.0029	0.0076	0.0135

Table 11: DeepONet Trunk Network Layer Conductance

Layer	Mean	Std Dev	Min	Max
FC-1 (input)	0.0052	0.0018	0.0036	0.0071
FC-2	0.0024	0.0009	0.0016	0.0034
FC-3	0.0018	0.0007	0.0012	0.0026
FC-4 (output)	0.0037	0.0013	0.0025	0.0052

Training Hyperparameters:

- Batch size: 128 (training), 256 (evaluation)
- Epochs: 100 maximum
- Early stopping patience: 15 epochs
- Validation frequency: Every epoch
- Loss weights: $\lambda_{\text{PDE}} = 0.1$, $\lambda_{\text{TS}} = 0.05$

Hardware Configuration:

- GPU: T4(Karggle)
- Framework: PyTorch 2.0.1, CUDA 11.8
- Python: 3.11

Reproducibility:

- Random seeds: 0–9 for all experiments
- Deterministic algorithms enabled where possible
- cudNN deterministic mode: True
- cudNN benchmark mode: False

Physics-Informed Loss Implementation

PDE Residual Computation:

The PDE residual requires automatic differentiation with respect to trunk inputs. Implementation details:

1. Enable gradient computation on trunk inputs: `trunk_input.requires_grad_(True)`
2. Forward pass through network: $\hat{T} = f(\text{branch}, \text{trunk})$
3. Compute first derivatives: $\frac{\partial \hat{T}}{\partial t}$, $\frac{\partial \hat{T}}{\partial p}$
4. Compute second derivative: $\frac{\partial^2 \hat{T}}{\partial p^2}$ using `create_graph=True`
5. Calculate residual: $r_{\text{PDE}} = \frac{\partial \hat{T}}{\partial t} - \kappa \frac{\partial^2 \hat{T}}{\partial p^2}$
6. Loss term: $\mathcal{L}_{\text{PDE}} = \frac{1}{N} \sum_{i=1}^{N} r_{\text{PDE}}^2$

Gradient Flow Considerations:

- Use full Jacobian computation (not per-column gradients)
- Detach trunk gradients after PDE computation to avoid gradient conflicts
- Apply gradient clipping after PDE backward pass

T-S Relationship Fitting:

Empirical temperature-salinity relationship fitted using polynomial regression on training data:

$$f_{\text{TS}}(s) = a_0 + a_1 s + a_2 s^2 \tag{33}$$

Fitted coefficients:

Table 12: FNO Layer Conductance

Layer	Mean	Std Dev	Min	Max
Lifting	0.0156	0.0041	0.0118	0.0203
Fourier-0	0.0189	0.0048	0.0144	0.0242
Fourier-1	0.0190	0.0049	0.0145	0.0244
Fourier-2	0.0165	0.0043	0.0125	0.0212
Fourier-3	0.0149	0.0039	0.0113	0.0191
Projection	0.0134	0.0035	0.0101	0.0172

Table 13: LSTM Layer Conductance

Layer	Mean	Std Dev	Min	Max
LSTM Forward	0.0183	0.0052	0.0135	0.0245
LSTM Backward	0.0179	0.0051	0.0132	0.0239
FC-1 (hidden)	0.0197	0.0055	0.0146	0.0264
FC-2 (output)	1.2162	0.3402	0.8956	1.6284

- $a_0 = -127.45$
- $a_1 = 8.23$
- $a_2 = -0.089$
- Tolerance: $\epsilon = 2.0$°C

Training Loop Implementation

Training Procedure per Epoch:

1. Shuffle training data
2. For each batch:
   - Load branch inputs, trunk inputs, targets
   - Zero gradients
   - Forward pass (enable PDE gradients for physics-informed models)
   - Compute data loss: $\mathcal{L}_{\text{data}}$
   - If physics-informed: Compute $\mathcal{L}_{\text{PDE}}$ and $\mathcal{L}_{\text{TS}}$
   - Total loss: $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{data}} + \lambda_{\text{PDE}}\mathcal{L}_{\text{PDE}} + \lambda_{\text{TS}}\mathcal{L}_{\text{TS}}$
   - Backward pass
   - Gradient clipping (max norm 1.0)
   - Optimizer step
3. Validation loop (no gradients):
   - Evaluate on validation set
   - Compute validation loss and metrics
   - Check early stopping criterion
4. Learning rate scheduling (optional): ReduceLROnPlateau with patience 5
5. Model checkpointing: Save best validation loss model

Early Stopping Logic:

- Monitor: Validation RMSE
- Patience: 15 epochs
- Minimum improvement: 0.001°C
- Restore best weights after early stop

Table 14: Training Set Feature Statistics

Feature	Mean	Std	Range
Pressure (dbar)	125.4	89.7	[2.3, 498.9]
Salinity (PSU)	36.82	0.34	[35.89, 37.65]
Temperature (°C)	19.64	4.12	[8.23, 27.41]
Latitude (°N)	25.03	2.87	[20.01, 29.98]
Longitude (°W)	-55.12	2.91	[-59.97, -50.04]
Time (days)	127.3	73.8	[0, 255]

Deep Ensemble Training

Ensemble Coordinator:

For each architecture:

1. Loop over seeds 0–9:
   - Set global random seed
   - Initialize model with random weights
   - Train model to convergence (early stopping)
   - Save trained model checkpoint
   - Evaluate on test set
   - Store predictions and metrics
2. Aggregate ensemble predictions:
   - Ensemble mean: $\bar{T} = \frac{1}{10}\sum_{k=1}^{10}\hat{T}_k$
   - Ensemble variance: $\sigma^2 = \frac{1}{10}\sum_{k=1}^{10}(\hat{T}_k - \bar{T})^2$
3. Compute statistics across seeds:
   - Mean RMSE $\pm$ Std
   - Mean MAE $\pm$ Std
   - Mean R^2 $\pm$ Std

Scalability Analysis:

- Training time scales linearly with dataset size
- Inference throughput independent of training set size
- Memory scales with batch size and model parameters
- Physics-informed methods benefit from GPU parallelization of derivative computation

7 Complete Layer Conductance Analysis

DeepONet Layer-wise Conductance Values

Branch Network Layers:

Layer	Mean	Std Dev	Min	Max
FC-1 (input)	0.0122	0.0034	0.0089	0.0165
FC-2	0.0098	0.0028	0.0071	0.0128
FC-3	0.0109	0.0031	0.0079	0.0142
FC-4 (output)	0.0104	0.0029	0.0076	0.0135

Trunk Network Layers:

Layer	Mean	Std Dev	Min	Max
FC-1 (input)	0.0052	0.0018	0.0036	0.0071
FC-2	0.0024	0.0009	0.0016	0.0034
FC-3	0.0018	0.0007	0.0012	0.0026
FC-4 (output)	0.0037	0.0013	0.0025	0.0052

Layer Importance Ratios

Branch vs Trunk Ratios (DeepONet):

- FC-1: 2.35× (0.0122 / 0.0052)
- FC-2: 4.08× (0.0098 / 0.0024)
- FC-3: 6.06× (0.0109 / 0.0018)
- FC-4: 2.81× (0.0104 / 0.0037)

LSTM Output Dominance:

- Output layer vs intermediate: 61.7× (1.2162 / 0.0197)
- Output layer vs recurrent: 67.9× (1.2162 / 0.0179)

8 Detailed Data Preprocessing and Feature Engineering

Argo Data Acquisition

Raw Data Statistics:

- Initial profiles retrieved: 1,247
- Total measurements before QC: 89,456
- Spatial coverage: $10^\circ \times 10^\circ$ North Atlantic box
- Temporal coverage: 365 days (2011)
- Depth range: Surface to 2000m

Quality Control Pipeline

Final Dataset Statistics After QC:

- Valid measurements: 35,131 (39.3% retention rate)
- Unique profiles: 554 (44.4% retention rate)
- Average measurements per profile: 63.4
- Depth distribution: 0–500m with bias toward surface

Data Splitting Strategy

Split Validation:

- No temporal overlap between splits
- Minimum 30-day gap between train-validation and validation-test
- Seasonal representation in each split verified

9 Training Procedures and Computational Details

Model Training Configuration

Training Configuration:

- Optimizer: Adam with adaptive learning rate
- Initial learning rate: 10^{-3}
- Batch size: 128
- Maximum epochs: 50
- Early stopping patience: 15
- Loss weights: $\lambda_{\text{PDE}} = 0.1$, $\lambda_{\text{TS}} = 0.05$

Deep Ensemble Training

Ensemble Configuration:

- Number of seeds: 10 per architecture (40 total models)
- Random seed range: 0–9 for reproducibility
- Model checkpointing: Best validation RMSE saved
- Computational resources: NVIDIA GPU with CUDA-enabled PyTorch

XAI Analysis Implementation

XAI Configuration:

- Library: Captum (Kokhlikyan et al. 2020)
- Sample size: 100 randomly sampled test instances per model
- Integrated Gradients: 50 Riemann steps, zero baseline
- GradientSHAP: 10 baseline samples from Gaussian distribution
- Layer conductance: Applied to all hidden layers
- Temporal aggregation: Sum over timesteps for LSTM sequences