scientific data

OPEN

DATA DESCRIPTOR

Spatiotemporal upscaling of sparse air-sea pCO2 data via physics-informed transfer learning

Siyeon Kim[1]✉, Juan Nathaniel[2]✉, Zhewen Hou[1], Tian Zheng[1] & Pierre Gentine[2]

Global measurements of ocean *pCO_2* are critical to monitor and understand changes in the global carbon cycle. However, *pCO_2* observations remain sparse as they are mostly collected on opportunistic ship tracks. Several approaches, especially based on direct learning, have been used to upscale and extrapolate sparse point data to dense estimates using globally available input features. However, these estimates tend to exhibit spatially heterogeneous performance. As a result, we propose a physics-informed transfer learning workflow to generate dense *pCO_2* estimates that are grounded in real-world measurements and remain physically consistent. The models are initially trained on dense input predictors against *pCO_2* estimates from Earth system model simulation, and then fine-tuned to sparse SOCAT observational data. Compared to the benchmark direct learning approach, our transfer learning framework shows major improvements of up to 56-92%. Furthermore, we demonstrate that using models that explicitly account for spatiotemporal structures in the data yield better validation performances by 50-68%. Our strategy thus presents a new monthly global *pCO_2* estimate that spans for 35 years between 1982-2017.

Background & Summary

Robust and dense measurements of key climate variables are crucial for monitoring our rapidly changing climate and benchmarking Earth system models used for climate projections[1]. However, many of these datasets are often sparse in both space and time. Consequently, there are numerous efforts to upscale sparse observations into globally gridded estimates, which are useful for evaluating terrestrial[2–4], hydrological[5], atmospheric, and ocean processes[6].

One of the pioneering upscaling methods relies on parameter inference, sometimes referred to as data assimilation[7,8], where a physical model is tuned to best represent available observations. Yet, the presence of structural errors in the underlying model can limit the quality of prediction[9–11]. A more recent method attempts to fit machine learning models through *direct learning* that maps a set of routinely available input predictors (e.g., from weather stations or satellite observations) to target variables often measured by sparse observations[12–16]. However, there are some inherent problems to this learning approach as evidenced in our present task of upscaling partial pressure of carbon dioxide (pCO_2). First, the observed data tends to be very sparse i.e., pCO_2 measurements from SOCAT[6,17] cover only 1-2% of the global ocean. Second, predictions outside of the available observations could be out of distribution and unconstrained, so any machine learning models will require robust extrapolation capability. This could lead to significant biases and physically inconsistent predictions of undersampled regions or time periods (e.g., southern ocean). For example, many global ocean biogeochemistry models (GOBMs) highlight the increasing significance of the southern ocean as a key carbon sink[18,19], which might be misrepresented in unconstrained models solely fitted with sparse observations.

In this work, we attempt to resolve these issues by developing a physics-informed *transfer learning* framework that can better extrapolate beyond available observations, by first (1) pre-training models on physical priors encoded in the outputs of GOBMs to ensure physical consistency, and (2) fine-tuning them using sparse observations to ensure real-world groundedness (Fig. 1). For both learning approaches, i.e., *direct* and *transfer*, we evaluate our models on pCO_2 data from held-out SOCAT tracks (*interpolation*) and pCO_2 estimates from an unseen member of GOBM (*extrapolation*). As demonstrated, we find that transfer learning improves

[1]Department of Statistics, Columbia University, New York, NY, 10027, USA. [2]Department of Earth and Environmental Engineering, Columbia University, New York, NY, 10027, USA. ✉e-mail: sk4973@columbia.edu; jn2808@columbia.edu

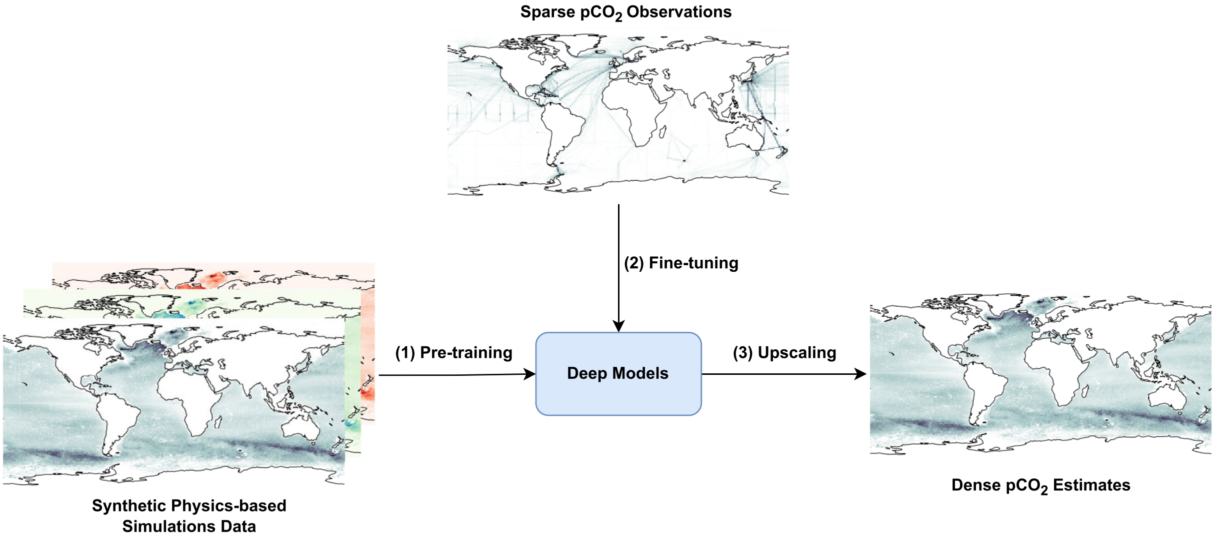


Fig. 1 Our proposed physics-informed transfer learning framework where we first pre-train deep models using synthetic physics-based simulation data from GOBMs. Then we fine-tune our models with sparse SOCAT observations in order to make a physically-informed, observation-grounded pCO_2 dense estimates.

direct learning by 56-92%. Furthermore, using models that explicitly account for spatiotemporal structures in the data improve those that do not by 50-68%. Thus, we present a new monthly global pCO_2 estimate that is physically-informed, observationally-grounded, and spatiotemporally-consistent, spanning for 35 years between 1982 and 2017.

Methods

Data Processing. Here, we describe the data used for our transfer learning approach during both pre-training and fine-tuning phases (direct learning uses identical data but skips the pre-training phase). In general, during the pre-training phase, we use a set of input predictors to target pCO_2 estimates from several GOBM ensemble members to learn meaningful physical knowledge. In the second phase, we fine-tune our pre-trained models using the same set of input predictors but targeting sparse SOCAT pCO_2 measurements[6]. We next discuss the choice of input predictors and the associated data preprocessing steps. All dataset used to train and evaluate our models, including predictors and GOBM estimates, are available and extensively discussed in the Large Ensemble Testbed[20].

Climate Model Data. We use the Large Ensemble Testbed[20], a comprehensive testing framework to assess pCO_2 reconstruction. The most significant merit of the testbed lies in its diverse representation of pCO_2 estimates from 25 ensemble members across four independent GOBMs[21] including CanESM[22], CESM[23], ESM2M[24], and MPI[25]. This enables us to analyze both between-model and within-model variability and improve the reliability of extrapolation capability of models, creating a conducive pre-training and testing ground for our work. Each ensemble member includes estimates with 421 monthly time steps from January 1982 to January 2017, forced by the RCP8.5 emission scenario, and bi-linearly interpolated to a 1° × 1° grid resolution.

SOCAT Data. The Surface Ocean Carbon Dioxide Atlas (SOCAT)[6] is a collaborative initiative of the international ocean carbon research community, with over 100 contributors, focused on the compilation of quality-controlled surface ocean fugacity of carbon dioxide observations. SOCAT facilitates the measurement of oceanic carbon sinks and the assessment of ocean acidification, in addition to evaluating marine biogeochemical models. All data within SOCAT comes from ships, drifters, autonomous surface platforms, and moorings that measure surface pCO_2. The gaps or lack of data in SOCAT in certain regions or during specific time periods pose challenges for accurately monitoring and analyzing oceanic carbon dioxide levels. Particularly noticeable is the regional sparsity in the southern ocean, especially during winter, which limits the reconstruction of long-term variations in oceanic carbon dioxide levels. This data sparsity affects the accuracy of carbon cycle simulations and hinders the assessment of ocean carbon uptake over different timescales, thereby necessitating efforts to improve data extrapolation in under-sampled regions and periods.

Input Data. The main input predictors for our models include: Sea Surface Temperature (SST) from NOAA:OISSTv2[26] and Surface Chlorophyll-a (Chl-a) from ESA:GlobColour[27] satellites; Sea Surface Salinity (SSS) from a compilation of in-situ data sources (see Met Office EN4[28]); and atmospheric CO2 mixing ratio (xCO2) from NOAA:GLOBALVIEW station sites[29]. The listed features are known indicators affecting pCO_2[30]. Previously, Mixed Layer Depth (MLD) from Argo floats[31] has been widely used as one of the main indicators for pCO_2 predictions and is based on a physical model. However, since our model performance heavily depends on fully covered direct observation inputs, we replaced MLD with Distance to Coast (D2C). The main reason why

D2C was chosen is to account for the scale and rotation invariance of pooling layers within the model and also the impact on ocean circulation. More specifically, in traditional computer vision tasks, these invariant properties benefit the performance as the convolutional layer can provide the same output regardless of the orientation of the input image. However, in the context of global ocean, such rotational invariance property can be harmful. The orientation of the continents, coastlines and oceans can provide important contextual information for interpretation and analysis and ocean circulation, affecting pCO_2. For our task, each pixel should hold specific spatial information, and the model should not be invariant to the rotation of images. Incorporating the D2C variable provides such direct spatial information on the spatial element, and we chose to incorporate it. Based on our experiments, replacing MLD with D2C yielded similar results and improved prediction along the coastline. This enhancement can be attributed to numeric values within continents, unlike other predictors, which can help the model compute gradients along the coastline.

Data Preprocessing. The monthly input and target data spanning between 1982 and 2017 are transformed into a tabular form where each 1° × 1° ocean pixel grid cell is considered as one observation. The latitude and longitude are also included for training baseline models, in addition to the five inputs to provide some spatial context to the model as they are useful to improve performance (i.e., CHL, D2C, SSS, SST, xCO2, latitude, and longitude). These inputs are then scaled using Min-Max scaling which ranges from 0 to 255, identical to the 8-bit image pixel value range. The continents are masked with 0 and are not factored into the loss calculation during the training process. The only exception is the D2C variable which includes values within the continents; these values within continents improve the model performance as described earlier. Given that xCO2 is represented as a single value for each time frame, it was encoded as an image with a single value, which was scaled relative to other time frames. Based on our ablation works, we find that including xCO2 improves the performance of the prediction.

Modeling Framework. The final choices of our modeling framework are supported by a k-fold cross validation approach (k=7) that randomly permute a wide range of model parameters (e.g., number of trees in RF, number of hidden layers in deep models) and their fitting hyperparameters (e.g., learning rate) to find the best performing set of configurations[32].

Baseline Models. Random Forest (RF)[33], Extreme Gradient Boosting (XGBoost)[34], and Feed Forward Neural Network (FFN)[35] are used as our baseline models. These models are selected for their reliability and universality across multiple prediction and upscaling tasks[36]. Unlike in previous studies, we do not divide the regions or include separate parameters to account for time to provide an equal testing environment for all models.

For RF, the final parameter set includes N_ESTIMATORS=20, MAX_DEPTH=10, and the remaining ones as defaulted by the `scikit-learn` package. For XGBoost: n_estimators=30, η=0.05, and the remaining ones as defaulted by the `xgboost` package. Finally, the FFN has 2 hidden layers of sizes [1024, 512], activated by *ReLU*, with a dropout rate of [0.2, 0.1], and optimized using *Adam*[37] with a learning rate of 10^{-3}.

Spatiotemporal Deep Models. The main models used in the work consist of U-NET[38] and Convolutional LSTM[39]. These models capture spatial-only and spatiotemporal information respectively. ConvLSTM, in particular, aims to generate not only a spatially consistent, but also temporally meaningful global pCO_2 estimate based on a set of fully contiguous inputs.

U-NET. The U-NET image segmentation model takes the form of a conventional encoder-decoder architecture. The U-NET is well known for its special U-shaped structure: the contraction part becomes U's left arm, and the expansion part becomes U's right arm. The contraction part uses max pooling operation to compress the image step by step but retain important information at different scales. Conversely, the expansion part expands the compressed information step by step and outputs predictions for different tasks. In Computer Vision, image segmentation models create a label for individual pixels in images to distinguish the borders of objects, for example. Recently, U-NET has been proven to be useful in regression tasks of images[40,41]. Our problem attempts to predict one pCO_2 value for each grid cell in a supervised regression manner. This can be done by changing the activation function into Exponential Linear Unit (*ELU*)[42] instead of the usual *Sigmoid* function. The choice of appropriate activation function strongly impacts the model performance. Specifically, the *ReLU* activation function frequently yields vanishing gradients, so it was imperative to use Leaky Rectified Linear Unit *ReLU* or *ELU* functions. Our final U-NET configuration consists of 6 convolution and deconvolution blocks with hidden channels of size [32, 32, 64, 64, 128, 128], with a strides of 5, activated by *ELU*. Similar to its FFN counterpart, U-NET is optimized using *Adam*[37] with a learning rate of 10^{-3}.

ConvLSTM. The ConvLSTM model was first used in precipitation nowcasting[39] which was then used for video-based action recognition and next-frame prediction tasks due to its ability to encode and predict spatiotemporal information. The model combines Long Short-Term Memory (LSTM), a popular recurrent neural network algorithm that can learn long-term dependencies, and Convolutional Neural Network (CNN), an image processing network to encode images to be fed into the LSTM cell. The LSTM model can retain long-term memory through memory cells and control gates to choose which gradients to be contained or passed on along adjacent cell, preventing gradients from vanishing. Our final ConvLSTM configuration consists of 4+1 ConvLSTM+Conv3D layers with hidden channels of size [32, 32, 32, 32, 1], strides of [5, 5, 3, 1, 3], activated by *ELU*. Similar to its FFN counterpart, ConvLSTM is optimized using *Adam*[37] with a learning rate of 10^{-3}.

Notation. We denote the stacked images of CHL, D2C, SSS, SST, xCO2 as $\mathbf{X} \in \mathbb{R}^{T \times H \times W \times C}$, where T, H, W, C are the batch size (along time dimension), height, width, and channel dimensions (number of features)

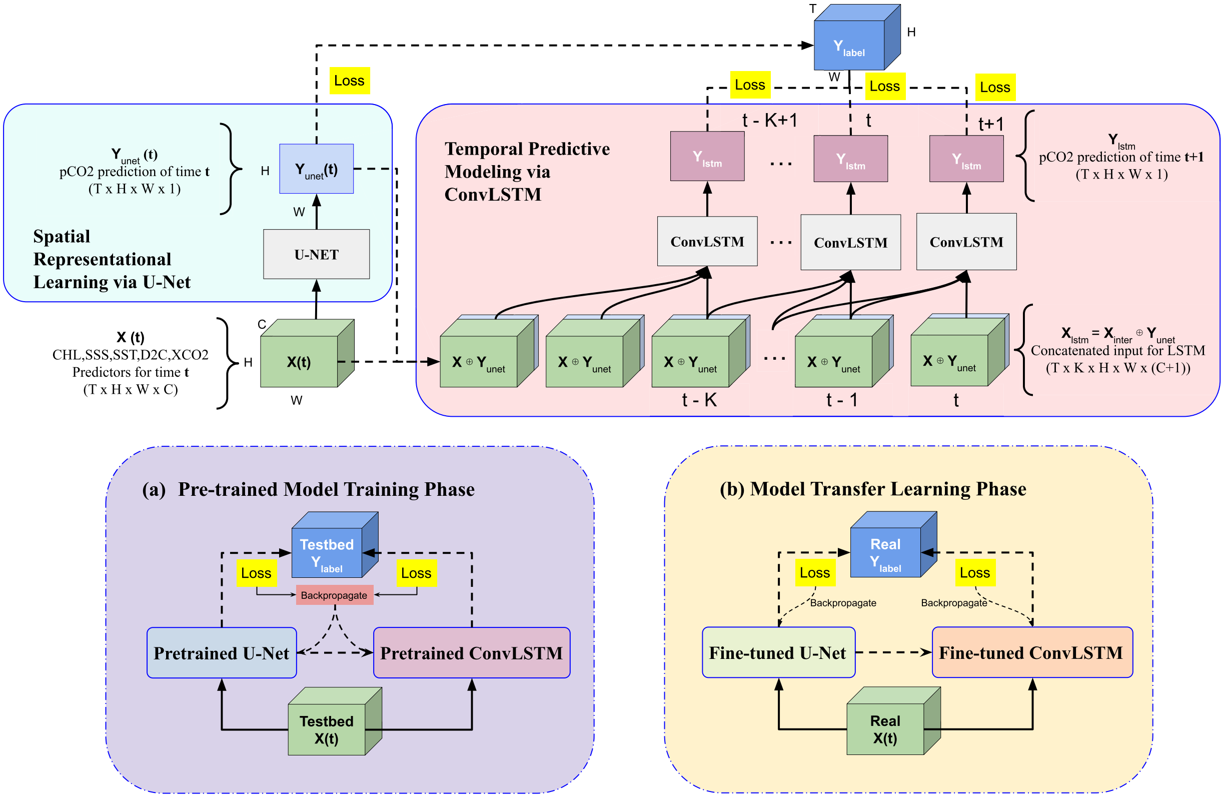


Fig. 2 The general step-by-step procedure of the training process. (**a**) The pre-trained model training phase; using five members of GOBM (e.g., CESM, MPI, CanESM) dense estimate, the input features are transformed into images to be fed into a U-NET model, and the output is concatenated for the ConvLSTM model. The model is updated with the summation of the loss from the U-NET and ConvLSTM model. K denotes the number of frames embedded in the input: $\{t-K, t-K+1, \ldots, t-1\}$. (**b**) The model fine-tuning phase; this process marks how the pre-trained model from the previous part will be applied to the real world data (sparse SOCAT tracks). The workflow is divided into two separate processes, but can be combined end-to-end.

respectively. We denote the pCO_2 as $\mathbf{Y} \in \mathbb{R}^{T\times H\times W\times 1}$. Suppose $\mathbf{X}_{\text{testbed}}$ and $\mathbf{Y}_{\text{testbed}}$ represents the data from ensemble testbed and $\mathbf{X}_{\text{real}}$ and $\mathbf{Y}_{\text{real}}$ represents the data from the SOCAT tracks.

Suppose the input and the output of U-Net is $\mathbf{X}_{\text{unet}} \in \mathbb{R}^{T\times H\times W\times C}$ and $\mathbf{Y}_{\text{unet}} \in \mathbb{R}^{T\times H\times W\times 1}$. We concatenate this two tensors in the last dimension to get $\mathbf{X}_{\text{inter}} = (\mathbf{X}_{\text{unet}}, \mathbf{Y}_{\text{unet}}) \in \mathbb{R}^{T\times H\times W\times (C+1)}$. We could transform $\mathbf{X}_{\text{inter}}$ with sliding window into the input of the ConvLSTM $\mathbf{X}_{\text{lstm}} \in \mathbb{R}^{T\times K\times H\times W\times (C+1)}$, where K is the number of contiguous frames embedded in the input (see Fig. 2). Finally the output of the ConvLSTM is $\mathbf{Y}_{\text{lstm}} \in \mathbb{R}^{T\times K\times H\times W\times 1}$.

Transfer Learning for pCO_2 Prediction. We describe our applications of (spatio)-temporal models to Earth system model data and SOCAT data using a two-phase framework: (1) pre-training on an independent GOBM and (2) fine-tuning on sparse SOCAT pCO_2 observations. Afterward, we will describe our evaluation strategy.

Transfer Learning Phase I: Pre-Training. In total, 5 ensemble members of GOBM (e.g., CESM[23]) are used for training and 1 for testing. In particular, we perform similar transfer learning procedure on CESM, MPI, and CanESM. The 5+1 members are chosen at random out of 25 members total available for each GOBM in the Large Ensemble Testbed[21]. Including additional members to train the model minimally slightly improved performance but heavily elongated the learning time.

First, we fit U-NET with $\mathbf{X}_{\text{unet}}$. The model then outputs pCO_2 prediction, $\mathbf{Y}_{\text{unet}}$. As hinted earlier, our problem of pCO_2 upscaling can be reduced to a next-frame prediction task. Thinking of the ocean pCO_2 as a sequence of images, the ConvLSTM model can be used to predict the next frame given a sequence of input frames. Here, the input frames include both the conventional input features as images and fully constructed pCO_2 frames from the U-NET model reconstruction. To be precise, the inferred pCO_2, $\mathbf{Y}_{\text{unet}}$, gets appended to the input of the ConvLSTM model as an extra channel: $\mathbf{X}_{\text{inter}}$. To embed the temporal component in ConvLSTM, we transform $\mathbf{X}_{\text{inter}}$ with sliding window. This transformation makes the final input, $\mathbf{X}_{\text{lstm}}$. Learning the sequential variability and seasonality between each time step, the ConvLSTM model will use information from $\{t-K, t-K+1, \ldots, t-1\}$ time frames to predict pCO_2 values at $\{t-K+1, t-K+2, \ldots, t\}$ time frames respectively. Unless otherwise specified, the K in this work is set to 3, limited by the lack of memory and high computational cost. With all these components, the final ConvLSTM model takes in $\mathbf{X}_{\text{lstm}}$ and outputs $\mathbf{Y}_{\text{lstm}}$. This concludes the pre-training phase using synthetic ensemble GOBM data.

Overall, the pre-training process is generally sensitive to learning rates and batch sizes; the optimized final learning rate was 0.001 with an *Adam* optimizer and batch size of 16. The best-performing model had 1,105,376 parameters; deeper models did not significantly improve the performance. We set the hyperparameters and training scheme for the ConvLSTM to be identical to those of the U-NET model.

Transfer Learning Phase II: Fine-Tuning. Until this point, the pre-trained model is fitted with synthetic dataset (i.e., based on simulations from GOBMs). Due to the sparsity of pCO_2 measurement in the real ocean, full coverage data is unavailable in real-world applications, i.e., we use the same input predictors in both phases, but different target sources. However, fully covered input features (e.g., SST) are available from satellites and observational products. This allows the input features of the fine-tuning phase to be identical to the pre-training phase. The only distinction is that during fine-tuning, we update and backpropagate (i.e. re-optimizing the weights and biases of the neural networks) from the loss computed using only sparse SOCAT pCO_2 tracks (i.e. the loss is computed only on the sparse data points but the inputs are contiguous in space and time). In other words, we only update the pre-trained model using the very sparse pCO_2 observed data, yet leverage the fact that the input features are still contiguous in space and time and that the pre-trained model already capture important physically-informed spatiotemporal features. This method allows the model to fully utilize sparse observations while maintaining the space-time correlation initially informed by the synthetic dataset. If we were to follow such a process on the real-world data directly (i.e., direct learning), the target would be too sparse to fit a robust (spatio)-temporal model. Furthermore, due to the reduced number of pCO_2 measurements, it is difficult to check and measure the extrapolative performance and effectiveness of the model. Hence, leveraging prior information encoded in different GOBMs, such as CESM, MPI, and CanESM, is needed to ensure physical consistency.

As noted in Fig. 2, the fine-tuning phase follows a similar two-step process described earlier in pre-training, where we fine-tune (1) U-NET: $\mathbf{X}_{\text{unet}} \rightarrow \mathbf{Y}_{\text{unet}}$, and (2) ConvLSTM: $\mathbf{X}_{\text{lstm}} \rightarrow \mathbf{Y}_{\text{lstm}}$ using sparse SOCAT pCO_2 tracks. Because the model's weight is being updated on sparse SOCAT tracks, the model can lose its extrapolation capability by overfitting on the sparse data. It is easier to detect where such laziness happens in the two-step fine-tuning process. Moreover, the two-step process allows us to analyze the performance differences between the U-NET (spatial only) and the ConvLSTM (spatiotemporal) models. Nonetheless, since we have noted earlier that the fine-tuning phase takes identical steps to that of pre-training, both can be reduced to a single end-to-end framework if necessary. We unfreeze the last four layers for the fine-tuning of U-NET and the final two layers for ConvLSTM. Therefore only 2-5% of the parameters are trainable, and the rest are unchanged from the pre-trained model. If we increase the percentage of trainable parameters, the training objective will significantly improve. However, we will experience overfitting and might lose the extrapolation capability by focusing solely on available tracks and ignoring previously learned physical spatiotemporal representations from synthetic dataset during the pre-training phase. We implement several strategies, including early stopping, to prevent overfitting discussed next.

Evaluation. We adapt the mean squared error (MSE) as the loss function that the models are optimized on. In particular, we mask out the continents from the loss calculation (i.e., applying a zero mask). The root mean squared error (RMSE) is then used as a metric to compare the performance across models and experimental setups. In order to avoid overfitting, we also apply early stopping during training that tracks the validation error with a patience of 5 epochs. For the pre-training phase of our transfer learning framework, the predictions from both baseline and (spatio)-temporal models are compared with pCO_2 estimates from unseen GOBM members. For the fine-tuning phase (i.e, also models fitted with direct learning), we perform two evaluation:

- **Interpolation**. Evaluating how well the model is in predicting pCO_2 measurements from held-out SOCAT pCO_2 tracks.
- **Extrapolation**. Evaluating how well the model is in predicting pCO_2 estimates from unseen GOBM members.

The reason for performing two different set of evaluations is to first ensure that the fine-tuned model captures sufficient information from sparse SOCAT pCO_2 observations (*interpolation*) before being deployed in real-world application where actual observations are unavailable, but the estimates are (*extrapolation*).

Data Records

Dataset is freely available at https://zenodo.org/records/12726686[43]. We generated two globally-gridded, monthly pCO_2 estimates based on both U-NET and ConvLSTM (K=6) models using our proposed transfer learning approach. The provision of estimates from both models allows for ensemble estimates (i.e., different models might learn different spatiotemporal structures), enabling further analysis such as uncertainty quantification. Each data file is in Network Common Data Form (NetCDF) format, with a spatial dimension of 1° × 1° and monthly temporal resolution between 1982 and 2017. Estimates for ConvLSTM model starts 6 months later since we use the first $K = 6$ as inputs. The file naming convention is *global_pco2_monthly_1982_2017_<−MODEL_NAME>.nc*, where *<MODEL_NAME>* is one of *unet* or *convlstm*.

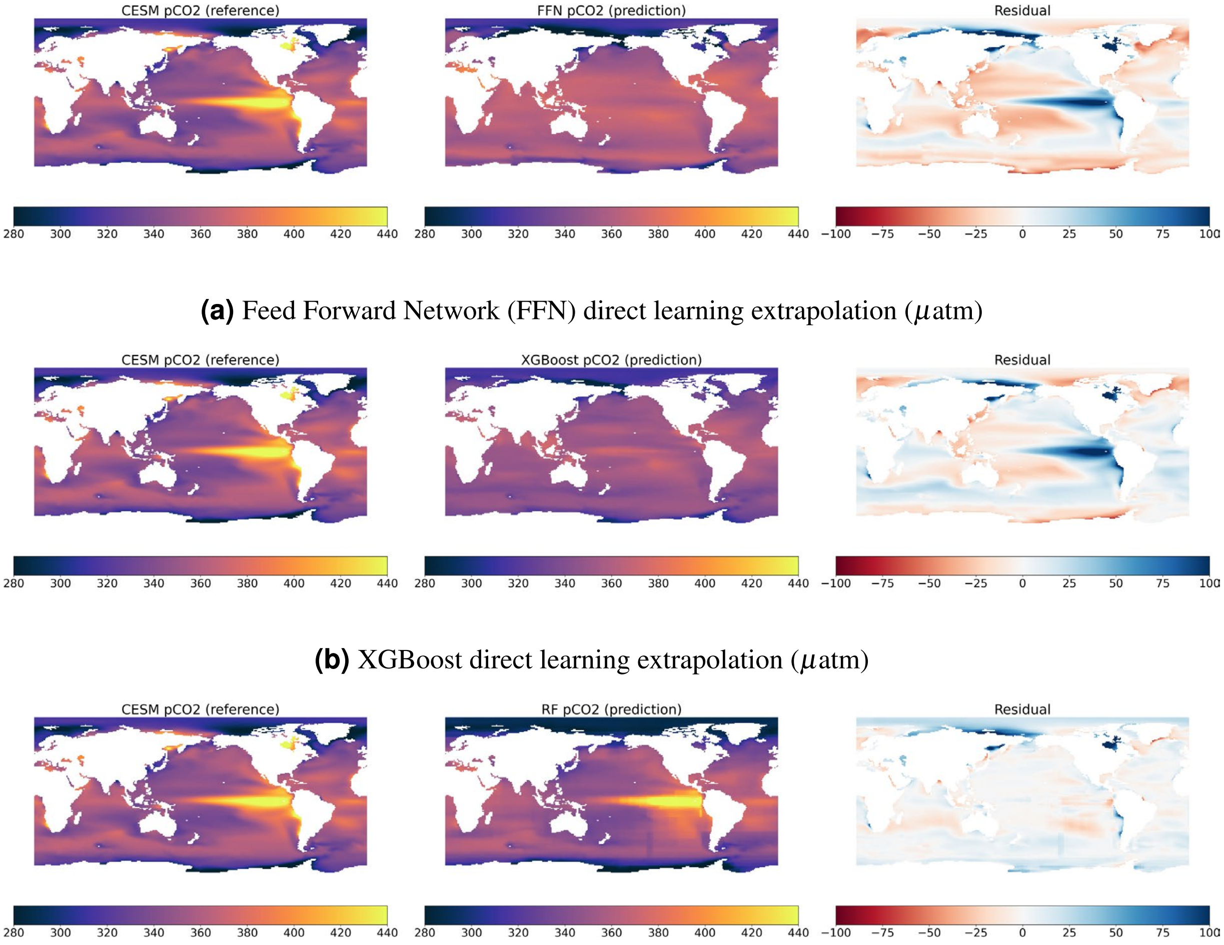
Technical Validation

Direct Learning.

First, we examine the performance of baseline models when trained directly on SOCAT data given dense input features, including CHL, D2C, SSS, SST, xCO2, latitude, and longitude. We then evaluate their performances on held-out SOCAT data (*interpolation*) and against unseen reference GOBM (e.g., CESM) ensemble member (*extrapolation*). In particular, we use FFN, XGBoost, and RF as our preliminary models, and CESM as our GOBM. The best among these, in terms of both interpolation and extrapolation results, will be used as baseline for subsequent analysis where a more sophisticated transfer learning approach is applied. As

Models	Interpolation RMSE (μatm)	Extrapolation RMSE (μatm)
FFN	33.32	35.75
XGBoost	31.40	32.82
RF	**20.15**	**31.37**

Table 1. Direct learning performance between different baseline models. The score is measured based on interpolation (i.e., pCO_2 measurements from held-out SOCAT data) and extrapolation (i.e., pCO_2 estimates from an unseen CESM member).



(a) Feed Forward Network (FFN) direct learning extrapolation (μatm)

(b) XGBoost direct learning extrapolation (μatm)

(c) Random Forest (RF) direct learning extrapolation (μatm)

Fig. 3 Qualitative extrapolative evaluation of predictions from baseline models trained directly, against reference simulation from an unseen member of CESM. The left column maps are the average pCO_2 values of the synthetic truth from CESM over time; the middle column maps are the average of the pCO_2 reconstruction from the baseline models (FFN, XGBoost, RF); the third column maps are the average residual values derived from subtracting the reference with the prediction.

summarized in Table 1 and illustrated in Fig. 3, we find that RF has the best interpolation and extrapolation performances, and will be subsequently used to baseline our transfer learning framework.

Transfer Learning. In contrast to direct learning, in transfer learning, we first (i) *pre-train* models to predict pCO_2estimates from physics-based simulations given dense input features including CHL, D2C, SSS, SST, xCO2, and (ii) *fine-tune* them using SOCAT data. Similarly, the interpolation and extrapolation results are evaluated using identical pCO_2 data/estimates from held-out SOCAT tracks and unseen GOBM ensemble member, respectively.

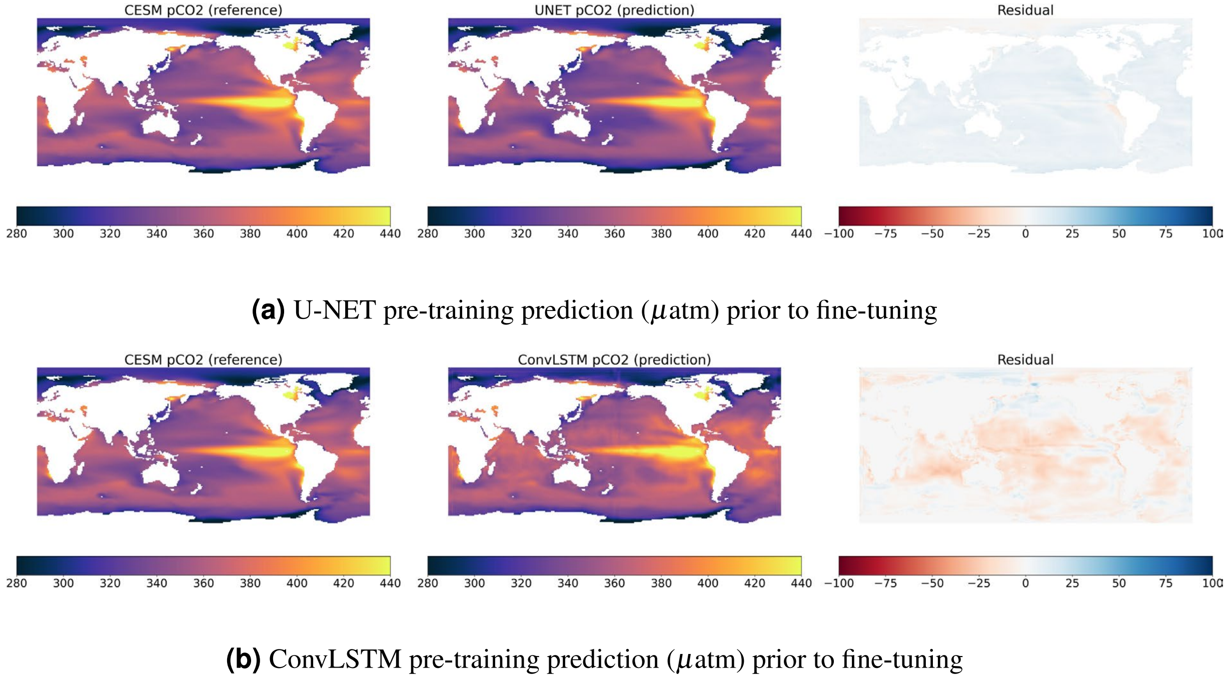
Pre-training evaluation. First, we compare the performance of different baseline and deep (spatio)-temporal models when trained on the Community Earth System Model (CESM) simulation to predict pCO_2 within and

Models	Training RMSE (μatm)	Testing RMSE (μatm)
RF	26.27	28.28
U-NET	**7.11**	**9.01** (↓ 68.16%)
ConvLSTM (K=1)	10.44	13.88 (↓ 50.92%)

Table 2. Pre-training performance between different models. The score is measured based on the training (5 members of CESM) and test data (1 unseen CESM member). Numbers in brackets indicate % improvements over RF baselines.

	Fine-tuned RMSE (μatm) (↓ is better)		
Pre-training	**RF (direct)**	**U-NET**	**ConvLSTM (K=1)**
CESM	20.15 / 31.37	**8.93 / 9.21**	17.72 / 19.53
MPI	18.24 / 32.51	43.00 / 41.08	**2.05 / 30.24**
CanESM	25.98 / 36.31	43.94 / 49.40	**2.08 / 31.86**

Table 3. Transfer learning performance of different models, including RF (direct), U-NET and ConvLSTM pre-trained on CESM/MPI/CanESM estimates and fine-tuned on SOCAT data. The moderate results given CESM pre-training motivates subsequent analysis, including as basis for the final data product. Scores are of interpolation / extrapolation RMSE.



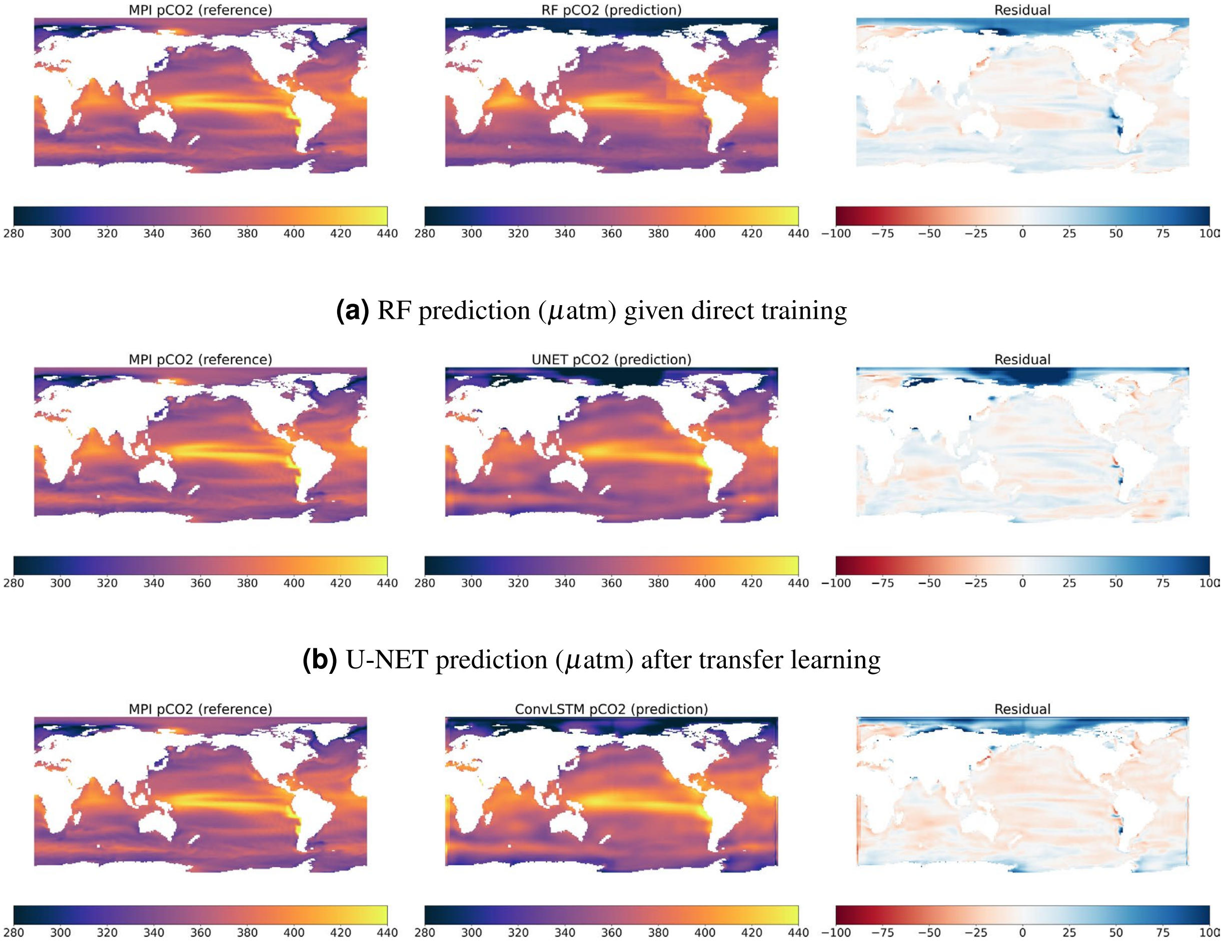
(a) U-NET pre-training prediction (μatm) prior to fine-tuning

(b) ConvLSTM pre-training prediction (μatm) prior to fine-tuning

Fig. 4 Qualitative evaluation of pre-trained models (U-NET and ConvLSTM), trained on the full synthetic dataset within the Community Earth System Model (CESM) simulation, prior to fine-tuning. The left column maps are the average pCO_2 values of the synthetic truth from the held-out CESM members over time; the middle column maps are the average of the pCO_2 reconstruction from the U-NET and ConvLSTM models; the third column maps are the average residual values derived from subtracting the reference with the prediction which highlight how well the models' capture spatial variability.

across different members. In essence, we evaluate the capacity of the models to predict the simulation results, which we refer to as the pre-trained model prediction. In this case, our models have access to contiguous temporal maps of synthetic pCO_2 for pre-training. As summarized in Table 2 and illustrated in Fig. 4, U-NET and ConvLSTM models are able to improve the RMSE score of RF by 68% and 51% on the test dataset respectively. The slightly worse performance of ConvLSTM compared to U-NET might be attributed to the former taking in the output of the latter, which results in error propagation (refer to the described process Fig. 2).

Fine-tuning evaluation. After pre-training, the final result and performance of the fine-tuned model on SOCAT is shown in Table 3 and Figs. 5–6. The fine-tuned U-NET and ConvLSTM models pre-trained on different synthetic data i.e., with CESM, MPI, CanESM, outperform the best baseline model (RF) trained directly on sparse



(a) RF prediction (μatm) given direct training

(b) U-NET prediction (μatm) after transfer learning

(c) ConvLSTM prediction (μatm) after transfer learning

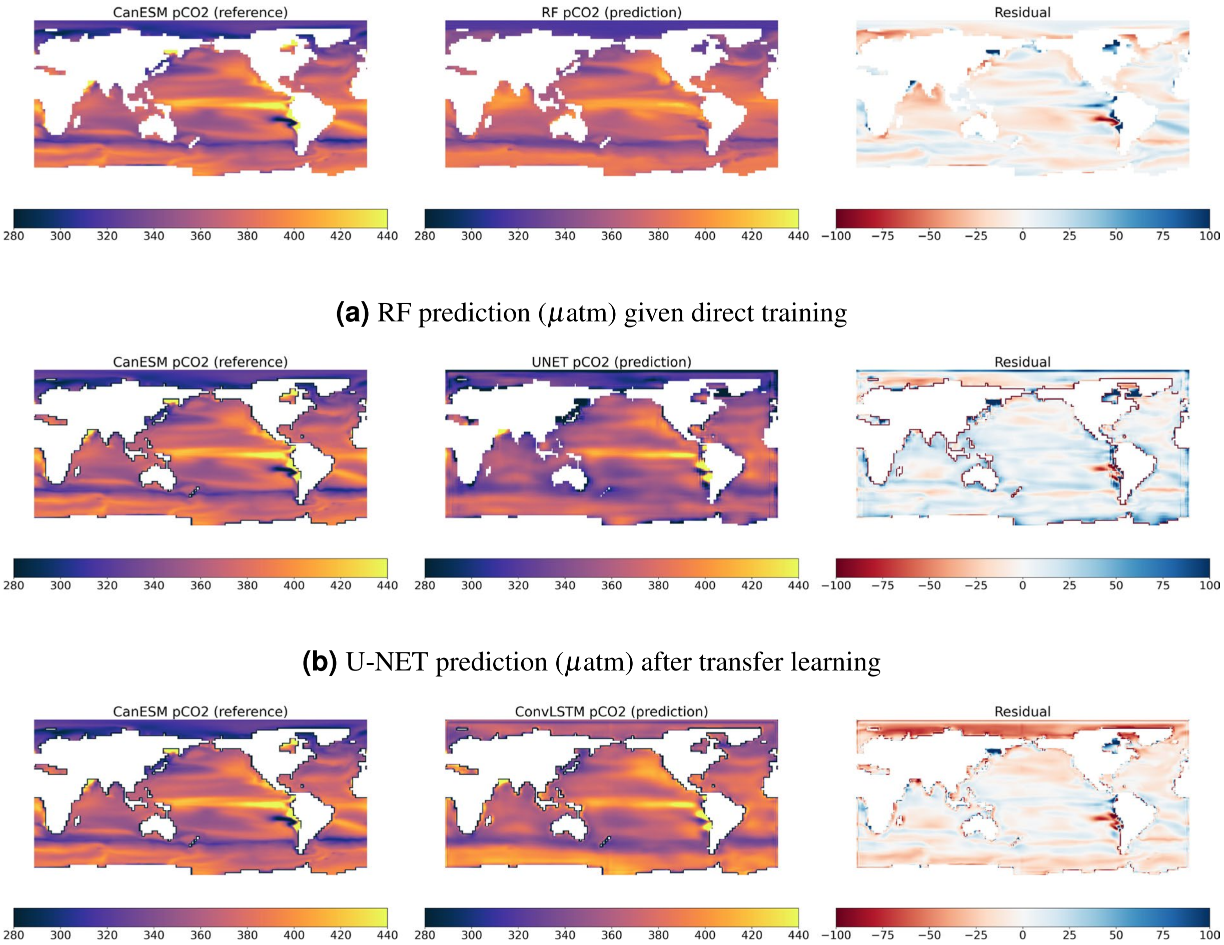
Fig. 5 Qualitative evaluation of fine-tuned model after pre-training on MPI, fine-tuned on sparse SOCAT tracks to mimic real-world application. Subfigures (**a–c**) showcase the performance of different models and their extrapolation capabilities given (left) average reference pCO_2 values, (middle) prediction, and (right) residual map over time derived from subtracting the reference with the prediction.

data without pre-training. In particular, the interpolation capabilities is improved by 56-92% with fine-tuning, showcasing the utility of encoding physical priors. Our study also demonstrates that the fine-tuned ConvLSTM (K=1) model is highly effective for predicting pCO_2 and capturing its spatiotemporal variability (except when pre-trained on CESM). These results have important implications as the model is able to capture improved understanding of climate patterns that exhibit complex spatiotemporal structures. Transfer learning involves using pre-trained models on similar tasks to improve performance on a new task, and in this case, the pre-training improved the ability of the model to learn the underlying relationships between different variables that influence pCO_2. As a result, the model was able to better capture the complex spatiotemporal patterns of pCO_2 in the ocean.

Based on this result after pre-training on different set of simulation data, the consistent improved performance is likely to carry over to real-world data as demonstrated in Table 3. This is because the model was evaluated using a range of physically-informed simulation dataset that were designed to mimic real-world conditions, and the findings indicated that the model was robust to interpolation and extrapolation cases. Nevertheless, measuring the true model's extrapolation capacity when real-world data is absent is nearly impossible, and must be interpreted with caution.

Ablation: Importance of Temporal Context.

As the final step of the work, we design an ablation study to understand the importance of incorporating temporal context. In particular, the experiment will look at pCO_2 prediction residual over time (Table 4 and Fig. 7) and across space (Fig. 8) for four different models. In an ideal scenario where temporal information is fully encoded, the temporal residuals (or errors) should exhibit characteristics similar to white noise: zero mean, constant variance, constant amplitude, no seasonality (randomness) and zero autocorrelation. Similarly, across space, the absolute residual should decrease with higher temporality: the number of K consecutive months used to make predictions. The four different models we consider include the best performing baseline i.e., RF, and ConvLSTM with different values of K, and are summarized as follow:



(a) RF prediction (μatm) given direct training

(b) U-NET prediction (μatm) after transfer learning

(c) ConvLSTM prediction (μatm) after transfer learning

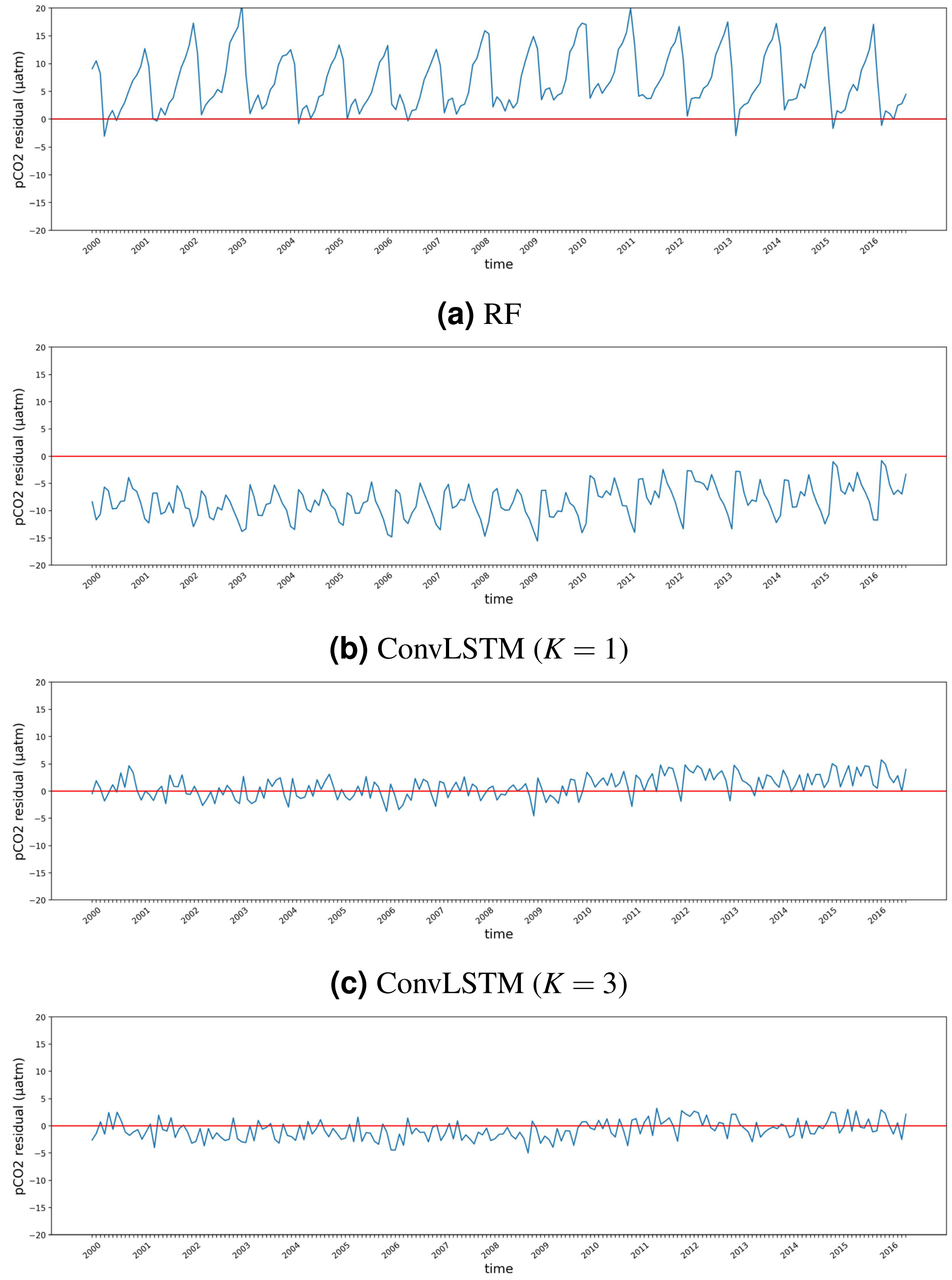
Fig. 6 Qualitative evaluation of fine-tuned model after pre-training on CanESM, fine-tuned on sparse SOCAT tracks to mimic real-world application. Subfigures (**a–c**) showcase the performance of different models and their extrapolation capabilities given (left) average reference pCO_2 values, (middle) prediction, and (right) residual map over time derived from subtracting the reference with the prediction.

1. *RF* (K=0) (baseline; direct learning),
2. *ConvLSTM* ($K = 1$), where $\{t-1\}$-th frame(s) will output pCO_2 prediction at time $\{t\}$,
3. *ConvLSTM* ($K = 3$), where $\{t-3,\ t-2,\ t-1\}$-th frame(s) will output pCO_2 prediction at time $\{t-2,\ t-1,\ t\}$; consists of 3-month temporality,
4. *ConvLSTM* ($K = 6$), where $\{t-6,\ t-5,\ \ldots,\ t-1\}$-th frame(s) will output pCO_2 prediction at time $\{t-5,\ t-4,\ \ldots,\ t\}$; consists of 6-month temporality. This final configuration has the most temporal information encoded.

Temporal Context on Error Correction. From Fig. 7, we can deduce how much each component of the model is responsible for the overall performance in pCO_2 upscaling and for the impact of temporal dynamics being captured. Examining the plot on subfigure (a) RF model shows a seasonal cycle on the residual, and performs worse than other models. This shows our concern that the point-wise model does not fully capture temporal information and provide seasonally biased prediction. Similar temporal patterns can be observed for subfigures (b) ConvLSTM with $K = 1$ and (c) ConvLSTM with $K = 3$ as limited amount of temporal information was incorporated. However, for subfigure (d) ConvLSTM with $K = 6$, the magnitude of the residual remains consistently low with minimal deviation away from zero even in long-term periodicity. Due to limited computational resource, we were unable to encode $K = 12$ and test the impact when we fully encode a year worth of observation frames. From Fig. 8, we observe that both RF and ConvLSTM ($K = 1$) have high absolute residual. The former also shows more discontinuity in space, as is expected from a non-spatial (point-wise) model. As we encode more temporality (i.e., subfigures (c) $K = 3$, and (d) $K = 6$), the absolute residual decreases, especially along the Pacific and Arctic ocean.

Model	Residual Average	Residual Variance
Random Forest	5.65	9.21
ConvLSTM (K=1)	−8.92	8.43
ConvLSTM (K=3)	0.19	4.75
ConvLSTM (K=6)	**0.10**	**3.22**

Table 4. Ablation of temporality on average residual over time across different model configurations.



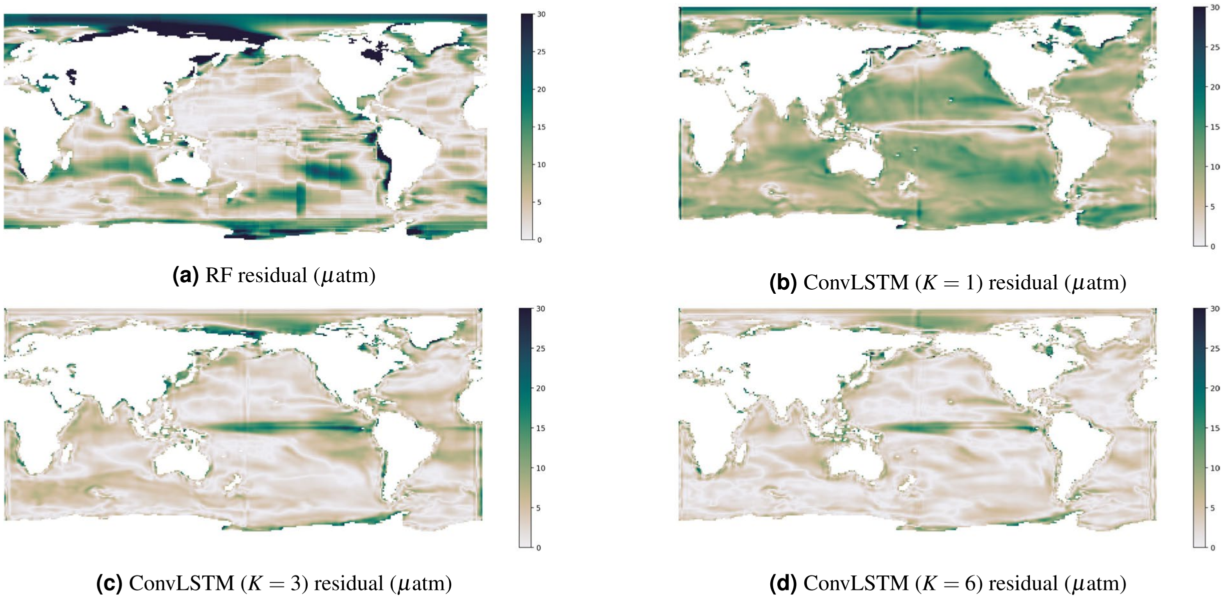
(a) RF

(b) ConvLSTM ($K = 1$)

(c) ConvLSTM ($K = 3$)

(d) ConvLSTM ($K = 6$)

Fig. 7 Accounting for more temporal context (larger K) reduces bias and magnitude of residuals in time, averaged across space. The x-axis describes time and y-axis the average residual.



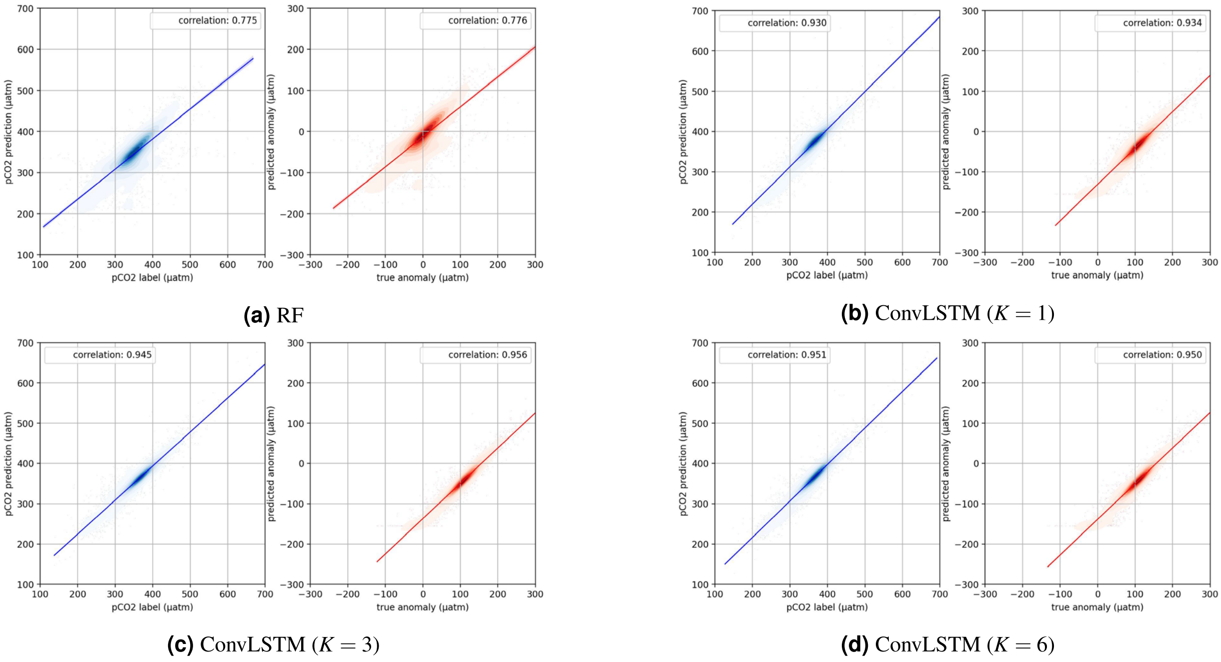
(a) RF residual (μatm)

(b) ConvLSTM ($K = 1$) residual (μatm)

(c) ConvLSTM ($K = 3$) residual (μatm)

(d) ConvLSTM ($K = 6$) residual (μatm)

Fig. 8 Accounting for more temporal context (larger K) reduces absolute residuals in space, averaged across time.



(a) RF

(b) ConvLSTM ($K = 1$)

(c) ConvLSTM ($K = 3$)

(d) ConvLSTM ($K = 6$)

Fig. 9 Ablation of temporality on the correlation between (left subpanels) pCO_2 label (μatm) and its prediction (μatm), (right subpanels) pCO_2 anomaly (μatm) and its prediction (μatm) across models.

Temporal Context on Detecting Anomalies. Finally, we study how adding more temporal information helps us in predicting pCO_2 anomaly – derived from subtracting mean climatology. As illustrated in Fig. 9, we observe that the correlation between pCO_2 prediction and truth are increasing as more temporality is included (*left* subpanels), from 0.775 in RF baseline to 0.951 in ConvLSTM ($K = 6$). However, this trend is not replicated in anomaly inference (*right* subpanels), where the correlation between pCO_2 anomaly truth and prediction plateaus at ConvLSTM ($K = 3$). This suggests the saturation of temporal information in making accurate prediction, especially for anomaly detection.

In conclusion, our study addresses the challenge of limited observations in the upscaling of carbon cycle, particularly ocean pCO_2, which is primarily reliant on sporadic and sparse ship-based data. Although various methods, including machine learning, have been used to extrapolate point data to a global scale, significant uncertainties still persist in these estimates. For one, the actual observations tend to be sparse, noisy, and biased, rendering direct learning inconsistent. In addition, many methods ignore the spatiotemporal variability that exist in the system by relying on point-wise models such as RF.

To overcome these limitations, we introduce a pre-trained model that treats dense input features and pCO_2 as video data frames, allowing us to capture both spatial and temporal autocorrelation. Our methodology employs image segmentation techniques like U-NET to predict pCO_2 values per pixel, with convolutional layers capturing spatial information. Furthermore, we implement a Convolutional LSTM (ConvLSTM) model, commonly used in video prediction, to capture temporal information. The model is pre-trained on Earth system model pCO_2 with full spatial coverage. By fine-tuning this model, through transfer learning, with actual sparse SOCAT data, we leverage the spatial and temporal correlations learned in the pre-trained phase, as oceanic variables are highly correlated in space and time.

Compared to the benchmark direct learning approach, our transfer learning framework shows major improvements of up to 56-92%. Furthermore, we demonstrate that using models that explicitly account for spatiotemporal structures in the data yield better validation performances by 50-68%. An ablation study also demonstrates the superiority of ConvLSTM model against baseline models in capturing long-term spatiotemporal dependency, especially with higher K. Also, the residuals of ConvLSTM do not demonstrate specific temporal or spatial structure unlike with baseline RF model, which suggest a robust representation of the spatiotemporal model. An interesting extension would be to expand $K > 6$ months to evaluate if there is an upper-limit to the extent of temporal information that can be captured. Nevertheless, our physics-informed transfer learning framework offers a pathway to robustly predict ocean carbon variables and beyond, by combining the strengths found even in sparse observations and imperfect model estimates. Despite the limitations of our physics-informed transfer learning approach (e.g., held-out SOCAT tracks used for interpolation validation might be dependent on those used for fine-tuning), we believe that it shows great promise, and future work should aim to better combine physics and sparse observations to further improve the method.

Code availability

The dataset used in this work, including the Large Ensemble Testbed, is available from[20]. Python was used for data processing, modeling, and final analysis, specifically using the $Tensorflow$[44] and $Scikit - learn$[45] packages for machine learning. The final code to reproduce the results is available on Github https://github.com/sk981102/ocean_co2/.

Received: 23 January 2024; Accepted: 1 October 2024;
Published: 8 October 2024

References

1. Righi, M. *et al*. Earth system model evaluation tool (esmvaltool) v2. 0–technical overview. *Geoscientific Model Development* **13**, 1179–1199 (2020).
2. Baldocchi, D. *et al*. Fluxnet: A new tool to study the temporal and spatial variability of ecosystem-scale carbon dioxide, water vapor, and energy flux densities. *Bulletin of the American Meteorological Society* **82**, 2415–2434 (2001).
3. Nathaniel, J., Liu, J. & Gentine, P. Metaflux: Meta-learning global carbon fluxes from sparse spatiotemporal observations. *Scientific Data* **10**, 440 (2023).
4. Friedlingstein, P. *et al*. Global carbon budget 2021. *Earth System Science Data* **14**, 1917–2005 (2022).
5. Addor, N., Newman, A. J., Mizukami, N. & Clark, M. P. The camels data set: catchment attributes and meteorology for large-sample studies. *Hydrology and Earth System Sciences* **21**, 5293–5313 (2017).
6. Bakker, D. C. *et al*. An update to the surface ocean co 2 atlas (socat version 2). *Earth System Science Data* **6**, 69–90 (2014).
7. Strebel, L., Bogena, H. R., Vereecken, H. & Hendricks Franssen, H.-J. Coupling the community land model version 5.0 to the parallel data assimilation framework pdaf: description and applications. *Geoscientific Model Development* **15**, 395–411 (2022).
8. Qu, Y., Nathaniel, J., Li, S. & Gentine, P. Deep generative data assimilation in multimodal setting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 449–459 (2024).
9. Anav, A. *et al*. Spatiotemporal patterns of terrestrial gross primary production: A review. *Reviews of Geophysics* **53**, 785–818 (2015).
10. Friedlingstein, P. *et al*. Climate–carbon cycle feedback analysis: results from the c4mip model intercomparison. *Journal of climate* **19**, 3337–3353 (2006).
11. Nathaniel, J. *et al*. Chaosbench: A multi-channel, physics-based benchmark for subseasonal-to-seasonal climate prediction. arXiv preprint arXiv:2402.00712 (2024).
12. Jung, M. *et al*. The fluxcom ensemble of global land-atmosphere energy fluxes. *Scientific data* **6**, 74 (2019).
13. Alemohammad, S. H. *et al*. Water, energy, and carbon with artificial neural networks (wecann): a statistically based estimate of global surface turbulent fluxes and gross primary productivity using solar-induced fluorescence. *Biogeosciences* **14**, 4101–4124 (2017).
14. Chen, S. *et al*. A machine learning approach to estimate surface ocean pco2 from satellite measurements. *Remote Sensing of Environment* **228**, 203–226 (2019).
15. Bennington, V., Galjanic, T. & McKinley, G. A. Explicit physical knowledge in machine learning for ocean carbon flux reconstruction: The pco2-residual method. *Journal of Advances in Modeling Earth Systems* **14**, e2021MS002960 (2022).
16. Skulovich, O. & Gentine, P. A long-term consistent artificial intelligence and remote sensing-based soil moisture dataset. *Scientific Data* **10**, 154 (2023).
17. Sabine, C. L. *et al*. Surface ocean co 2 atlas (socat) gridded data products. *Earth System Science Data* **5**, 145–153 (2013).
18. Gruber, N. *et al*. Trends and variability in the ocean carbon sink. *Nature Reviews Earth & Environment* **4**, 119–134 (2023).
19. Heimdal, T. H., McKinley, G. A., Sutton, A. J., Fay, A. R. & Gloege, L. Assessing improvements in global ocean pco 2 machine learning reconstructions with southern ocean autonomous sampling. *Biogeosciences Discussions* **2023**, 1–35 (2023).
20. Gloege, L., Yan, M., Zheng, T. & McKinley, G. A. Improved quantification of ocean carbon uptake by using machine learning to merge global models and pco2 data. *Journal of Advances in Modeling Earth Systems* **14**, e2021MS002620 (2022).
21. Gloege, L. *et al*. Quantifying errors in observationally based estimates of ocean carbon sink variability. *Global Biogeochemical Cycles* **35**, e2020GB006788 (2021).
22. Fyfe, J. C. *et al*. Large near-term projected snowpack loss over the western united states. *Nature communications* **8**, 14996 (2017).
23. Kay, J. E. *et al*. The community earth system model (cesm) large ensemble project: A community resource for studying climate change in the presence of internal climate variability. *Bulletin of the American Meteorological Society* **96**, 1333–1349 (2015).
24. Rodgers, K. B., Lin, J. & Frölicher, T. L. Emergence of multiple ocean ecosystem drivers in a large ensemble suite with an earth system model. *Biogeosciences* **12**, 3301–3320 (2015).
25. Maher, N. *et al*. The max planck institute grand ensemble: enabling the exploration of climate system variability. *Journal of Advances in Modeling Earth Systems* **11**, 2050–2069 (2019).

26. Reynolds, R. W., Rayner, N. A., Smith, T. M., Stokes, D. C. & Wang, W. An improved in situ and satellite sst analysis for climate. *Journal of climate* **15**, 1609–1625 (2002).
27. Maritorena, S., d'Andon, O. H. F., Mangin, A. & Siegel, D. A. Merged satellite ocean color data products using a bio-optical model: Characteristics, benefits and issues. *Remote Sensing of Environment* **114**, 1791–1804 (2010).
28. Good, S. A., Martin, M. J. & Rayner, N. A. En4: Quality controlled ocean temperature and salinity profiles and monthly objective analyses with uncertainty estimates. *Journal of Geophysical Research: Oceans* **118**, 6704–6716 (2013).
29. Conway, T. J. *et al*. Evidence for interannual variability of the carbon cycle from the national oceanic and atmospheric administration/climate monitoring and diagnostics laboratory global air sampling network. *Journal of Geophysical Research: Atmospheres* **99**, 22831–22855 (1994).
30. Landschützer, P. *et al*. A neural network-based estimate of the seasonal to inter-annual variability of the atlantic ocean carbon sink. *Biogeosciences* **10**, 7793–7815 (2013).
31. Roemmich, D. & Owens, W. The argo project: Global ocean observations for understanding and prediction of climate variability. *OCEANOGRAPHY-WASHINGTON DC-OCEANOGRAPHY SOCIETY-* **13**, 45–50 (2000).
32. Bergstra, J. & Bengio, Y. Random search for hyper-parameter optimization. Journal of machine learning research **13** (2012).
33. Ho, T. K. Random decision forests. In Proceedings of 3rd international conference on document analysis and recognition, vol. 1, 278–282 (IEEE, 1995).
34. Chen, T. & Guestrin, C. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 785–794 (2016).
35. Bebis, G. & Georgiopoulos, M. Feed-forward neural networks. *Ieee Potentials* **13**, 27–31 (1994).
36. Laruelle, G. G. *et al*. Global high-resolution monthly pco 2 climatology for the coastal ocean derived from neural network interpolation. *Biogeosciences* **14**, 4545–4561 (2017).
37. Kingma, D. P. & Ba, J. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014).
38. Ronneberger, O., Fischer, P. & Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, 234–241 (Springer, 2015).
39. Shi, X. *et al*. Convolutional lstm network: A machine learning approach for precipitation nowcasting. Advances in neural information processing systems **28** (2015).
40. Wang, Z., Zou, S., Sun, H. & Chen, Y. Forecast global ionospheric tec: Apply modified u-net on vista tec data set. *Space Weather* **21**, e2023SW003494 (2023).
41. Chen, J., Gildin, E. & Killough, J. E. Transfer learning-based physics-informed convolutional neural network for simulating flow in porous media with time-varying controls. arXiv preprint arXiv:2310.06319 (2023).
42. Clevert, D.-A., Unterthiner, T. & Hochreiter, S. Fast and accurate deep network learning by exponential linear units (elus). arXiv preprint arXiv:1511.07289 (2015).
43. Kim, S., Nathaniel, J., Hou, Z., Zheng, T. & Gentine, P. Spatiotemporal Upscaling of Sparse Air-Sea pCO2 Data via Physics-Informed Transfer Learning, https://doi.org/10.5281/zenodo.12726686 (2024).
44. Abadi, M. *et al*. {TensorFlow}: a system for {Large - Scale} machine learning. In 12th USENIX symposium on operating systems design and implementation (OSDI 16), 265–283 (2016).
45. Pedregosa, F. *et al*. Scikit-learn: Machine learning in python. *the Journal of machine Learning research* **12**, 2825–2830 (2011).

Acknowledgements

The authors acknowledge funding from National Science Foundation (NSF) Science and Technology Center (STC) Learning the Earth with Artificial Intelligence and Physics (LEAP), Award #2019625-STC, DOE Advanced Scientific Computing Research (ASCR) program (DE-SC0022255), M2LInES research funding by the generosity of Eric and Wendy Schmidt by recommendation of the Schmidt Sciences program and European Research Council grant USMILE.

Author contributions

P.G. and T.Z. conceived the idea and experiments. S.K. and J.N. conducted the experiments and wrote the initial drafts. S.K., J.N., Z.H. analysed the results. All authors reviewed the manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to S.K. or J.N.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.