G. Model Architectures

In Figure 9, We show visualizations of architectures described in Sec.3.1.4.

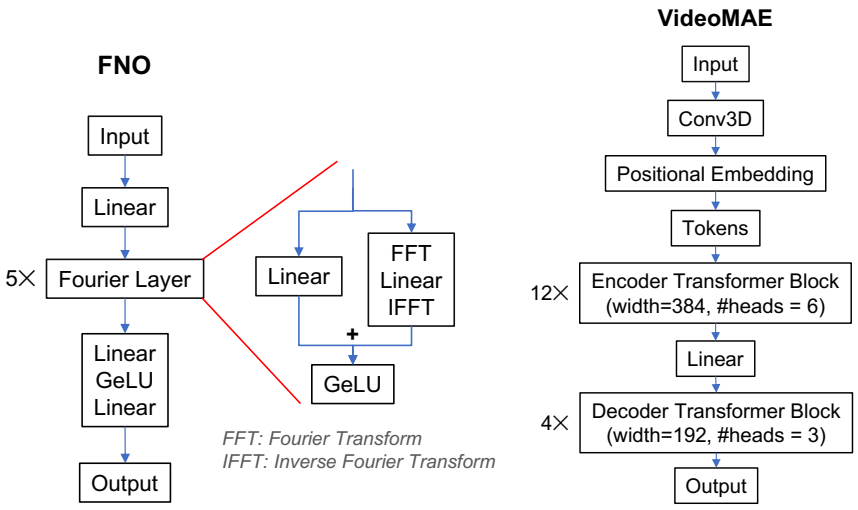

Figure 9. Visualizations of architectures we studied. Left: FNO (Li et al., 2021). Right: VideoMAE (Tong et al., 2022).

H. More Comprehensive Experiments on Real-World Data

In this section, we move to a broader range of benchmarks. We will study real-world and noisy data instead of toy datasets, providing even more comprehensive experiments.

Model and Training. We adopt the same VideoMAE architecture (Tong et al., 2022) used in Sec.4.1. See Figure 9 for details of the architecture. We use 15 consecutive temporal snapshots to forecast the next time step. We train VideoMAE with Adam and keep other hyperparameters the same as in Table 2 column "N.S. (PDEBench)".

H.1. ECMWF Reanalysis v5 (ERA5)

The ECMWF (European Centre for Medium-Range Weather Forecasts) offers ERA5, a publicly accessible, extensive dataset (Hersbach et al., 2020), which delivers hourly data on multiple atmospheric variables spanning from 1979 to today. ERA5 represents a type of atmospheric reanalysis dataset (Kalnay et al., 2018), integrating observations from a variety of measurement sources with numerical models through data assimilation (Kalnay, 2003). Essentially, it reconstructs the most accurate estimation of the Earth's atmospheric conditions over time. This dataset has been extensively utilized in prior scientific machine learning (SciML) studies (Pathak et al., 2022; Ren et al., 2023).

In this experiment, we focus on forecasting the important and challenging **temperature** atmospheric variable. We utilize data from 2006 to 2015, with the snapshots taken every 6 hours and a spatial resolution of 360×360. The total number of snapshots is 14600. We apply the mean-standard deviation normalization to the data and downsample the snapshots to a spatial resolution of 180×180. We split 75% of the data for pretrain and 25% for finetune. For each split, we further separate 80% of the data for training, 10% for validation, and 10% for testing.

Results. We show our results in Figure 10 (bottom left). Compared with directly training neural operators from scratch ("random init."), pretraining on unlabeled PDE data (2D snapshots without temporal dynamics) can help neural operators achieve better performance and reduce the amount of simulated data.

H.2. ScalarFlow

ScalarFlow (Eckert et al., 2019) is a dataset of reconstructions of real-world smoke plumes. It assembles the first large-scale dataset of realistic turbulent flows. The availability of a large, volumetric data set opens up a wide range of applications, including re-simulations, novel visualization, and metric evaluations. The dataset contains 104 real-world smoke flow

density reconstructions. Each reconstruction is captured from five different viewpoints for 150 temporal frames spanning 2.5 seconds. The original spatial resolution is 1062×600. We crop the snapshots to 900×600 to remove the padding and background. We remove the first 15 timeframes for each simulation to avoid the initial transient phase at the beginning of the smoke flow generation. We then downscale the snapshots to 180×120 and apply mean-standard deviation normalization to the data. With a total of 70200 snapshots, we split 80% of the data for pretrain and 20% for finetune. For each split, we further separate 80% of the data for training, 10% for validation, and 10% for testing.

Results. We show our results in Figure 10 (bottom right). Pretraining neural operators on unlabeled PDE data, specifically 2D snapshots that lack temporal dynamics, as opposed to training them from the beginning with no prior knowledge ("random init."), can enhance their performance. This approach also decreases the necessity for simulated data.

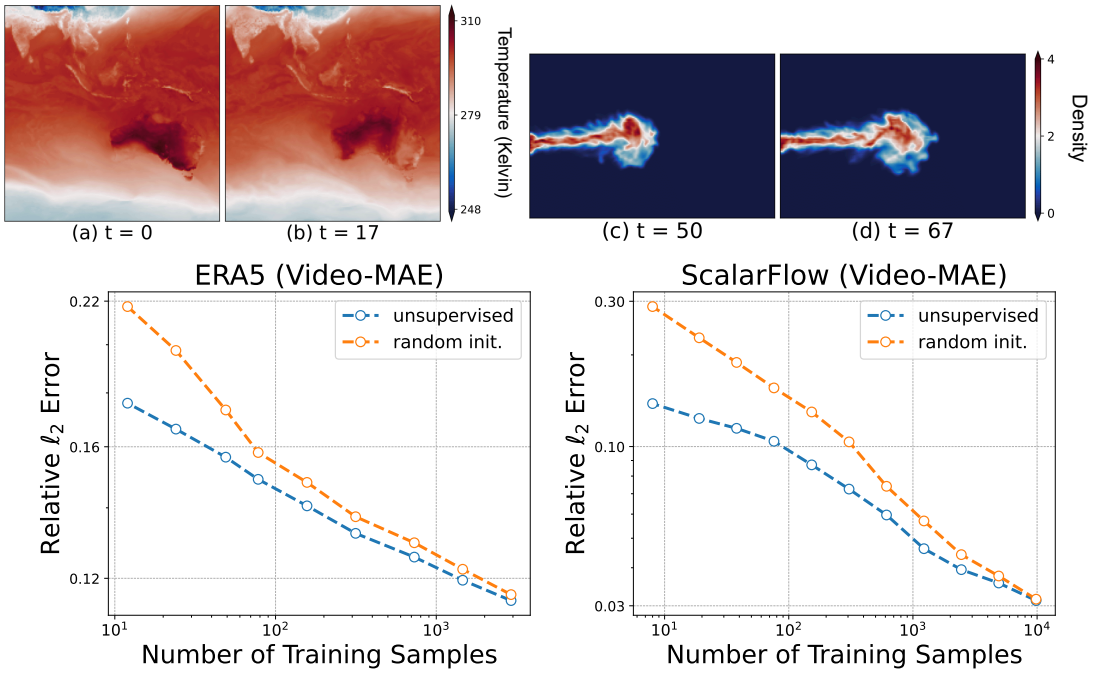

Figure 10. On **real-world scientific data**, pretraining neural operators on unlabeled PDE data improves its performance and data efficiency. **Top row**: We show snapshot examples from ERA5 temperature (**a**, **b**) and ScalarFlow (**c**, **d**) at different temporal steps. **Bottom row**: We pretrain VideoMAE (Tong et al., 2022) with unlabeled snapshots (no temporal dynamics), and then fine-tune across different numbers of temporal snapshots on ERA5 (bottom left) and ScalarFlow (bottom right). "random init.": models are trained from scratch with random initialization.

I. More Benefits of Unsupervised Pretraining

We provide further analysis on benefits of our unsupervised pretraining. Beyond reduced overfitting and meaningful representations shown in Fig. 4, our method can also accelerate the convergence of model's fine-tuning. As demonstrated in Fig. 11, compared with the random initialization, our method achieves lower training error with much fewer epochs or iterations.

J. Data Generation

J.1. Unlabeled PDE Data

We generate data for Poisson and Helmholtz (Subramanian et al., 2023), Reaction-Diffusion on PDE-Bench (Takamoto et al., 2022) and 2D incompressible Navier-Stokes on PINO Dataset (Li et al., 2021) following the procedure mentioned in the paper. We additionally organize the code in the repository. For unlabeled data generation, we discard the code for solvers, which expedite the generation speed, as shown in Table 8.

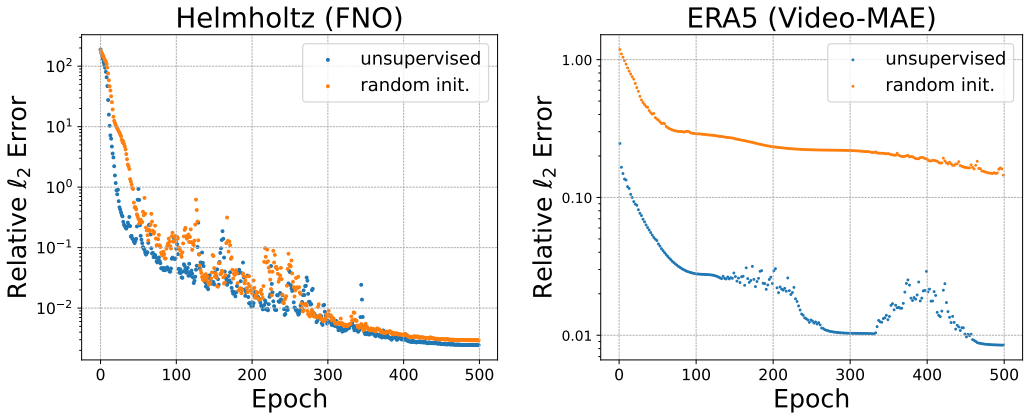

Figure 11. Unsupervised pretraining can accelerate model convergence and minimize training error faster than random initialization (beyond reduced overfitting and meaningful representations in Fig. 4). Left: FNO on Helmholtz. Right: VideoMAE on ERA5.

J.2. OOD Samples

The OOD data generation procedure is similar to the unlabeled data, except for the changes in the physical parameters coefficients. For Poisson and Helmholtz, we consider changing the range of diffusion eigenvalue and waver number respectively. For Navier-Stokes equation, we change the Reynolds number. For Reaction-Diffusion equation, we do not consider the OOD cases. We list the coefficients in Table 1.

References

Eckert, M.-L., Um, K., and Thuerey, N. Scalarflow: a large-scale volumetric data set of real-world scalar transport flows for computer animation and machine learning. *ACM Transactions on Graphics (TOG)*, 38(6):1–16, 2019.

Hersbach, H., Bell, B., Berrisford, P., Hirahara, S., Horányi, A., Muñoz-Sabater, J., Nicolas, J., Peubey, C., Radu, R., Schepers, D., et al. The era5 global reanalysis. *Quarterly Journal of the Royal Meteorological Society*, 146(730): 1999–2049, 2020.

Kalnay, E. *Atmospheric modeling, data assimilation and predictability*. Cambridge university press, 2003.

Kalnay, E., Kanamitsu, M., Kistler, R., Collins, W., Deaven, D., Gandin, L., Iredell, M., Saha, S., White, G., Woollen, J., et al. The ncep/ncar 40-year reanalysis project. In *Renewable energy*, pp. Vol1_146–Vol1_194. Routledge, 2018.

Li, Z., Zheng, H., Kovachki, N., Jin, D., Chen, H., Liu, B., Azizzadenesheli, K., and Anandkumar, A. Physics-informed neural operator for learning partial differential equations. *arXiv preprint arXiv:2111.03794*, 2021.

Pathak, J., Subramanian, S., Harrington, P., Raja, S., Chattopadhyay, A., Mardani, M., Kurth, T., Hall, D., Li, Z., Azizzadenesheli, K., et al. Fourcastnet: A global data-driven high-resolution weather model using adaptive fourier neural operators. *arXiv preprint arXiv:2202.11214*, 2022.

Ren, P., Erichson, N. B., Subramanian, S., San, O., Lukic, Z., and Mahoney, M. W. Superbench: A super-resolution benchmark dataset for scientific machine learning. *arXiv preprint arXiv:2306.14070*, 2023.

Subramanian, S., Harrington, P., Keutzer, K., Bhimji, W., Morozov, D., Mahoney, M. W., and Gholami, A. Towards foundation models for scientific machine learning: Characterizing scaling and transfer behavior. *arXiv preprint arXiv:2306.00258*, 2023.

Takamoto, M., Praditia, T., Leiteritz, R., MacKinlay, D., Alesiani, F., Pflüger, D., and Niepert, M. Pdebench: An extensive benchmark for scientific machine learning. *Advances in Neural Information Processing Systems*, 35:1596–1611, 2022.

Tong, Z., Song, Y., Wang, J., and Wang, L. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. *Advances in neural information processing systems*, 35:10078–10093, 2022.