Generated using the official AMS LaTeX template v6.1 two-column layout. This work has been submitted as an extended abstract for the AMS Annual Meeting 2024 Poster **900**.

ANALYZING AND EXPLORING TRAINING RECIPES FOR LARGE-SCALE TRANSFORMER-BASED WEATHER PREDICTION

Jared D. Willard,[a] Peter Harrington,[a] Shashank Subramanian,[a] Ankur Mahesh,[b] Travis A. O'Brien,[c] William D. Collins[b,d]

[a] *Lawrence Berkeley National Laboratory, National Energy Research Scientific Computing Center*
[b] *University of California, Berkeley, Department of Earth and Planetary Science*
[c] *Indiana University Bloomington, Department of Earth and Atmospheric Sciences*
[d] *Lawrence Berkeley National Laboratory, Earth & Environmental Sciences Area*

The rapid rise of deep learning (DL) in numerical weather prediction (NWP) has led to a proliferation of models which forecast atmospheric variables with comparable or superior skill than traditional physics-based NWP. However, among these leading DL models, there is a wide variance in both the training settings and architecture used. Further, the lack of thorough ablation studies makes it hard to discern which components are most critical to success. In this work, we show that it is possible to attain high forecast skill even with relatively off-the-shelf architectures , simple training procedures, and moderate compute budgets. Specifically, we train a minimally modified SwinV2 transformer on ERA5 data, and find that it attains superior forecast skill when compared against IFS. We present some ablations on key aspects of the training pipeline, exploring different loss functions, model sizes and depths, and multi-step fine-tuning to investigate their effect. We also examine the model performance with metrics beyond the typical ACC and RMSE, and investigate how the performance scales with model size.

1. INTRODUCTION

As the impacts of climate change continue to grow in severity, it is becoming more important than ever to forecast weather phenomena with high accuracy and fidelity. While operational forecasting has long been performed by numerical weather prediction (NWP) models like the Integrated Forecast System (IFS), the availability of high-quality reanalysis data (Hersbach et al. 2020) and the onset of advanced deep learning techniques has led to a proliferation of data-driven forecast models in recent years. The power of deep learning in weather prediction has progressed rapidly, producing models that compete with or outperform leading NWP systems in key forecast metrics (Lam et al. 2023; Bi et al. 2023) only 4 years after the earliest pioneering works (e.g., Dueben and Bauer 2018). This, combined with the fact that deep learning weather models offer unique capabilities that can augment the capabilities of NWP (Ben-Bouallegue et al. 2023), has generated substantial interest in how to design the most effective deep learning approaches for weather prediction.

Recent literature has proposed an abundance of training recipes, network architectures, inference configurations, and compute budgets for the task of global medium-range forecasting. Proposed architectures range from graph neural networks (Keisler 2022; Lam et al. 2023), transformers (Bi et al. 2023; Chen et al. 2023a,b; Nguyen et al. 2023), neural operators (Pathak et al. 2022; Bonev et al. 2023), and convolutional neural networks (Karlbauer et al. 2023); the training recipes include different loss functions, normalization methods, time-stepping schemes, and varied sets of atmospheric variables and resolutions/grids. Expanding beyond just deterministic forecasting and conventional deep learning, other works have also explored diffusion models (Price et al. 2023) and hybrid physics-ML schemes (Kochkov et al. 2023). This progress, while exciting, also presents a challenge for researchers, as the differences between models are multifold and sometimes entangled, so separating their effects is not always possible. While some works present ablations and analysis on parts of their models, it is challenging to do so comprehensively, and there remains a need for analysis under restricted and controlled settings.

With the advent of open benchmarks providing detailed and informative model evaluation (Rasp et al. 2023; Brenowitz et al. 2024), comparisons between and within models are becoming easier, and the recent Stormer model (Nguyen et al. 2023) exemplifies what can be done with extensive ablation studies. Their work takes a standard Vision Transformer (ViT) and explores the impact of key components of the training pipeline and model architecture on the downstream performance of the model, assessed by the

Corresponding author: Jared D. Willard, jared.d.willard@gmail.com

root-mean-square-error (RMSE) of deterministic forecasts for key weather variables. Their improvements achieve highly competitive accuracy when compared against other leading models at $\sim 1.4°$ resolution.

In this work, we take a similar approach and aim to assess the effect of different training settings, and the interplay between them, but *at full ERA5 resolution* (0.25°). The decision to work at full resolution is motivated by the fact that high-resolution forecasts are simply more useful, and that excessive blurring and lack of fine-scale detail is a current shortcoming of current deep-learning-based models (Ben-Bouallegue et al. 2023; Price et al. 2023; Brenowitz et al. 2024). We choose the SwinV2 (Liu et al. 2022) architecture as a representative transformer-based architecture that works well at high resolutions, and is a relatively "off-the-shelf" architecture widely available in deep learning libraries. With SwinV2, we explore several different aspects of the training pipeline. We do not aim to provide a comprehensive analysis of all proposed training strategies, but focus on what can achieve good deterministic skill at full ERA5 resolution on 1-5 day lead times. This focus is partly motivated by observations that multi-step fine-tuning and other methods that most improve skill at long lead times exacerbate the issue of blurring, as model predictions tend to resemble more of an ensemble mean rather than a deterministic forecast (Price et al. 2023; Brenowitz et al. 2024). Our contributions are as follows:

- We present a minimally modified SwinV2 model trained on ERA5 at full 0.25° resolution that outperforms IFS in deterministic skill.
- We present detailed ablations over key training and model settings, finding the effects of latitude-weighting, channel-weighting, and multi-step fine-tuning to be somewhat entangled; they are generally positive but not always constructive or additive with each other.
- We confirm that multi-step fine-tuning can improve RMSE but affect sharpness and ensemble spread in transformer architectures as well.

2. DATASET & MODEL DETAILS

a. Data

We use the ERA5 (Hersbach et al. 2020) dataset, provided by ECMWF (European Center for Medium-Range Weather Forecasting). ERA5 contains hourly reanalysis data at a spatial resolution of 0.25° ($\sim$ 25 km) from years 1979 to present day. For this study we subsample ERA5 on $\Delta t = 6$ hour time intervals, and select 73 variables from the full dataset to include in the model (this closely follows previous work, e.g. Nguyen et al. (2023); Bi et al. (2022)). These are geopotential height (z), winds (u, v), temperature (t), and specific humidity (q) at 13 vertical pressure levels (50hPa, 100hPa, 150hPa, 200hPa, 250hPa, 300hPa, 400hPa, 500hPa, 600hPa, 700hPa, 850hPa, 925hPa, and 1000hPa), along with 8 single-level/surface variables: surface winds at 10m and 100m (u10, v10, u100, v100), 2m temperature (t2m), surface pressure (sp), mean sea level pressure (msl), and total column water vapor (tcwv). We also include as static additional inputs the land-sea mask, orography, and cosine of zenith angle (indicating time of day/year). As mentioned in the previous section, we focus on results at full ERA5 resolution and thus do not downsample the data as in Nguyen et al. (2023). All data is normalized by the global mean and standard deviation per variable before training. We use years 1979-2015 as well as 2019 for training data, and 2016-17 for validation, then evaluate on 2018 in line with other recent DL-based forecast models.

b. Model Architecture

We base our model on the SwinV2 (Liu et al. 2022) implementation available in v0.9.2 of the `timm` library[1]. While other DL-based forecast models have made extensive modifications to Swin backbones (Bi et al. 2023; Chen et al. 2023b), we find it sufficient to just minimally modify two aspects of the SwinV2 architecture:

- **Window shifting:** The shifting of attention windows in Swin is implemented as a `torch.roll` operation, followed by masking to ensure attention within shifted windows doesn't cross image boundaries (since 2D images are non-periodic generally). In the case of ERA5, rolling along the horizontal (zonal) direction is fine, since this axis is periodic, so we only apply the masking along the vertical dimension. This slightly simplifies the SwinV2 code.
- **Position embedding:** In SwinV2 the position biases within attention windows are carefully defined in a relative coordinate system, to better allow transferring across multiple resolutions/window sizes. In our case we are restricting ourselves to data at fixed resolution, and thus find it sufficient to drop the relative position embeddings in favor of a standard "absolute" position embedding as in a standard ViT, which is added to the latent space immediately after patch embedding.
- **Non-hierarchical structure:** SwinV2 uses a common vision transformer technique to incorporate a hierarchy across the network layers that sequentially merges patches and decreases resolution in order to both reduce computational cost and model various scales. We alter the model to be non-hierarchical and maintain the same feature resolution in all layers, which has shown to be effective for spatiotemporal forecasting in earth science (Gao et al. 2022).

[1] https://github.com/huggingface/pytorch-image-models

With these modifications we train our forecast models in the standard setting, giving the state $\mathbf{X}_t$ as input and predicting the next state $\mathbf{X}_{t+\Delta t}$. In inference, the model is rolled out autoregressively to produce forecasts for lead times larger than Δt. For our baseline SwinV2 model we use an embedding dimension of 768, depth of 12 layers, patch size of 4, 8 attention heads, and local attention window size 9x18. The model is trained with latitude-weighted (described in the following section) $\mathcal{L}_2$ loss using the Adam optimizer with a DropPath rate of 0.1, learning rate of 0.001, and batch size of 64, and trains for 70 epochs on 64 A100 GPUs. Activation checkpointing is used as necessary to fit fine-tuning configurations into GPU memory. Modifications and ablations with respect to this baseline model are described in subsequent sections.

c. Ablations & Experiments

Model size: In preliminary experimentation we found that, for a given embedding dimension and depth, the largest possible window size and smallest patch sizes performed best. Thus to probe the effect of larger model sizes, we explore growing both the depth and width dimensions. We evaluate the performance of a model with twice as many layers (`depth=24`), as well as double the embedding dimension (`embed_dim=1536`).

Channel weighting: Recent work has found it beneficial to carefully weight channels (different weather variables at different vertical levels) in the loss function during training (Nguyen et al. 2023). In particular, the method first pioneered by Lam et al. (2023) of down-weighting as pressure level decreases (higher vertical levels are weighted less) and down-weighting according to the standard deviation of temporal differences ($\sigma_{\delta\mathbf{X}}$) has been found to work well. Beyond these physically-motivated weights, Lam et al. (2023) also manually impose weights that preferentially emphasize certain surface variables, like t2m, which we adopt as well for consistency. Similar to Nguyen et al. (2023) we evaluate the effect of this configuration compared against the standard loss where all channels are given equal weight.

We note this experiment is also entangled with "direct" vs. "residual" prediction – the channel-weighting method, which partially weights according to the temporal differences ($\sigma_{\delta\mathbf{X}}$), predicts the difference between the input and target timesteps, whereas the standard $\mathcal{L}_2$ loss directly predicts the target timestep. In this work we do not explore disentangling these two configurations, but in principle one could separately apply pressure-level/$\sigma_{\delta\mathbf{X}}$ and direct/residual prediction.

Multi-step fine-tuning: Implemented in many works, (Pathak et al. 2022; Lam et al. 2023; Bonev et al. 2023; Nguyen et al. 2023; Chen et al. 2023b), this method aims to improve long-term forecast performance by optimizing the loss over multiple (autoregressive) timesteps. While this improves deterministic RMSE at longer lead times, it has also been found to adversely affect forecast sharpness and ensemble spread (Price et al. 2023; Brenowitz et al. 2024). We evaluate the effects of fine-tuning models with 4 and 8 timesteps to build upon the observations found in previous works. Fine-tuning is done at a reduced learning rate of 1e-4 for 15 epochs.

Latitude-weighting: A large number of works, dating back to the original WeatherBench baseline Rasp et al. (2020), have additionally weighted the loss according to the cosine of each grid cell's latitude. This is motivated by the spherical geometry, and compensates for the difference in area between cells near the poles versus the equator in the equiangular projection. We examine the performance of models with and without this weighting applied, though it is applied by default if unspecified.

d. Evaluation

We use the recent open-source `earth2mip` package[2] (Brenowitz et al. 2024) to score and evaluate models. Scores are averaged over 11 initial conditions evenly spaced throughout 2018, and forecasts are rolled out to 7 days at 6 hour intervals. We primarily focus on latitude-weighted deterministic RMSE to compare between different models, but doing so imposes trade-offs with other metrics and desirable aspects of forecast quality (e.g. sharpness/bluriness). Thus we additionally measure energy spectra and the ensemble spread/skill in a lagged-ensemble (Tracton and Kalnay 1993) forecast for some of our models, to further illuminate these trade-offs.

3. RESULTS

a. Model size, channel weighting, & multi-step fine-tuning

We examine variations of model size and the effect on forecast skill in Figure 1. While doubling the embedding dimension or depth improves performance generally, the model variant with increased embedding dimension consistently outperforms the deeper model at all lead times. The deeper variant struggles to beat the baseline in t2m and u10, and the gap between it and the model with larger embedding dimension becomes as large as 10-15% in RMSE at 7-day lead times.

In Figure 2 we compare the baseline configuration against a model trained with channel-weighted loss and confirm that the channel-weighting seems to improve the model's forecasting accuracy across the majority of lead times. For some variables (e.g., u10) the improvement is more apparent at later lead times. We note the improvement is present even in variables which are actually down-weighted by the loss (e.g., z500, which is weighted less since it is further from the surface).

[2] https://github.com/NVIDIA/earth2mip

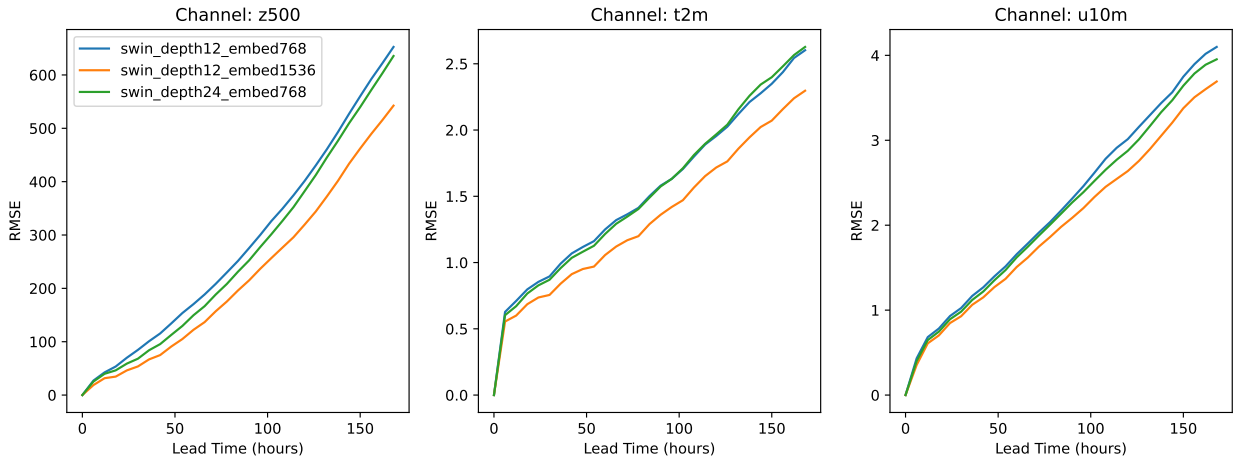

FIG. 1. Caption

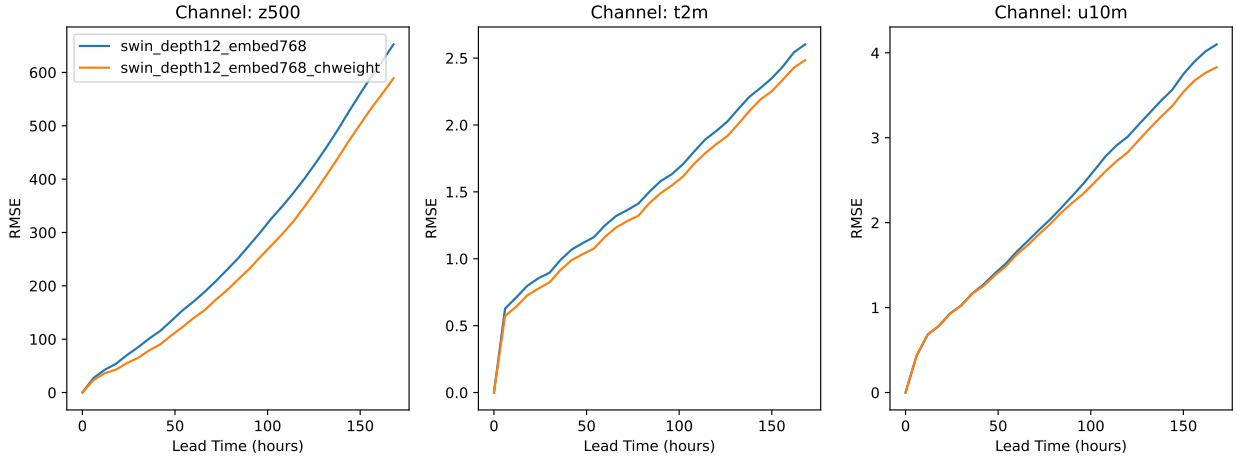

FIG. 2. RMSE comparison of forecasts at lead times up to 7 days for the baseline depth 12 and embedding dimension 768 swin model with and without custom channel-weighting

In Figure 3 we show the effect of multi-step fine-tuning and channel-weighting simultaneously applied, finding it effective in improving RMSE as previous works have shown. The overall effect seems to be that multi-step fine-tuning improves performance most at lead times much larger than the fine-tuning window. For example, up to lead times of ~2-3 days, there is no benefit from fine-tuning up to 8 steps (48 hours) versus just 4 (24 hours) – both improve over the baseline by the same amount. However, at lead times of 5 days and beyond, the gap between the two fine-tuning configurations has increased substantially and the 8-step model performs much better.

b. Downstream effects of multi-step fine-tuning

Given the stark improvement in RMSE offered by multi-step training, it is worth reiterating that these improvements come at the cost of other qualities desirable in weather models: forecast sharpness and ensemble spread. As a quick demonstration, we show in Figure 4 the power spectra of model predictions compared against the ground truth ERA5 (averaged over all lead times and initial conditions). The multi-step fine-tuned models are clearly deficient in higher wavenumbers for u10, indicating blurring. This effect is not present in all variables, as seen in the spectra for, e.g. surface temperature, whose fine-scale power might be more dominated by (static) orographic features like mountain ranges and coastlines rather than dynamics. The spikes and pileup near Nyquist frequency are caused by the patch embedding in SwinV2.

We examine spread and performance of an ensemble constructed from lagged forecasts in Figure 5. The lagged ensemble procedure and motivation is described in detail by Brenowitz et al. (2024), but generally we can expect

that models which are more intrinsically more dispersive (i.e., create ensembles with larger spread) to have a larger spread/skill ratio, which should ideally be 1 for an optimal real-world ensemble. In Figure 5 we observe that indeed the spread-skill ratio is diminished for both multi-step fine-tuned models, confirming the issues presented in Brenowitz et al. (2024). This decrease in spread-skill ratio is not catastrophic, as there still appears to be ensemble skill gains from fine-tuning as shown in ensemble mean RMSE and CRPS in the first and third rows respectively. In particular, the 8-step fine-tuned model has better skill at all lead times than the baseline for all three variables in both metrics.

c. *Effects of latitude-weighted loss*

Though the previous results have all incorporated the latitude-weighted loss, in Table 1 we examine this choice more closely and compare across several model configurations with and without applying latitude-weighting. The results are mixed, highlighting the need for caution when drawing conclusions from just one or two experiments alone. For the 8-step fine-tuned model, the latitude-weighted loss is generally helpful and achieves the lowest RMSE at lead times of 2 and 4 days (at 7-days the RMSE is roughly the same with and without latitude-weighting), with the best overall configuration using both channel-weighting and latitude-weighting in the loss. Surprisingly, the conclusion is reversed for a model that is only trained over single-step predictions – the latitude-weighted loss does significantly poorer at all lead times, and the best configuration uses neither channel-weighting nor latitude-weighting. Thus the effects of channel-weighting, latitude-weighting, and multi-step training are entangled; since it is common practice to do more hyperparameter tuning using less expensive configurations first (e.g., tune 1-step training first and then apply multi-step training) this can cause problems during the model development process.

Table 1. RMSE for Z500 (m) at 2,4,and 7 day forecast lead times for different model configurations trained with and without latitude-weighted loss. All models have depth 12 and 768 embedding dimension. The best results for 1-step and 8-step training are highlighted in bold.

Channel weighting	# step training	Latitude-weighted	Z500 (2day)	Z500 (4day)	Z500 (7day)
–	1	–	**95.74**	**237.36**	**549.18**
–	1	✓	134.29	299.96	652.52
✓	1	–	102.31	261.08	572.33
✓	1	✓	106.94	253.72	589.08
–	8	–	85.48	218.65	**503.28**
–	8	✓	84.01	217.19	507.65
✓	8	–	88.48	227.77	505.18
✓	8	✓	**83.20**	**216.66**	503.93

d. *Additional experiments*

Beyond this main set of experiments, we also partially explore other methods proposed in the literature, but do not run complete ablation suites due to the initially poor performance we observe. In particular, we attempt to use the variable tokenization embedding (Nguyen et al. 2023), which uses a cross-attention operation to fuse information between variables after patch embedding. This had previously only been demonstrated on much coarser resolution, and we find that applying it to full 0.25° resolution data poses significant computational challenges due to the memory cost. With activation checkpointing and breaking the operation to run in chunks sequentially, we are able to get it to fit on 80GB A100s, but we observe a slight degradation in the training and validation loss, contrary to the observations of (Nguyen et al. 2023). Since the checkpointing and chunked computation significantly increases the runtime, we do not pursue this further.

We also explore the alternate negative log likelihood (NLL) loss from (Chen et al. 2023a), which down-weights the loss according to a network-predicted uncertainty. Initial experiments under this configuration showed the 1-step RMSE on the validation set lagging behind the baseline configuration, and the predictions were noticeably more blurry. Thus we did not find this configuration worth including in more extensive ablations.

e. *Comparison with IFS*

Three of the top performing models from this work are compared with the IFS deterministic forecasts for 2018 in Figure 6. The IFS forecasts are done using 668 initial times which include the 11 initial times from our analysis. We see both the 8-step fine-tuned models, one with channel-weighting and latitude-weighted loss and one without, outperform the deterministic IFS forecast at all displayed lead times in terms of RMSE for z500, t2m, and u10m. The higher embedding dimension model trained only on single-step prediction only outperformed IFS on most lead times (at shorter and longer ends of the range) for z500, on all but the lead times over 6 days for t2m, and all lead times for u10m.

4. Conclusions

In this work we have demonstrated that relatively off-the-shelf architectures can outperform IFS and achieve highly competitive forecast skill with the proper training procedure. We find that increasing model size, applying channel weighting in the loss, and training over multiple time steps all improve deterministic forecast skill. We also see the effectiveness of the latitude-weighted loss to vary across different configurations, finding it more effective when used in conjunction with multi-step training. We confirm that multi-step training can also adversely affect

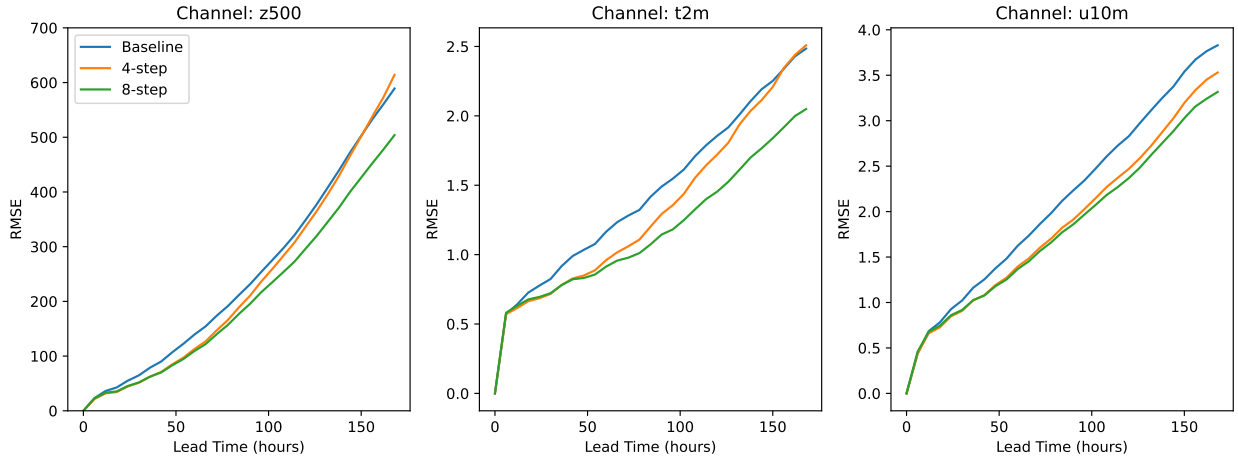

FIG. 3. RMSE comparison of forecasts at lead times up to 7 days for the model trained with custom channel-weighting, depth 12, and embedding dimension 768 alongside its variants with 4 and 8-step fine tuning

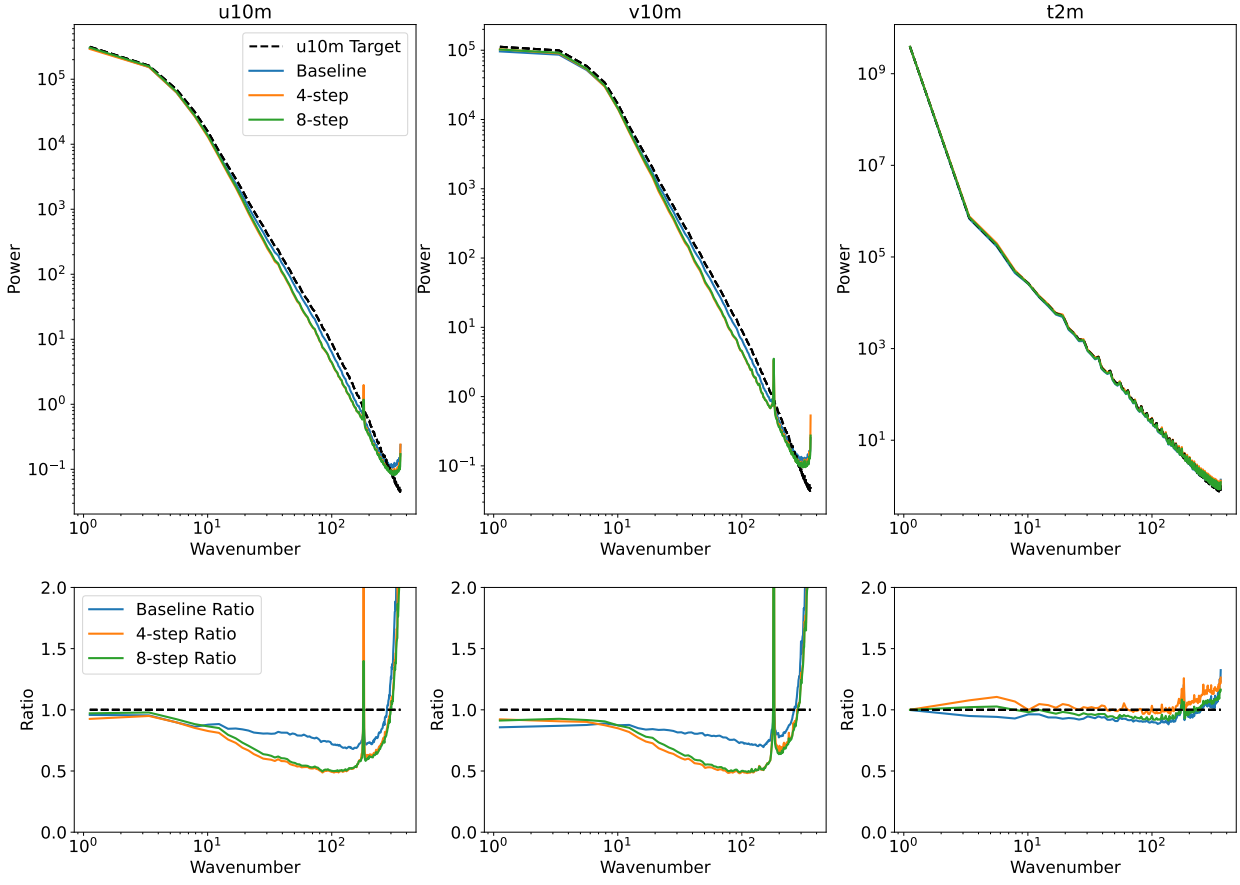

FIG. 4. Spatial frequency representation for the z500, t2m, and u10m variables across three model configurations: the baseline model using channel-weighting, and two fine-tuned versions of that model using 4-step and 8-step fine-tuning respectively. For each variable and model, the upper plot shows the power spectral density (PS1D) of both the target ERA5 (black line) and prediction (red line) on a logarithmic scale. The lower plot displays the ratio of predicted to actual PS1D values, with a ratio of 1 (dashed line) indicating perfect alignment. Ratios above or below 1 indicate overestimations or underestimations of power at specific spatial frequencies, respectively.

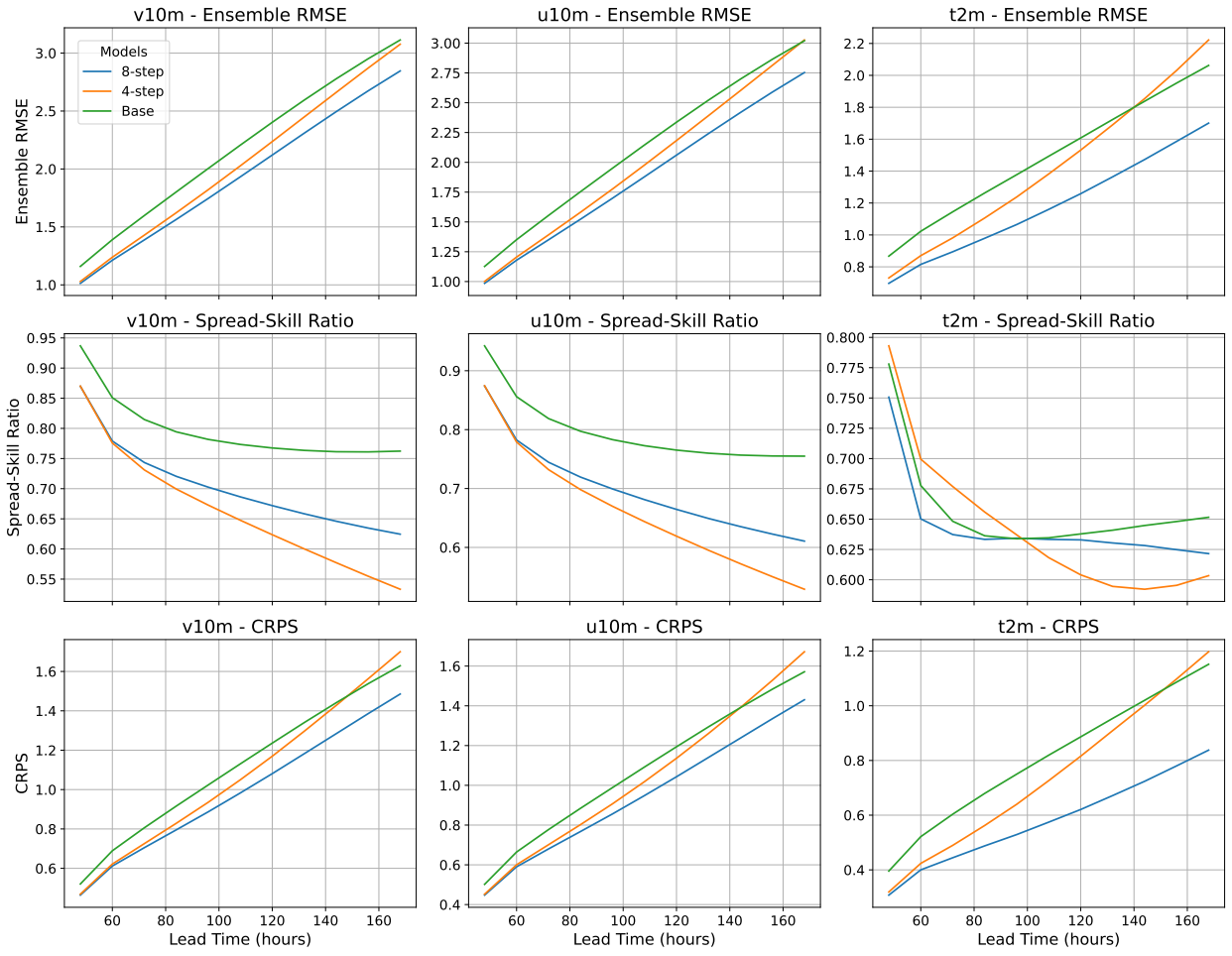

FIG. 5. Lagged ensemble forecast performance for v10, u10, and t2m across three model configurations: the baseline model using channel-weighting, and two fine-tuned versions of that model using 4-step and 8-step fine-tuning. The first row is ensemble mean RMSE calculated over the 9 member ensemble, the second row is the spread skill calculated by dividing the standard deviations of predictions across the ensemble by the ensemble RMSE, and the third row is continuous ranked probability score calculated across the ensemble.

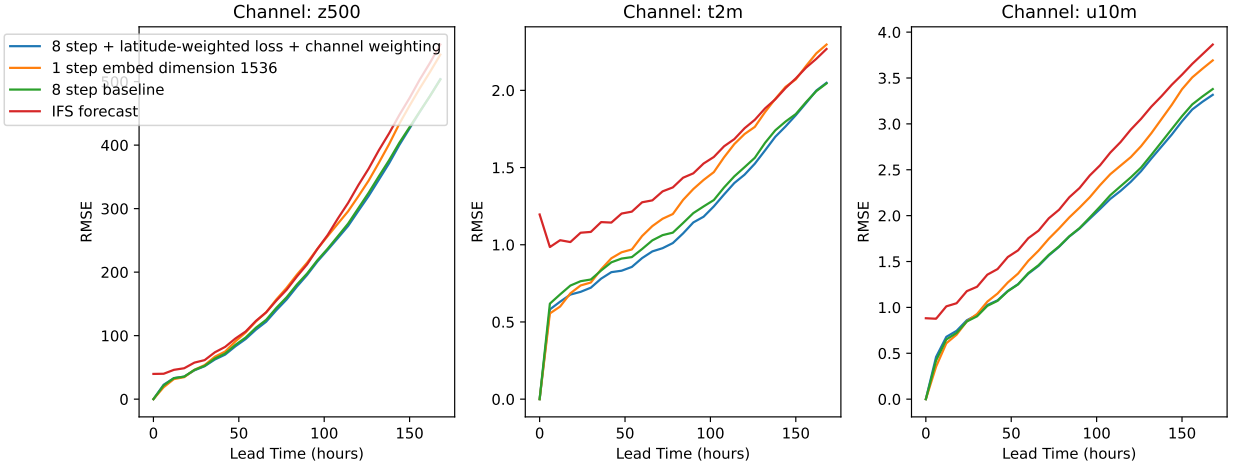

FIG. 6. RMSE comparison of forecasts of z500, t2m, and u10m at lead times up to 7 days for 3 of the previously shown models along with the deterministic forecast from IFS

forecast sharpness and ensemble spread, highlighting the need for other methods to stabilize rollouts and improve deterministic skill. We find some other innovations proposed in previous literature to either be ineffective in our setting or infeasible at full ERA5 resolution. Since our models are trained with moderate compute budgets (e.g., a total of 48 hours on 64 A100 GPUs for pre-training and fine-tuning the base model), we hope these models and findings will be of great practical use to the community.

Acknowledgments. This research was supported by the Director, Office of Science, Office of Biological and Environmental Research of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231 and by the Regional and Global Model Analysis Program area within the Earth and Environmental Systems Modeling Program. The research used resources of the National Energy Research Scientific Computing Center (NERSC), also supported by the Office of Science of the U.S. Department of Energy, under Contract No. DE-AC02-05CH11231. The computation for this paper was supported in part by the DOE Advanced Scientific Computing Research (ASCR) Leadership Computing Challenge (ALCC) 2023-2024 award 'Huge Ensembles of Weather Extremes using the Fourier Forecasting Neural Network' to William Collins (LBNL).

Data availability statement.

References

Ben-Bouallegue, Z., and Coauthors, 2023: The rise of data-driven weather forecasting. *arXiv preprint arXiv:2307.10128*.

Bi, K., L. Xie, H. Zhang, X. Chen, X. Gu, and Q. Tian, 2022: Pangu-weather: A 3d high-resolution model for fast and accurate global weather forecast. *arXiv preprint arXiv:2211.02556*.

Bi, K., L. Xie, H. Zhang, X. Chen, X. Gu, and Q. Tian, 2023: Accurate medium-range global weather forecasting with 3d neural networks. *Nature*, **619 (7970)**, 533–538.

Bonev, B., T. Kurth, C. Hundt, J. Pathak, M. Baust, K. Kashinath, and A. Anandkumar, 2023: Spherical fourier neural operators: Learning stable dynamics on the sphere. *arXiv preprint arXiv:2306.03838*.

Brenowitz, N. D., and Coauthors, 2024: A practical probabilistic benchmark for ai weather models. *arXiv preprint arXiv:2401.15305*.

Chen, K., and Coauthors, 2023a: Fengwu: Pushing the skillful global medium-range weather forecast beyond 10 days lead. *arXiv preprint arXiv:2304.02948*.

Chen, L., X. Zhong, F. Zhang, Y. Cheng, Y. Xu, Y. Qi, and H. Li, 2023b: Fuxi: A cascade machine learning forecasting system for 15-day global weather forecast. *arXiv preprint arXiv:2306.12873*.

Dueben, P. D., and P. Bauer, 2018: Challenges and design choices for global weather and climate models based on machine learning. *Geoscientific Model Development*, **11 (10)**, 3999–4009, https://doi.org/10.5194/gmd-11-3999-2018, URL https://gmd.copernicus.org/articles/11/3999/2018/.

Gao, Z., X. Shi, H. Wang, Y. Zhu, Y. B. Wang, M. Li, and D.-Y. Yeung, 2022: Earthformer: Exploring space-time transformers for earth system forecasting. *Advances in Neural Information Processing Systems*, **35**, 25 390–25 403.

Hersbach, H., and Coauthors, 2020: The era5 global reanalysis. *Quarterly Journal of the Royal Meteorological Society*, **146 (730)**, 1999–2049.

Karlbauer, M., N. Cresswell-Clay, R. A. Moreno, D. R. Durran, T. Kurth, and M. V. Butz, 2023: Advancing parsimonious deep learning weather prediction using the healpix mesh. 2311.06253.

Keisler, R., 2022: Forecasting global weather with graph neural networks. *arXiv preprint arXiv:2202.07575*.

Kochkov, D., and Coauthors, 2023: Neural general circulation models. *arXiv preprint arXiv:2311.07222*.

Lam, R., and Coauthors, 2023: Learning skillful medium-range global weather forecasting. *Science*, **382 (6677)**, 1416–1421.

Liu, Z., and Coauthors, 2022: Swin transformer v2: Scaling up capacity and resolution. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 12 009–12 019.

Nguyen, T., and Coauthors, 2023: Scaling transformer neural networks for skillful and reliable medium-range weather forecasting. *arXiv preprint arXiv:2312.03876*.

Pathak, J., and Coauthors, 2022: Fourcastnet: A global data-driven high-resolution weather model using adaptive fourier neural operators. *arXiv preprint arXiv:2202.11214*.

Price, I., and Coauthors, 2023: Gencast: Diffusion-based ensemble forecasting for medium-range weather. *arXiv preprint arXiv:2312.15796*.

Rasp, S., P. D. Dueben, S. Scher, J. A. Weyn, S. Mouatadid, and N. Thuerey, 2020: Weatherbench: a benchmark data set for data-driven weather forecasting. *Journal of Advances in Modeling Earth Systems*, **12 (11)**, e2020MS002 203.

Rasp, S., and Coauthors, 2023: Weatherbench 2: A benchmark for the next generation of data-driven global weather models. *arXiv preprint arXiv:2308.15560*.

Tracton, M. S., and E. Kalnay, 1993: Operational ensemble prediction at the national meteorological center: Practical aspects. *Weather and Forecasting*, **8 (3)**, 379–398.