ClimaX: A Foundation Model for Weather and Climate

Tung Nguyen [1] **Johannes Brandstetter** [2] **Ashish Kapoor** [3] **Jayesh K. Gupta** [*2] **Aditya Grover** [*1]

Abstract

Recent data-driven approaches based on machine learning aim to directly solve a downstream forecasting or projection task by learning a data-driven functional mapping using deep neural networks. However, these networks are trained using curated and homogeneous climate datasets for specific spatiotemporal tasks, and thus lack the generality of currently used physics-informed numerical models for weather and climate modeling. We develop and demonstrate ClimaX, a flexible and generalizable deep learning model for weather and climate science that can be trained using heterogeneous datasets spanning different variables, spatiotemporal coverage, and physical groundings. ClimaX extends the Transformer architecture with novel encoding and aggregation blocks that allow effective use of available compute and data while maintaining general utility. ClimaX is pretrained with a self-supervised learning objective on climate datasets derived from CMIP6. The pretrained ClimaX can then be finetuned to address a breadth of climate and weather tasks, including those that involve atmospheric variables and spatiotemporal scales unseen during pretraining. Compared to existing data-driven baselines, we show that this generality in ClimaX results in superior performance on benchmarks for weather forecasting and climate projections. Our source code is available at https://github.com/microsoft/ClimaX.

1. Introduction

Modeling weather and climate is an omnipresent challenge for science and society. Currently, numerical methods for global modeling of weather and climate are parameterized via various general circulation models (GCM) (Lynch, 2008). GCMs represent system of differential equations relating the flow of energy and matter in the atmosphere, land, and ocean that can be integrated over time to obtain forecasts for relevant atmospheric variables (Lynch, 2008; Bauer et al., 2015). While extremely useful in practice, GCMs also suffer from many challenges, such as accurately representing physical processes and initial conditions at fine resolutions, as well as technological challenges in large-scale data assimilation and computational simulations (Bauer et al., 2020). These factors limit their use in many scenarios, especially in simulating atmospheric variables quickly at very short time scales (e.g., a few hours) or accurately at long time scales (e.g., beyond 5-7 days) (Zhang et al., 2019).

In contrast, there has been a steady rise in data-driven approaches for forecasting of atmospheric variables, especially for meteorological applications (Grover et al., 2015; Dueben & Bauer, 2018; Weber et al., 2020; Scher & Messori, 2019; Scher, 2018; Kashinath et al., 2021; Schultz et al., 2021; Reichstein et al., 2019; Huntingford et al., 2019; Schneider et al., 2017). The key idea here is to train deep neural networks to predict the target atmospheric variables using decades of historical global datasets, such as the ERA-5 reanalysis dataset (Hersbach et al., 2020). With growing compute and data size, these models can achieve accuracies competitive with state-of-the-art numerical models in many scenarios, such as nowcasting of precipitation (Ravuri et al., 2021; Sønderby et al., 2020) and medium-range forecasting of variables like temperature, wind, and humidity (Weyn et al., 2020; Rasp & Thuerey, 2021; Keisler, 2022; Pathak et al., 2022; Bi et al., 2022; Lam et al., 2022). However, unlike GCMs, these networks are not explicitly grounded in physics, and lack general-purpose utility for Earth system sciences as they are trained for a specific task.

Variants of the aforementioned challenges apply broadly throughout machine learning (ML). In disciplines such as natural language processing and computer vision, it is well acknowledged that ML models trained to solve a single task using supervised learning are label-hungry during training and brittle when deployed outside their training distribution (Taori et al., 2020). Recent works have shown that it is possible to mitigate the supervision bottleneck by *pretraining* (Devlin et al., 2018; He et al., 2022) large unsupervised

[*]Equal contribution [1]UCLA [2]Microsoft [3]Scaled Foundations. Correspondence to: Tung Nguyen <tungnd@cs.ucla.edu>, Jayesh K. Gupta <jkg@cs.stanford.edu>, Aditya Grover <adityag@cs.ucla.edu>.

Accepted after peer-review at the 1st workshop on Synergy of Scientific and Machine Learning Modeling, SynS & ML ICML, Honolulu, Hawaii, USA. July, 2023. Copyright 2023 by the author(s).

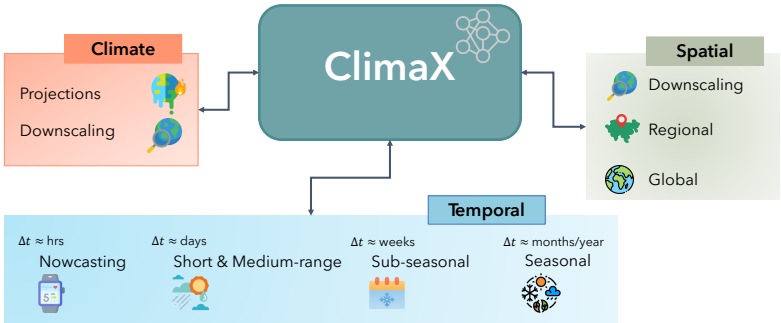

Figure 1. ClimaX is a foundation model for any weather and climate modeling task. On the weather front, these tasks include standard forecasting for various lead times at various resolutions, both globally and regionally. On the climate front, making climate projections and downscaling from lower-resolution model outputs are standard tasks.

"foundation" models (Bommasani et al., 2021) on huge passive datasets, such as text and images scraped from the internet (Ramesh et al., 2022; Brown et al., 2020; Liu et al., 2021; Reed et al., 2022b). Post pretraining, there are many ways to *finetune* the same model on arbitrary target task(s) with little to none (i.e., zero-shot) additional supervision. Besides low target supervision, these models also generalize better to shifts outside their training distribution (Hendrycks et al., 2020; Zhang et al., 2022), improving their reliability.

Inspired by the above successes, this work aims to design and train a foundation model for weather and climate that can be efficiently adapted for many tasks concerning the Earth's atmosphere. We propose ClimaX, a foundation model for weather and climate. Empirically, we show that a single pretrained model can be finetuned for many tasks (e.g., multi-scale weather forecasting, climate projections, downscaling) under a range of conditions involving different spatiotemporal resolutions, geographical regions, and target variables, including those unseen during training. Notably, our results are state-of-the-art on ClimateBench (Watson-Parris et al., 2022) and competitive with the operational Integrated Forecasting System (IFS) (Wedi et al., 2015) on WeatherBench (Rasp et al., 2020).

2. Approach

2.1. Input representation

We are interested in gridded prediction tasks, wherein the model takes an input of shape $V \times H \times W$ and predicts an output of shape $V' \times H' \times W'$. V is the number of input variables, which can be weather conditions such as geopotential and temperature, or climate forcing factors such as CO_2 and SO_2. H and W refer to the spatial resolution of the input data, which depends on how densely we grid the globe. Similarly, V', H', W' refer to the variables and spatial resolution of the predicted outputs.

2.2. Model architecture

We build ClimaX architecture upon Vision Transformers (ViT) (Dosovitskiy et al., 2020) because of its flexibility and scalability. In addition, we propose two major architectural changes, namely *variable tokenization* and *variable aggregation* to further improve the flexibility and generality.

Variable tokenization. Given an input of shape $V \times H \times W$, ViT tokenizes the input into a sequence of $(H/p) \times (W/p) = h \times w$ patches, with each patch having a size of $V \times p^2$, where p is the patch size. This tokenization scheme works well for image data, as V is always the RGB channels, which is the same for all datasets. However, this is not true for climate and weather data, where the number of physical variables can vary between different datasets. For example, in the CMIP6 project (Eyring et al., 2016), each dataset contains simulated data of a different climate model, and thus has a different set of underlying variables. Therefore, we propose *variable tokenization*, a novel tokenization scheme that tokenizes each variable in the input separately. Specifically, each input variable as a spatial map of shape $H \times W$ is tokenized into a sequence of $h \times w$ patches, which results in $V \times h \times w$ patches in total. Finally, each input patch of size p^2 is linearly embedded to a vector of dimension D, where D is the chosen embedding size. The output of the variable tokenization module therefore has a dimension of $V \times h \times w \times D$. Figure 2 illustrates our proposed tokenization scheme.

Variable aggregation. While variable tokenization allows ClimaX to learn from datasets with varying numbers of input variables, it has two inherent problems. First, it results in a sequence of length $V \times h \times w$ which increases linearly with the number of variables. Since we use attention to model the sequence, the memory complexity scales quadratically with the number of variables. This is computationally expensive, as we can have up to 48 input variables in our experiments. We therefore propose *variable aggregation* to solve the two mentioned challenges. For each spatial position in the $h \times w$

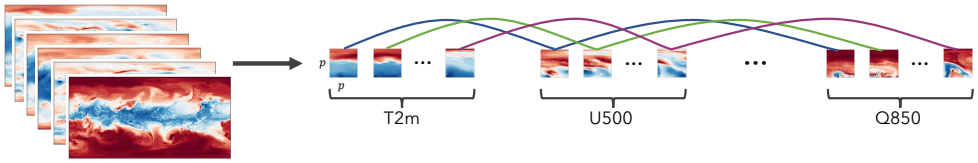

Figure 2. Variable tokenization tokenizes each variable independently, and position-based variable aggregation reduces a sequence of length $V \times h \times w$ to $h \times w$.

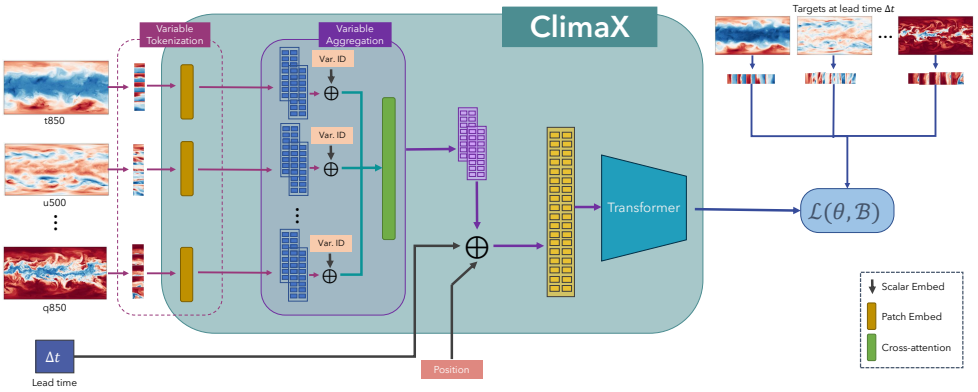

Figure 3. Pretraining phase of ClimaX. Variables are encoded using variable tokenization, and subsequently aggregated using variable aggregation. Together with position embedding and lead time embedding those are fed to the ViT backbone.

map, we perform a cross-attention operation, in which the query is a learnable vector, and the keys and values are the V embedding vectors of V variables at that position. The cross-attention module outputs a single vector for each spatial position, thus reducing the sequence length to $h \times w$, significantly lowering the computational cost.

Transformer. Post variable aggregation, we seek a sequence model for generating the output tokens. While in principle, one could use any sequence model, we propose to extend a standard Vision Transformer (ViT). Moreover, since the standard ViT treats image modeling as pure sequence-to-sequence problems, it can perform tasks that some other variations cannot (Liu et al., 2021; 2022), such as learning from spatially incomplete data, where the input does not necessarily form a complete grid. This is useful in the regional forecasting task we consider in Appendix C.1. In the experiments, we report results with 8 attention layers, an embedding size of 1024, and a hidden dimension of 1024×4. After the attention layers, we employ a 2-layer MLP prediction head that takes a token and outputs a vector of size $V' \times p^2$, for more details see Appendix D.

2.3. Datasets

Pretraining datasets. We use five datasets (MPI-ESM, TaiESM, AWI-ESM, HAMMOZ, CMCC) of the CMIP6 archive (Eyring et al., 2016). Appendix F.1 describes the pretraining data in details.

Finetuning and evaluation datasets. For various weather related downstream tasks, we use the ERA5 reanalysis data as described in Appendix F.2. For climate projections tasks, we evaluate Climax on ClimateBench (Watson-Parris et al., 2022), a recent climate projection benchmark.

2.4. Training

Pretraining. We pretrain ClimaX on CMIP6 data to predict future weather conditions given the current conditions. That is, given the weather snapshot X_t of shape $V \times H \times W$ at a particular time t, ClimaX learns to predict the future weather scenario $X_{t+\Delta t}$ of the same shape at lead time Δt. To obtain a pretrained model that is generally applicable to various temporal forecasting tasks, we randomize the lead time from 6 hours to 168 hours (i.e., 1 week) during pretraining. We add the lead time embedding to the tokens to inform the model of how long it is forecasting into the future. The lead time embedding module is a single-layer MLP that maps a scalar to an embedded vector. Figure 3 depicts the forward pass of ClimaX. For an input X_t, we sample a lead time $\Delta t \sim \mathcal{U}[6, 168]$ and get the corresponding ground truth $X_{t+\Delta t}$. Input variables are tokenized separately using variable tokenization, and are subsequently aggregated at each spatial location, resulting in a sequence of $h \times w$ unified tokens. We add the tokens with the lead time embedding and positional embedding before feeding the sequence to the ViT backbone. The output of the last attention layer is fed to a prediction head, which transforms the sequence

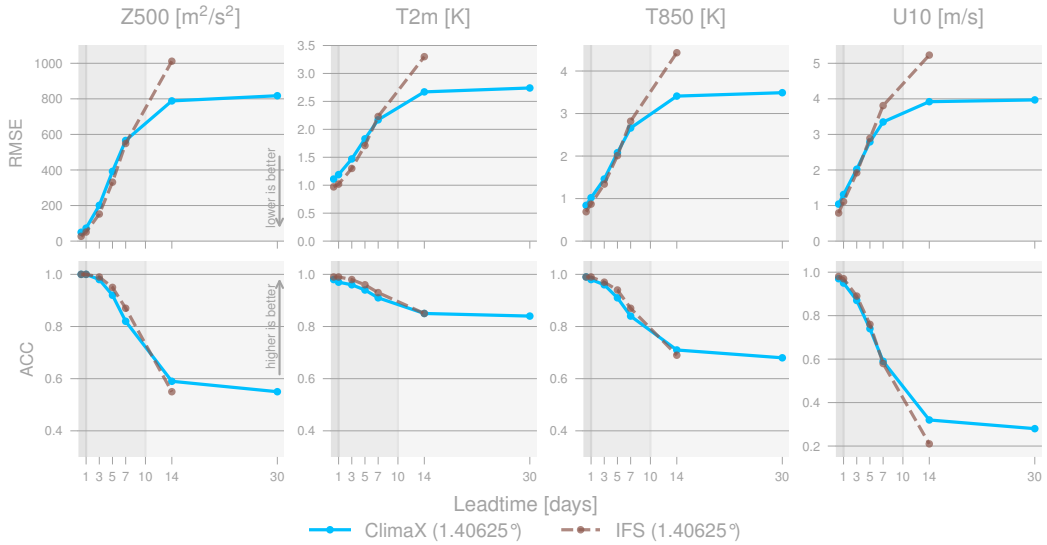

Figure 4. Global forecasting performance on ERA5 at $1.40625°$. Lower RMSE and higher ACC indicate better performance.

back to the original shape of $V \times H \times W$. We employ the latitude-weighted mean squared error (Rasp et al., 2020) as our objective function, detailed in Appendix E.1.1.

Finetuning. ClimaX has four learnable components: the token embedding layers, the variable aggregation module, the attention blocks, and the prediction head. We evaluate the performance of ClimaX on various downstream tasks, which we categorize into two finetuning scenarios: one in which the downstream variables belong to the set of pretraining variables, and the other with variables unseen during pretraining. In the first case, we finetune the entire model, and in the latter, we replace the embedding layers and the prediction head with newly initialized networks, and either finetune or freeze the other two components. We present more details of each downstream task in Section 3.

3. Experiments

We finetune ClimaX on a diverse set of downstream tasks to evaluate its performance and generality. Due to limited space, this section only presents global forecasting and climate projection results. We present applications of ClimaX to regional forecasting, sub-seasonal-to-seasonal forecasting, and climate downscaling, as well as the scaling laws analysis of ClimaX and ablation studies in Appendix C.

Global forecasting. Given global weather conditions X_t at a particular time t, we forecast the weather at a future time $X_{t+\Delta t}$, where Δt is the lead time. The input variables include 6 atmospheric variables at 7 vertical levels, 3 surface variables, and 3 constant fields, resulting in 48 input variables in total. We detail the variables in Table 9. We evaluate ClimaX on predicting four target variables: geopotential at 500hPa (Z500), the temperature at 850hPa (T850), the temperature at 2 meters from the ground (T2m), and zonal wind speed at 10 meters from the ground (U10). We consider seven lead times: 6 hours, $\{1, 3, 5, 7\}$ days, 2 weeks, and 1 month, spanning a range from nowcasting to short and medium-range forecasting. We consider predicting each target variable at each lead time a separate task, and finetune a separate model for each task.

We compare ClimaX to the Integrated Forecasting System (IFS), the current gold standard in weather forecasting. Following (Rasp et al., 2020), we split the data into three sets, in which the training data is from 1979 to 2015, the validation data is in 2016, and the test data is in 2017 and 2018. We finetune ClimaX using the latitude-weighted MSE loss in Equation (1). We perform early stopping on the validation loss for all deep learning models, and evaluate the best checkpoint on the test set. For IFS, we download the predictions from the TIGGE archive (Bougeault et al., 2010) for the year 2018. We compare two methods on latitude-weighted root mean squared error (RMSE) and latitude-weighted anomaly correlation coefficient (ACC), two commonly used metrics in previous works. The formulations of the two metrics are in Appendix G.1.

Figure 4 shows the performance of ClimaX and the baselines at $1.40625°$. The performance of ClimaX closely matches that of IFS even for short horizons, and is superior in forecasting at 7 days and beyond. The trends are similar for both RMSE and ACC. We include other additional task-specific baselines (Pathak et al., 2022; Bi et al., 2022; Lam et al., 2022) in Appendix G.2. These baselines are trained on higher-resolution ERA5 ($0.25°$) so are not directly comparable.

Table 1. Performance of ClimaX and the baselines on ClimateBench. Spatial and Global denote the normalized root mean squared error NRMSE_s and the NRMSE of the global mean NRMSE_g, respectively. Total is a weighted combination of Spatial and Global.

	Surface temperature				Diurnal temperature range				Precipitation				90th percentile precipitation			
	Spatial	Global	Total	RMSE	Spatial	Global	Total	RMSE	Spatial	Global	Total	RMSE	Spatial	Global	Total	RMSE
ClimateBench-NN (reproduced)	0.123	0.080	0.524	0.404	7.465	1.233	13.632	0.150	2.349	**0.151**	**3.104**	0.553	3.108	**0.282**	4.517	1.594
ClimateBench-NN (paper)	0.107	0.044	0.327	N/A	9.917	1.372	16.778	N/A	**2.128**	0.209	3.175	N/A	**2.610**	0.346	**4.339**	N/A
Cli-ViT	0.086	0.044	0.305	0.362	6.997	1.759	15.792	0.146	2.224	0.241	3.430	0.550	2.800	0.329	4.447	1.579
ClimaX	0.086	**0.043**	0.300	0.362	7.148	0.961	11.952	0.147	2.360	0.206	3.390	0.554	2.739	0.332	4.397	1.575
$\text{ClimaX}_{\text{frozen}}$	**0.085**	**0.043**	**0.297**	**0.360**	**6.688**	**0.810**	**10.739**	**0.144**	2.193	0.183	3.110	**0.549**	2.681	0.342	4.389	**1.572**

Climate projection. To further test the generality of ClimaX, we evaluate our model on ClimateBench (Watson-Parris et al., 2022), a recently introduced climate projection benchmark. The aim of ClimateBench is to predict the annual mean global distributions of surface temperature, diurnal temperature range, precipitation, and the 90th percentile of precipitation, given four anthropogenic forcing factors: carbon dioxide (CO_2), sulfur dioxide (SO_2), black carbon (BC), and methane (CH_4). This is not a temporal modeling task, as we do not predict the future given the past. Instead, we answer questions like *what will be the annual mean temperature for a specified CO_2 level?* In particular, note that input variables and the task itself are completely different from pretraining.

As the input and output variables are unseen during pretraining, we replace the pretrained embedding layers and prediction heads with newly initialized networks, while keeping the attention layers and the variable aggregation module. We consider two finetuning protocols, in which we either freeze[1] ($\text{ClimaX}_{\text{frozen}}$) or finetune (ClimaX) the attention layers. The details of the finetuning pipeline are in Appendix E.2.4. We compare ClimaX with $\text{ClimaX}_{\text{frozen}}$, Cli-ViT, and the best baseline from ClimateBench. Following (Watson-Parris et al., 2022), we use the standard mean squared error (Equation (1) without the weighting term) as the loss function. We evaluate all methods on RMSE, NRMSE_s (Spatial), NRMSE_g (Global), and Total = $\text{NRMSE}_s + 5 \times \text{NRMSE}_g$ (Watson-Parris et al., 2022), see Table 1. Details of the metrics are in Appendix G.1. $\text{ClimaX}_{\text{frozen}}$ and ClimaX perform best in predicting two temperature-related variables, demonstrating that pretrained attention layers can serve as a strong feature extractor in seemingly unrelated tasks. Since downstream data is scarce (ClimateBench has only 754 data points), further finetuning the attention layer can lead to overfitting and thus slightly hurt the performance. In two precipitation-related tasks, $\text{ClimaX}_{\text{frozen}}$ slightly underperforms ClimateBench baseline in terms of NRMSE_s and NRMSE_g, but outperforms on RMSE. We hypothesize that this was because ClimaX did not observe the precipitation variable during pretraining, which has very different behaviors from other variables.

[1]We finetune the LayerNorm in $\text{ClimaX}_{\text{frozen}}$, as suggested by Lu et al. (2022).

Broader impact

Large scale neural network surrogates for weather and climate modeling are destined to play an important role in the modeling of many related phenomena we experience on a daily basis. Therefore, developing even better models might enable us to achieve shortcuts or even better alternatives for computationally expensive simulations. Consequently, such models might help to advance various fields of weather and climate research. Examples related to this paper are nowcasting, predictions of extreme events, more reliable climate projections, and many more. On the plus side, new and better models might potentially be directly or indirectly related to reducing the carbon footprint, providing better anticipation of natural disasters, or better understanding of our earth. On the downside, relying purely on simulations always requires rigorous cross-checks and monitoring, especially when we "learn to simulate", or when we predict into the unknown as we do for climate projections.

References

Adopted, I. Climate change 2014 synthesis report. *IPCC: Geneva, Szwitzerland*, 2014.

Arcomano, T., Szunyogh, I., Pathak, J., Wikner, A., Hunt, B. R., and Ott, E. A machine learning-based global atmospheric forecast model. *Geophysical Research Letters*, 47(9):e2020GL087776, 2020.

Balaji, V., Couvreux, F., Deshayes, J., Gautrais, J., Hourdin, F., and Rio, C. Are general circulation models obsolete? *Proceedings of the National Academy of Sciences*, 119 (47):e2202075119, 2022.

Baño-Medina, J., Manzanas, R., and Gutiérrez, J. M. Configuration and intercomparison of deep learning neural models for statistical downscaling. *Geoscientific Model Development*, 13(4):2109–2124, 2020.

Bauer, P., Thorpe, A., and Brunet, G. The quiet revolution of numerical weather prediction. *Nature*, 525(7567):47–55, 2015.

Bauer, P., Quintino, T., Wedi, N., Bonanni, A., Chrust, M., Deconinck, W., Diamantakis, M., Düben, P., English, S., Flemming, J., et al. *The ecmwf scalability programme:*

Progress and plans. European Centre for Medium Range Weather Forecasts, 2020.

Beusch, L., Gudmundsson, L., and Seneviratne, S. I. Emulating earth system model temperatures with mesmer: from global mean temperature trajectories to grid-point-level realizations on land. *Earth System Dynamics*, 11(1): 139–159, 2020.

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

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., Brynjolfsson, E., Buch, S., Card, D., Castellon, R., Chatterji, N. S., Chen, A. S., Creel, K. A., Davis, J., Demszky, D., Donahue, C., Doumbouya, M., Durmus, E., Ermon, S., Etchemendy, J., Ethayarajh, K., Fei-Fei, L., Finn, C., Gale, T., Gillespie, L. E., Goel, K., Goodman, N. D., Grossman, S., Guha, N., Hashimoto, T., Henderson, P., Hewitt, J., Ho, D. E., Hong, J., Hsu, K., Huang, J., Icard, T. F., Jain, S., Jurafsky, D., Kalluri, P., Karamcheti, S., Keeling, G., Khani, F., Khattab, O., Koh, P. W., Krass, M. S., Krishna, R., Kuditipudi, R., Kumar, A., Ladhak, F., Lee, M., Lee, T., Leskovec, J., Levent, I., Li, X. L., Li, X., Ma, T., Malik, A., Manning, C. D., Mirchandani, S. P., Mitchell, E., Munyikwa, Z., Nair, S., Narayan, A., Narayanan, D., Newman, B., Nie, A., Niebles, J. C., Nilforoshan, H., Nyarko, J. F., Ogut, G., Orr, L., Papadimitriou, I., Park, J. S., Piech, C., Portelance, E., Potts, C., Raghunathan, A., Reich, R., Ren, H., Rong, F., Roohani, Y. H., Ruiz, C., Ryan, J., R'e, C., Sadigh, D., Sagawa, S., Santhanam, K., Shih, A., Srinivasan, K. P., Tamkin, A., Taori, R., Thomas, A. W., Tramèr, F., Wang, R. E., Wang, W., Wu, B., Wu, J., Wu, Y., Xie, S. M., Yasunaga, M., You, J., Zaharia, M. A., Zhang, M., Zhang, T., Zhang, X., Zhang, Y., Zheng, L., Zhou, K., and Liang, P. On the opportunities and risks of foundation models. *ArXiv*, 2021. URL https://crfm.stanford.edu/assets/report.pdf.

Bougeault, P., Toth, Z., Bishop, C., Brown, B., Burridge, D., Chen, D. H., Ebert, B., Fuentes, M., Hamill, T. M., Mylne, K., et al. The thorpex interactive grand global ensemble. *Bulletin of the American Meteorological Society*, 91(8): 1059–1072, 2010.

Brandstetter, J., Berg, R. v. d., Welling, M., and Gupta, J. K. Clifford neural layers for PDE modeling. *arXiv preprint arXiv:2209.04934*, 2022a.

Brandstetter, J., Worrall, D., and Welling, M. Message passing neural PDE solvers. *arXiv preprint arXiv:2202.03376*, 2022b.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901, 2020.

Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H. W., Sutton, C., Gehrmann, S., et al. PaLM: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*, 2022.

Cong, Y., Khanna, S., Meng, C., Liu, P., Rozi, E., He, Y., Burke, M., Lobell, D. B., and Ermon, S. Satmae: Pre-training transformers for temporal and multi-spectral satellite imagery. *arXiv preprint arXiv:2207.08051*, 2022.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Dueben, P. D. and Bauer, P. Challenges and design choices for global weather and climate models based on machine learning. *Geoscientific Model Development*, 11(10):3999–4009, 2018.

Ernst, L. Structured attention transformers on weather prediction. Master's thesis, ETH Zurich, Scalable Parallel Computing Laboratory, 2021.

Eyring, V., Bony, S., Meehl, G. A., Senior, C. A., Stevens, B., Stouffer, R. J., and Taylor, K. E. Overview of the coupled model intercomparison project phase 6 (cmip6) experimental design and organization. *Geoscientific Model Development*, 9(5):1937–1958, 2016.

Fan, H., Xiong, B., Mangalam, K., Li, Y., Yan, Z., Malik, J., and Feichtenhofer, C. Multiscale vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6824–6835, 2021.

Grover, A. Rethinking machine learning for climate science: A dataset perspective. In *AAAI Symposium on The Role of AI in Responding to Climate Challenges*, 2022.

Grover, A., Kapoor, A., and Horvitz, E. A deep hybrid model for weather forecasting. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 379–386, 2015.

Gupta, J. K. and Brandstetter, J. Towards multi-spatiotemporal-scale generalized pde modeling. *arXiv preprint arXiv:2209.15616*, 2022.

Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M. H., Brett, M., Haldane, A., del Río, J. F., Wiebe, M., Peterson, P., Gérard-Marchant, P., Sheppard, K., Reddy, T., Weckesser, W., Abbasi, H., Gohlke, C., and Oliphant, T. E. Array programming with NumPy. *Nature*, 585(7825):357–362, September 2020. doi: 10.1038/s41586-020-2649-2. URL https://doi.org/10.1038/s41586-020-2649-2.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

He, K., Chen, X., Xie, S., Li, Y., Dollár, P., and Girshick, R. Masked autoencoders are scalable vision learners. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16000–16009, 2022.

Hendricks, L. A., Mellor, J., Schneider, R., Alayrac, J.-B., and Nematzadeh, A. Decoupling the role of data, attention, and losses in multimodal transformers. *Transactions of the Association for Computational Linguistics*, 9:570–585, 2021.

Hendrycks, D., Liu, X., Wallace, E., Dziedzic, A., Krishnan, R., and Song, D. Pretrained transformers improve out-of-distribution robustness. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 2744–2751, 2020.

Henighan, T., Kaplan, J., Katz, M., Chen, M., Hesse, C., Jackson, J., Jun, H., Brown, T. B., Dhariwal, P., Gray, S., et al. Scaling laws for autoregressive generative modeling. *arXiv preprint arXiv:2010.14701*, 2020.

Hersbach, H., Bell, B., Berrisford, P., Biavati, G., Horányi, A., Muñoz Sabater, J., Nicolas, J., Peubey, C., Radu, R., Rozum, I., et al. Era5 hourly data on single levels from 1979 to present. *Copernicus Climate Change Service (C3S) Climate Data Store (CDS)*, 10, 2018.

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.

Hoffmann, J., Borgeaud, S., Mensch, A., Buchatskaya, E., Cai, T., Rutherford, E., Casas, D. d. L., Hendricks, L. A., Welbl, J., Clark, A., et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.

Höhlein, K., Kern, M., Hewson, T., and Westermann, R. A comparative study of convolutional neural network models for wind field downscaling. *Meteorological Applications*, 27(6):e1961, 2020.

Höök, M. and Tang, X. Depletion of fossil fuels and anthropogenic climate change—a review. *Energy policy*, 52:797–809, 2013.

Hoyer, S. and Hamman, J. xarray: N-d labeled arrays and datasets in python. *Journal of Open Research Software*, 5(1):10, April 2017. doi: 10.5334/jors.148.

Huang, G., Sun, Y., Liu, Z., Sedra, D., and Weinberger, K. Q. Deep networks with stochastic depth. In *European conference on computer vision*, pp. 646–661. Springer, 2016.

Huntingford, C., Jeffers, E. S., Bonsall, M. B., Christensen, H. M., Lees, T., and Yang, H. Machine learning and artificial intelligence to aid climate change research and preparedness. *Environmental Research Letters*, 14(12):124007, 2019.

Hurrell, J. W., Holland, M. M., Gent, P. R., Ghan, S., Kay, J. E., Kushner, P. J., Lamarque, J.-F., Large, W. G., Lawrence, D., Lindsay, K., et al. The community earth system model: a framework for collaborative research. *Bulletin of the American Meteorological Society*, 94(9):1339–1360, 2013.

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

Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J., and Amodei, D. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

Kashinath, K., Mustafa, M., Albert, A., Wu, J., Jiang, C., Esmaeilzadeh, S., Azizzadenesheli, K., Wang, R., Chattopadhyay, A., Singh, A., et al. Physics-informed machine learning: case studies for weather and climate modelling. *Philosophical Transactions of the Royal Society A*, 379(2194):20200093, 2021.

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

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Kochkov, D., Smith, J. A., Alieva, A., Wang, Q., Brenner, M. P., and Hoyer, S. Machine learning–accelerated computational fluid dynamics. *Proceedings of the National Academy of Sciences*, 118(21):e2101784118, 2021.

Lam, R., Sanchez-Gonzalez, A., Willson, M., Wirnsberger, P., Fortunato, M., Pritzel, A., Ravuri, S., Ewalds, T., Alet, F., Eaton-Rosen, Z., et al. Graphcast: Learning skillful medium-range global weather forecasting. *arXiv preprint arXiv:2212.12794*, 2022.

Law, K., Stuart, A., and Zygalakis, K. Data assimilation. *Cham, Switzerland: Springer*, 214:52, 2015.

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

Liu, Y., Ganguly, A. R., and Dy, J. Climate downscaling using ynet: A deep convolutional network with skip connections and fusion. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 3145–3153, 2020.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., and Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10012–10022, 2021.

Liu, Z., Hu, H., Lin, Y., Yao, Z., Xie, Z., Wei, Y., Ning, J., Cao, Y., Zhang, Z., Dong, L., Wei, F., and Guo, B. Swin transformer v2: Scaling up capacity and resolution. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Lorenz, E. The nature and theory of the general circulation of the atmosphere. *World meteorological organization*, 161, 1967.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Lu, K., Grover, A., Abbeel, P., and Mordatch, I. Pretrained transformers as universal computation engines. In *AAAI Conference on Artificial Intelligence*, 2022.

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

Lynch, P. The origins of computer weather prediction and climate modeling. *Journal of computational physics*, 227 (7):3431–3444, 2008.

Mansfield, L. A., Nowack, P. J., Kasoar, M., Everitt, R. G., Collins, W. J., and Voulgarakis, A. Predicting global patterns of long-term climate change from short-term simulations using machine learning. *npj Climate and Atmospheric Science*, 3(1):1–9, 2020.

Markou, S., Requeima, J., Bruinsma, W. P., Vaughan, A., and Turner, R. E. Practical conditional neural processes via tractable dependent predictions. *arXiv preprint arXiv:2203.08775*, 2022.

Masson-Delmotte, V., Zhai, P., Pirani, A., Connors, S. L., Péan, C., Berger, S., Caud, N., Chen, Y., Goldfarb, L., Gomis, M., et al. Climate change 2021: the physical science basis. *Contribution of working group I to the sixth assessment report of the intergovernmental panel on climate change*, 2, 2021.

Meehl, G. A., Boer, G. J., Covey, C., Latif, M., and Stouffer, R. J. The coupled model intercomparison project (cmip). *Bulletin of the American Meteorological Society*, 81(2): 313–318, 2000.

Miralles, D. G., Gentine, P., Seneviratne, S. I., and Teuling, A. J. Land–atmospheric feedbacks during droughts and heatwaves: state of the science and current challenges. *Annals of the New York Academy of Sciences*, 1436(1): 19–35, 2019.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 8024–8035. Curran Associates, Inc., 2019.

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.

Phillips, N. A. The general circulation of the atmosphere: A numerical experiment. *Quarterly Journal of the Royal Meteorological Society*, 82(352):123–164, 1956.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pp. 8748–8763. PMLR, 2021.

Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., and Chen, M. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

Rasp, S. and Thuerey, N. Data-driven medium-range weather prediction with a resnet pretrained on climate simulations: A new model for weatherbench. *Journal of Advances in Modeling Earth Systems*, 13(2): e2020MS002405, 2021.

Rasp, S., Dueben, P. D., Scher, S., Weyn, J. A., Mouatadid, S., and Thuerey, N. Weatherbench: a benchmark data set for data-driven weather forecasting. *Journal of Advances in Modeling Earth Systems*, 12(11):e2020MS002203, 2020.

Ravishankara, A., Randall, D. A., and Hurrell, J. W. Complex and yet predictable: The message of the 2021 nobel prize in physics. *Proceedings of the National Academy of Sciences*, 119(2):e2120669119, 2022.

Ravuri, S., Lenc, K., Willson, M., Kangin, D., Lam, R., Mirowski, P., Fitzsimons, M., Athanassiadou, M., Kashem, S., Madge, S., et al. Skilful precipitation nowcasting using deep generative models of radar. *Nature*, 597(7878):672–677, 2021.

Reed, C. J., Gupta, R., Li, S., Brockman, S., Funk, C., Clipp, B., Candido, S., Uyttendaele, M., and Darrell, T. Scale-mae: A scale-aware masked autoencoder for multi-scale geospatial representation learning. *arXiv preprint arXiv:2212.14532*, 2022a.

Reed, S., Zolna, K., Parisotto, E., Colmenarejo, S. G., Novikov, A., Barth-maron, G., Giménez, M., Sulsky, Y., Kay, J., Springenberg, J. T., Eccles, T., Bruce, J., Razavi, A., Edwards, A., Heess, N., Chen, Y., Hadsell, R., Vinyals, O., Bordbar, M., and de Freitas, N. A generalist agent. *Transactions on Machine Learning Research*, 2022b. URL https://openreview.net/forum?id=1ikK0kHjvj. Featured Certification.

Reichstein, M., Camps-Valls, G., Stevens, B., Jung, M., Denzler, J., Carvalhais, N., et al. Deep learning and process understanding for data-driven earth system science. *Nature*, 566(7743):195–204, 2019.

Rodrigues, E. R., Oliveira, I., Cunha, R., and Netto, M. Deepdownscale: a deep learning strategy for high-resolution weather forecast. In *2018 IEEE 14th International Conference on e-Science (e-Science)*, pp. 415–422. IEEE, 2018.

Ronneberger, O., Fischer, P., and Brox, T. U-Net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241. Springer, 2015.

Rosenzweig, C., Karoly, D., Vicarelli, M., Neofotis, P., Wu, Q., Casassa, G., Menzel, A., Root, T. L., Estrella, N., Seguin, B., et al. Attributing physical and biological impacts to anthropogenic climate change. *Nature*, 453 (7193):353–357, 2008.

Sachindra, D., Ahmed, K., Rashid, M. M., Shahid, S., and Perera, B. Statistical downscaling of precipitation using machine learning techniques. *Atmospheric research*, 212: 240–258, 2018.

Satoh, M. *Atmospheric circulation dynamics and circulation models*. Springer Science & Business Media, 2004.

Scher, S. Toward data-driven weather and climate forecasting: Approximating a simple general circulation model with deep learning. *Geophysical Research Letters*, 45 (22):12–616, 2018.

Scher, S. and Messori, G. Weather and climate forecasting with neural networks: using general circulation models (gcms) with different complexity as a study ground. *Geoscientific Model Development*, 12(7):2797–2809, 2019.

Schneider, T., Lan, S., Stuart, A., and Teixeira, J. Earth system modeling 2.0: A blueprint for models that learn from observations and targeted high-resolution simulations. *Geophysical Research Letters*, 44(24):12–396, 2017.

Schultz, M. G., Betancourt, C., Gong, B., Kleinert, F., Langguth, M., Leufen, L. H., Mozaffari, A., and Stadtler, S. Can deep learning beat numerical weather prediction? *Philosophical Transactions of the Royal Society A*, 379 (2194):20200097, 2021.

Sillmann, J., Thorarinsdottir, T., Keenlyside, N., Schaller, N., Alexander, L. V., Hegerl, G., Seneviratne, S. I., Vautard, R., Zhang, X., and Zwiers, F. W. Understanding, modeling and predicting weather and climate extremes: Challenges and opportunities. *Weather and climate extremes*, 18:65–74, 2017.

Sønderby, C. K., Espeholt, L., Heek, J., Dehghani, M., Oliver, A., Salimans, T., Agrawal, S., Hickey, J., and Kalchbrenner, N. MetNet: A neural weather model for precipitation forecasting. *arXiv preprint arXiv:2003.12140*, 2020.

Taori, R., Dave, A., Shankar, V., Carlini, N., Recht, B., and Schmidt, L. Measuring robustness to natural distribution shifts in image classification. *Advances in Neural Information Processing Systems*, 33:18583–18599, 2020.

Thrun, S. and Pratt, L. *Learning to learn*. Springer Science & Business Media, 2012.

Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., and Jégou, H. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, pp. 10347–10357. PMLR, 2021.

Vandal, T., Kodra, E., Ganguly, S., Michaelis, A., Nemani, R., and Ganguly, A. R. Deepsd: Generating high resolution climate change projections through single image super-resolution. In *Proceedings of the 23rd acm sigkdd*

international conference on knowledge discovery and data mining, pp. 1663–1672, 2017.

Vandal, T., Kodra, E., and Ganguly, A. R. Intercomparison of machine learning methods for statistical downscaling: the case of daily and extreme precipitation. *Theoretical and Applied Climatology*, 137(1):557–570, 2019.

Vaughan, A., Tebbutt, W., Hosking, J. S., and Turner, R. E. Convolutional conditional neural processes for local climate downscaling. *arXiv preprint arXiv:2101.07950*, 2021.

Verkuil, R., Kabeli, O., Du, Y., Wicky, B. I., Milles, L. F., Dauparas, J., Baker, D., Ovchinnikov, S., Sercu, T., and Rives, A. Language models generalize beyond natural proteins. *bioRxiv*, pp. 2022–12, 2022.

Vitart, F. and Robertson, A. W. The sub-seasonal to seasonal prediction project (s2s) and the prediction of extreme events. *npj Climate and Atmospheric Science*, 1(1):1–7, 2018.

Vitart, F., Robertson, A. W., Spring, A., Pinault, F., Roškar, R., Cao, W., Bech, S., Bienkowski, A., Caltabiano, N., Coning, E. D., Denis, B., Dirkson, A., Dramsch, J., Dueben, P., Gierschendorf, J., Kim, H. S., Nowak, K., Landry, D., Lledó, L., Palma, L., Rasp, S., and Zhou, S. Outcomes of the WMO prize challenge to improve subseasonal to seasonal predictions using artificial intelligence. *Bulletin of the American Meteorological Society*, 103(12):E2878–E2886, December 2022. doi: 10.1175/bams-d-22-0046.1.

Wang, W., Bao, H., Dong, L., Bjorck, J., Peng, Z., Liu, Q., Aggarwal, K., Mohammed, O. K., Singhal, S., Som, S., et al. Image as a foreign language: Beit pretraining for all vision and vision-language tasks. *arXiv preprint arXiv:2208.10442*, 2022.

Warner, T. T. *Numerical weather and climate prediction*. cambridge university press, 2010.

Watson-Parris, D., Rao, Y., Olivié, D., Seland, Ø., Nowack, P., Camps-Valls, G., Stier, P., Bouabid, S., Dewey, M., Fons, E., et al. Climatebench v1. 0: A benchmark for data-driven climate projections. *Journal of Advances in Modeling Earth Systems*, 14(10):e2021MS002954, 2022.

Weber, T., Corotan, A., Hutchinson, B., Kravitz, B., and Link, R. Deep learning for creating surrogate models of precipitation in earth system models. *Atmospheric Chemistry and Physics*, 20(4):2303–2317, 2020.

Wedi, N., Bauer, P., Denoninck, W., Diamantakis, M., Hamrud, M., Kuhnlein, C., Malardel, S., Mogensen, K., Mozdzynski, G., and Smolarkiewicz, P. *The modelling infrastructure of the Integrated Forecasting System: Recent advances and future challenges*. European Centre for Medium-Range Weather Forecasts, 2015.

Weyn, J. A., Durran, D. R., and Caruana, R. Improving data-driven global weather prediction using deep convolutional neural networks on a cubed sphere. *Journal of Advances in Modeling Earth Systems*, 12(9): e2020MS002109, 2020.

Weyn, J. A., Durran, D. R., Caruana, R., and Cresswell-Clay, N. Sub-seasonal forecasting with a large ensemble of deep-learning weather prediction models. *Journal of Advances in Modeling Earth Systems*, 13(7): e2021MS002502, 2021.

Wightman, R. Pytorch image models. https://github.com/rwightman/pytorch-image-models, 2019.

Wilby, R. L. and Wigley, T. M. Downscaling general circulation model output: a review of methods and limitations. *Progress in physical geography*, 21(4):530–548, 1997.

Yuan, L., Chen, D., Chen, Y.-L., Codella, N., Dai, X., Gao, J., Hu, H., Huang, X., Li, B., Li, C., et al. Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432*, 2021.

Yuan, Y. and Lin, L. Self-supervised pretraining of transformers for satellite image time series classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:474–487, 2020.

Zhai, X., Kolesnikov, A., Houlsby, N., and Beyer, L. Scaling vision transformers. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12104–12113, 2022.

Zhang, C., Zhang, M., Zhang, S., Jin, D., Zhou, Q., Cai, Z., Zhao, H., Liu, X., and Liu, Z. Delving deep into the generalization of vision transformers under distribution shifts. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7277–7286, 2022.

Zhang, F., Sun, Y. Q., Magnusson, L., Buizza, R., Lin, S.-J., Chen, J.-H., and Emanuel, K. What is the predictability limit of midlatitude weather? *Journal of the Atmospheric Sciences*, 76(4):1077–1091, 2019.

Zhuang, J. xesmf: Universal regridder for geospatial data, 2018.

A. Detailed Background and Related Work

Current weather and climate models in use today rely extensively on numerical methods and computational simulations to predict and understand the Earth's weather and climate systems. These tasks include various *numerical weather prediction* (NWP) systems which use computer simulations to make short-term forecasts of weather conditions as well as climate models which use similar techniques to simulate and predict the long-term changes in the Earth's climate. Most notably, at the core of both weather and climate models lie the same set of primitive equations.

For climate modeling, earth system models (ESM) (Hurrell et al., 2013), or "coupled models", that couple together simulations which govern the atmosphere, cryosphere, land, and ocean processes are considered the state-of-the-art. Primarily these simulations are based on general circulation models (GCMs) (Satoh, 2004; Lynch, 2008; Adopted, 2014; Masson-Delmotte et al., 2021) which date back to the works of Phillips (1956); Lorenz (1967) solving Navier-Stokes equations on a rotation sphere to model fluid circulation. These models are often used to perform various *factor sensitivity* studies to examine how the changes in certain forcing factors like greenhouse gas concentrations can affect the global or regional climate and help in *climate projections* to help understand future conditions.

Numerical Weather Prediction (NWP) models share many components of GCMs, especially the atmospheric components (Bauer et al., 2015; Lynch, 2008; Kalnay, 2003). However, incorporating *data assimilation* (Law et al., 2015; Grover, 2022) which involves combining observations and various measurements of the atmosphere and oceans together with these numerical models is important for accurate forecasts and simulations. Another significant distinction between weather and climate models is the framing of the solution for underlying equations: *initial value problem* for weather, while *boundary value problem* for climate (Bauer et al., 2015). Different difficulty levels of these solution approaches results in the fact where climate models tend to be global often at coarser spatio-temporal resolutions while weather models can range from global to local and regional models of very high spatio-temporal resolutions (Warner, 2010).

Despite their noted success, including the recent 2021 Nobel Prize in Physics (Ravishankara et al., 2022), there is considerable debate around the limitations of general circulation models (GCMs), particularly structural errors across models and the fact that current GCMs are designed to reproduce observed climate (Balaji et al., 2022). The climate science community has been aware of these challenges which resulted in the creation of Coupled Model Intercomparison Project (CMIP) as a standardized protocol for evaluating and comparing the performance of different climate models (Meehl et al., 2000). As we will see in the following sections, not only has CMIP been playing a crucial role in the advancement of our understanding of climate change and its potential impacts, its evaluation procedure has resulted in enormous quantity of data making modern deep learning based approaches quite attractive for many tasks. Notably, encoding this knowledge into a "foundation" machine learning model with much faster inference and data assimilation capabilities can pave the way for a much wider impact.

A.1. Data sources

Unlike data in computer vision or natural language processing, weather and climate data is not solely based on sensed data, instead incorporates information from a diverse range of sources. For example, *reanalysis* weather data blends meteorological observations with past short-range weather forecasts via data assimilation (Bauer et al., 2015). The data measurements themselves are highly heterogeneous, representing various physical variables with different data types (e.g. pressure, temperature, humidity) that are recorded at different, relatively sparse, spatial locations at different temporal frequencies. These measurements can be integrated together with known physics inform the design of climate simulations, which again produce data with different variables at different scales. From a machine learning perspective, the plethora of available data thus spans multiple axes: from direct weather measurements at land, sea, or atmosphere, over multiple decades of re-analyzed weather data at different spatial scales, to physics-informed climate projections for various scenarios. Most notably, the data shares the same set of primitive equations, but with fairly different characteristics. Below we describe two of the most commonly used data sources for weather and climate modeling.

A.1.1. CMIP6

The Coupled Model Intercomparison Project (CMIP) (Meehl et al., 2000) is an international effort across different individual climate modeling groups to come together to compare and evaluate their global climate models. While the main goal of CMIP is to improve the understanding of Earth's climate system and improve the accuracy of its simulations, the recent data from their experimental runs is easily accessible on the CMIP6 (Eyring et al., 2016) archive. In CMIP6, where "6" refers to the most recent phase of the project, 49 groups are involved with their experiments covering wide range of climate variables

including temperature, precipitation, sea level and others from hundreds of models. This results in global projections of various climate scenarios from as early as 1850 onwards, all following similar governing equations, but with different *forcings*, e.g., greenhouse gas emissions that affect the climate.

A.1.2. ERA5

The ERA5 reanalysis archive (Hersbach et al., 2018; 2020) of the European Center for Medium-Range Weather Forecasting (ECMWF) is the predominant data source for learning and benchmarking weather forecasting systems. Once completed, the ERA5 reanalysis is set to embody a detailed record of the global atmosphere, land surface and ocean waves from 1950 onwards. The currently available ERA5 reanalysis data combines the state of the art forecasting model called Integrated Forecasting System (IFS) (Wedi et al., 2015) of ECMWF with available observations to provide the best guess of the state of the atmosphere, ocean-wave and land-surface quantities at any point in time. In its raw form, the available reanalyzed data is huge: 40 years, from 1979 to 2018, on a $0.25^\circ \times 0.25^\circ$ global latitude-longitude grid of the Earth's sphere, at hourly intervals with different climate variables at 37 different altitude levels plus the Earth's surface. The grid overall contains 721×1440 grid points for latitude and longitude, respectively. The altitude levels are presented as pressure levels.

A.2. Tasks

Given the scale of data availability, increasing compute requirements of current numerical methods despite it being difficult to incorporate real observational data into them, machine learning is increasingly finding applications in many of the tasks related to weather and climate modeling. When it comes to **weather**, the main task of interest is *forecasting* the future values of key weather variables. These tasks can take the following forms depending on temporal and spatial horizons of interest:

- **Global forecasting** tasks that range from a few hours (i.e., nowcasting) to days and weeks in lead time (i.e., short and medium range forecasting). Often these tasks are evaluated on the ERA5 reanalysis dataset (see Appendix A.1.2) with Operational IFS (Wedi et al., 2015) of the European Center for Medium-Range Weather Forecasting (ECMWF) being the current state-of-the-art NWP baselines.

- **Regional forecasting** tasks which could range from weather forecasting in continental North America or Europe to individual state, county or city.

- **Sub-seasonal to seasonal prediction (S2S)** (Vitart & Robertson, 2018; Vitart et al., 2022) which is the task of forecasting the weather with lead times between 2 weeks and 2 months. S2S bridges the gap between weather forecasting and seasonal climate prediction, and is critical to disaster mitigation. Often at such long horizons, predicting instantaneous values of key weather variables can be a difficult task and therefore the focus is often on averaged value of key weather variables over a certain time horizon, e.g. weekly average precipitation.

Whereas deep learning approaches for regional or S2S tasks are scarce, most of the recent and concurrent work focuses on global forecasting tasks. Rasp & Thuerey (2021) were the first to use pretraining on climate simulations to achieve good data-driven medium-range weather prediction with a ResNet (He et al., 2016), Weyn et al. (2020) used CNNs on a cubed sphere for global weather prediction, Weyn et al. (2021) forecast weather sub-seasonally with a large ensemble of deep-learning weather prediction models, Keisler (2022) applied a graph neural network based approach to weather forecasting, Ravuri et al. (2021) use deep generative models of radar for precipitation nowcasting, Arcomano et al. (2020) build a reservoir computing-based, low-resolution, global prediction model, and MetNet (Sønderby et al., 2020) takes as input radar and satellite data to forecast probabilistic precipitation maps. These approaches are complemented by general machine learning models for fluid dynamics (Li et al., 2020; Kochkov et al., 2021; Lu et al., 2021; Brandstetter et al., 2022a;b). Finally, recent state-of-the-art neural weather models such as FourCastNet (Pathak et al., 2022), Pangu-weather (Bi et al., 2022), or GraphCast (Lam et al., 2022), which also perform global forecasting tasks, use the highest resolution 0.25° ERA5 data, and are optimized on the respective hardware resources.

On the other hand, **climate** tasks have to deal with much longer time horizons. Possible categories of tasks where machine learning can help include climate projection and climate model downscaling:

- **Climate projection** is the task of generating estimates of climate change under different future socio-economic scenarios. Usually, this takes the form of figuring out the response of the climate system to different forcing factors such as

greenhouse gases and aerosol emissions. Climate projection is a crucial task in understanding and preparing for the potential impacts of climate change.

While the application of machine learning in this field is still in its early stages, recent efforts have been made to standardize evaluation in this domain. One example of this is ClimateBench (Watson-Parris et al., 2022), which is a benchmark dataset drawing on CMIP6 to provide an evaluation framework for machine learning models that aim to improve the accuracy of climate projections. This benchmark aims to provide a consistent and reliable evaluation method for various machine learning models that are applied to climate projections.

- A more popular application of ideas in machine learning is towards **downscaling** of climate model. Global climate models typically have a coarse spatial resolution, which means that they can only provide a rough estimate of climate conditions at a local or regional scale. Moreover, the simulations often reflect systematic biases that deviate from trends in the observation data. The aim of climate model downscaling is to create locally accurate climate information from global climate projections by relating those to observed local climatological conditions. This process improves the spatial and temporal resolution of the data, making it more suitable for use in local and regional analyses. Downscaling methods can be dived into *dynamic* approaches that relate outputs of global climate models with those of regional climate models, and *statistical* approaches that infer the desired transformations using data-driven approaches (Wilby & Wigley, 1997). Dynamic approaches are physically consistent, but can be slow and have large biases, whereas statistical approaches need large amounts of data to learn expressive mappings that are hold for target output scenarios.

Similar to weather forecasting, deep learning has emerged as appealing alternative in climate science as well. Recent approaches comprise surrogate models to emulate climate projections (Weber et al., 2020; Scher & Messori, 2019; Scher, 2018; Beusch et al., 2020; Mansfield et al., 2020), extract contextual cues from existing datasets or simulations (Reichstein et al., 2019; Huntingford et al., 2019; Schneider et al., 2017), and perform climate model downscaling (Sachindra et al., 2018; Vandal et al., 2017; Baño-Medina et al., 2020). Climate model downscaling usually inputs low-resolution reanalysis data and local orographic information to obtain high-resolution local information. Many recent approaches are based on convolutional architectures (Höhlein et al., 2020; Vaughan et al., 2021; Markou et al., 2022).

A.3. Foundation models

Bommasani et al. (2021) gave the term "foundation models" to the emerging paradigm of training scalable deep learning models on broad data via self-supervision which could then be adapted (often via finetuning) to a wide range of downstream tasks. Current notable examples include BERT (Devlin et al., 2018), GPT (Brown et al., 2020) and PaLM (Chowdhery et al., 2022), in language, CLIP (Radford et al., 2021), Florence (Yuan et al., 2021), BEiT (Wang et al., 2022) for vision-language. Outside applications on data crawled from web, this paradigm has also started finding success in various scientific domains like protein design (Verkuil et al., 2022). Key significance of such models has been identified as *emergence* with respect to model capabilities and *homogenization* with respect to methodologies for different tasks, domains, and modalities, enabled by the principles of transfer learning (Thrun & Pratt, 2012) at scale. While a foundation model itself should be considered incomplete, it can provide a common basis from which various task-specific models can be derived. Current research at the intersection of weather and climate science and ML has largely focused on designing separate models for every task of interest despite potential availability of fairly diverse large scale data with shared underlying physics and geology across these tasks. A few recent works have proposed pretraining techniques for satellite imagery and remote sensing (Yuan & Lin, 2020; Cong et al., 2022; Reed et al., 2022a) but they have so far not been applied to multi-sensory data and variables in weather and climate.

B. Discussion and Future Work

The scaling of datasets, model architectures, and computation has resulted in a transformative impact in various subdisciplines of artificial intelligence, from natural language processing to computer vision, as well as scientific applications. In particular, it has led to the emergence of general-purpose foundation models that are trained on large datasets and compute clusters, and can be efficiently adapted to a variety of downstream tasks, both in terms of compute and data supervision. ClimaX represents a pioneering effort to enable such broad scaling and generality in data-driven models for weather and climate that goes beyond limitations of both traditional numerical modeling and existing data-driven forecasting methods. Unlike ClimaX, numerical models scale only in terms of computation and not in terms of dataset size, whereas existing data-driven models are typically limited to specific tasks and lack general-purpose applicability across a wide range of tasks.

In addition to traditional considerations in language and vision, foundation models like ClimaX open up new opportunities for scaling through the use of simulation datasets and grid resolutions. To simplify our approach, we chose to use pretraining datasets that include standard variables that have been benchmarked in previous research on data-driven forecasting (Rasp et al., 2020; Pathak et al., 2022). Additionally, we avoided datasets that simulate future scenarios under different forcings to prevent any potential leakage for the climate projection task. Future research could explore incorporating both observational and simulated datasets that include a wider range of climate variables, higher spatiotemporal resolutions, and even extend into future scenarios. Further, we showed that resolution plays a crucial role in scaling of ClimaX. Due to our compute restrictions, we trained ClimaX on low to moderate resolutions. Nevertheless, our empirical trends suggest that scaling to higher resolutions ($0.25°$) is likely to lead to even better results. Future scaling efforts can benefit from better sequence modeling architectures, especially those designed for multimodal spatiotemporal inputs. As we saw in ClimaX, the number of channels for climate datasets is much larger than those in standard multimodal settings (e.g., audio-video, vision-language). Moreover, in practice, there is also a significant range of resolutions across different climate datasets. This heterogeneity drastically increases the raw length of input sequences for standard architectures such as ViT. In the future, we believe that investigating single multi-scale architectures (e.g., Fan et al. (2021)) can potentially aid in scaling to such diverse multi-resolution and multi-modal datasets by learning to infer features relevant to atmospheric phenomena at increasing spatial resolutions.

In conclusion, we believe that the generality of our approach has potential applications beyond the tasks considered in this work. It would be interesting to explore the generalization of a pretrained ClimaX backbone to other Earth systems science tasks, such as predicting extreme weather events (Miralles et al., 2019; Sillmann et al., 2017) and assessing anthropogenic contributions to climate change (Rosenzweig et al., 2008; Höök & Tang, 2013), as well as broader domains that are closely tied to weather and climate conditions, such as agriculture, demography, and actuarial sciences.

C. Additional experiments

Neural baselines. In global forecasting, we compare ClimaX with IFS (Wedi et al., 2015), the current gold standard in weather forecasting. In tasks we do not have a baseline, we compare with two CNN based baselines UNet (Ronneberger et al., 2015) and ResNet (He et al., 2016) with specific architecture details borrowed from (Gupta & Brandstetter, 2022; Rasp et al., 2020), as described in Appendix D.2.

C.1. Forecasting

Regional forecasting. We next evaluate ClimaX on *regional forecasting* of relevant variables in North America, where the task is to forecast the future weather in North America given the current weather condition in the same region. We create a new dataset from the ERA5 data at $1.40625°$ that has the same set of variables but just focuses on the North America region. We call this dataset ERA5-NA and present details of how to construct it in Appendix F.2. Training, validation, and test splits are done similarly to Section 3. Since the task has not been considered in previous works, we compare ClimaX with the two CNN baselines ResNet and UNet, and the scratch-trained version of ClimaX, which we refer to as Cli-ViT. In addition, we finetune two ClimaX models, in which one was pretrained on CMIP6 at $1.40625°$, and the other was pretrained on $5.625°$ data. To finetune the low-resolution model on higher-resolution data, we follow the common practice of interpolating the positional embedding (Dosovitskiy et al., 2020; Touvron et al., 2021). We denote this model as ClimaX-pos-interp. We evaluate all methods on predicting Z500, T2m, and T850 at lead times of 3, 5, and 7 days. Latitude-weighted RMSE is used as the evaluation metric.

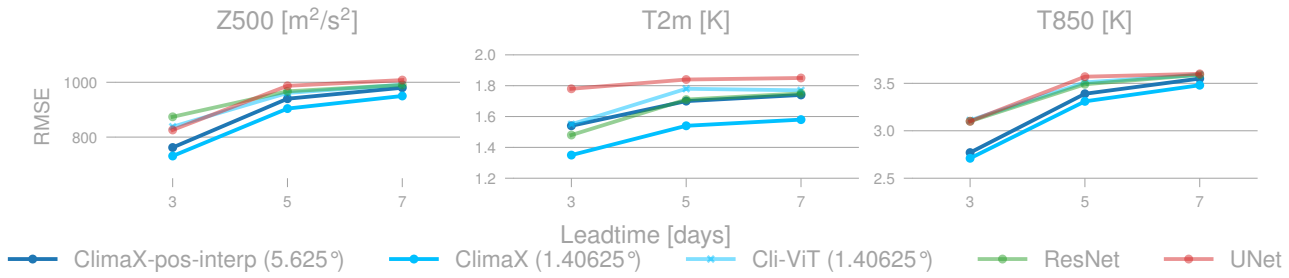

Figure 5. Regional (North America) forecasting performance.

Figure 5 compares the performance of ClimaX and the baselines. ClimaX is the best performing method among different target variables and lead times. Interestingly, even though pretrained on data at a lower resolution, ClimaX-pos-interp achieves the second best performance in predicting Z500 and T850, and only underperforms ResNet in predicting T2m at 3-day lead time. This result shows that ClimaX can gain strong performance on tasks that have different spatial coverage or even different spatial resolution from pretraining.

Sub-seasonal to seasonal prediction. Sub-seasonal to seasonal (S2S) prediction is the task of forecasting at a time range between 2 weeks and 2 months (Vitart & Robertson, 2018), which bridges the gap between weather forecasting and climate projection. Compared to the other two well-established tasks, S2S prediction has received much less attention, despite having a significant socioeconomic value in disaster mitigation efforts. To the best of our knowledge, S2S prediction has not been considered in previous deep learning works. Here, following the S2S competition (https://s2s-ai-challenge.github.io/), we aim to predict the biweekly average statistics of weeks 3-4 and weeks 5-6, which correspond to lead times of 2 weeks and 4 weeks, respectively. We construct ERA5-S2S, a new dataset from 5.625° ERA5 that has the same input variables, but the output variables are averaged from the lead time to 2 weeks ahead into the future. We compare ClimaX with ResNet, UNet, and Cli-ViT on the S2S prediction of four target variables: T850, T2m, U10, and V10.

Table 2. RMSE of ClimaX and baselines on 5.625° ERA5-S2S prediction tasks.

	T850		T2m		U10	
	Weeks 3-4	Weeks 5-6	Weeks 3-4	Weeks 5-6	Weeks 3-4	Weeks 5-6
ResNet	2.12	2.13	1.88	2.16	1.91	1.94
UNet	1.91	1.95	1.67	1.79	1.85	1.90
Cli-ViT	1.96	1.96	1.79	1.90	1.83	1.92
ClimaX	**1.89**	**1.92**	**1.66**	**1.70**	**1.81**	**1.86**

Table 2 compares the RMSE of ClimaX and the baselines. ClimaX achieves the lowest error for all variables, and the performance gap with the best baseline UNet is larger at increasing lead times. ClimaX also has significant performance gains over its scratch-trained counterpart Cli-ViT, showing the effectiveness of our pretraining procedure in capturing features that are generally useful for various temporal prediction tasks.

C.2. Climate model downscaling

Climate models are often run at coarse grids due to their high computational cost. Although these predictions are useful in understanding large-scale climate trends, they do not provide sufficient detail to analyze regional and local phenomena. Downscaling aims to obtain higher-resolution projections and reduce biases from the outputs of these models. To evaluate the applicability of ClimaX to the task of climate model downscaling, we construct a new dataset based on CMIP6 and ERA5 data sources for coarse inputs and higher resolution targets. Specifically, we use all MPI-ESM, a dataset from CMIP6, and its variables listed in Table 8 at 5.625° as input, and train separate models to downscale to each ERA5 target variable at 1.40625°. We compare ClimaX with Cli-ViT and the two CNN baselines, UNet and ResNet, as most recent deep downscaling methods (Vandal et al., 2017; Rodrigues et al., 2018; Höhlein et al., 2020; Vandal et al., 2019; Liu et al., 2020) are based on convolution. We were not able to compare with YNet (Liu et al., 2020), the current best method on deep downscaling as we did not have access to high-resolution auxiliary data such as elevation and topographical information. For all methods, we first bilinearly interpolate the input to match the resolution of the desired output before feeding it to the model. We evaluate all methods on RMSE, Pearson correlation, and Mean bias, which were commonly used in existing deep downscaling works (Vandal et al., 2017; Liu et al., 2020). Details of the metrics are in Appendix G.1.

Table 3. Performance of ClimaX and the baselines on downscaling from MPI-ESM (5.625°) to ERA5 (1.40625°).

	Z500			T850			T2m		
	RMSE	Pearson	Mean bias	RMSE	Pearson	Mean bias	RMSE	Pearson	Mean bias
ResNet	825.75	0.96	−108.54	3.60	0.96	0.19	2.89	0.98	0.14
UNet	858.35	0.95	35.10	3.66	0.96	−0.34	2.95	0.98	0.16
Cli-ViT	811.61	0.96	−54.32	3.58	0.97	−0.29	2.80	0.99	−0.06
ClimaX	**807.43**	0.96	**2.70**	**3.49**	0.97	**−0.11**	**2.79**	0.99	**−0.06**

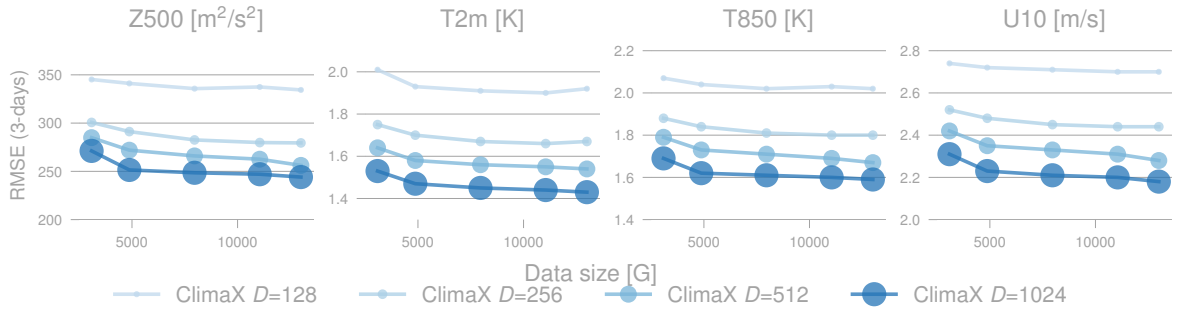

Figure 6. ClimaX scaling performance with respect to model capacity. The RSME on ERA5 3-day forecasting is shown for different variables as a function of the number of CMIP6 5.625° data seen during pre-training. Curves indicate different model sizes.

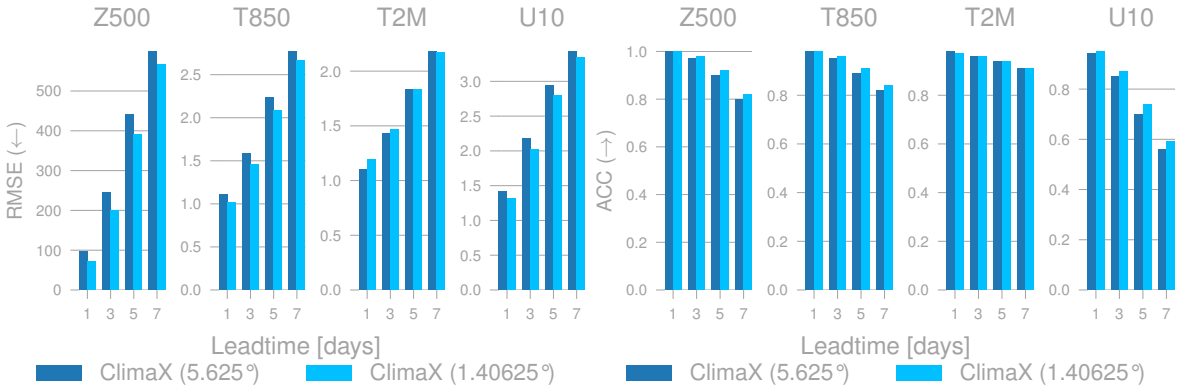

Figure 7. Scaling performance with respect to data resolution. ClimaX (1.40625°) achieves consistently better performance.

Table 3 compares ClimaX and the baselines quantitatively. ClimaX achieves the lowest RMSE and a mean bias closest to 0 for all three target variables, and performs similarly to the baselines in terms of Pearson correlation. During pretraining ClimaX has successfully captured the spatial structure of weather data, which helps in downstream tasks like downscaling. Figure 18 in Appendix visualizes the downscaled predictions of ClimaX for the three target variables. The input is at a much lower resolution and contains a lot of bias compared to the ground truth. While the prediction is missing some fine details, it has successfully captured the general structure of the ERA5 data and removed input biases.

C.3. Scaling laws analysis

Transformers have shown favorable scaling properties for language (Kaplan et al., 2020; Hoffmann et al., 2022), vision (Zhai et al., 2022), or even multi-modal tasks (Henighan et al., 2020; Hendricks et al., 2021; Reed et al., 2022b). That is, their performance improves with respect to data size and model capacity given sufficient compute. In this section, we study the scaling laws of ClimaX. Figure 6 presents the performance of ClimaX as a function of data size and model capacity. The x-axis is the pretraining data size measured in Gigabytes, and corresponding to 1 to 5 CMIP6 datasets. The y-axis shows the RMSE of ClimaX on the 3-day forecasting task, comparing four ClimaX models of different size by varying the embedding dimension from 128 to 1024. All experiments are conducted on the 5.625° data. RMSE values of the two biggest models decreases consistently as we increase the data and model size, highlighting the ability of ClimaX to learn from diverse and heterogeneous data sources. For the two smaller models increasing data size does not gain much improvement, and can sometimes event hurt performance. These results show that larger models not only perform better by simply pretraining on more data, but are also more data efficient.

In addition to data size and model capacity, data resolution is another important scaling dimension in the context of weather and climate. In many vision tasks such as classification, understanding the general, high-level structure of the image is sufficient to make accurate predictions. However, to model the underlying complex physical processes that govern weather and climate, it is important to capture fine-grained details. High-resolution data contains finer details and local processes

of weather conditions that are not present in the low-resolution data, and thus provides stronger signals for training deep learning models. Figure 7 compares the performance of ClimaX pretrained and finetuned on $5.625°$ and $1.40625°$ data on global forecasting. Except for T2m at 1 day and 3 days lead times, ClimaX ($1.40625°$) consistently achieves lower RMSE and higher ACC than the low-resolution model. We note that for the high-resolution data we have to use a larger patch size (4 compared to 2 for low-resolution data) due to lack of memory issue. We can further improve the performance of ClimaX on the $1.40625°$ data by reducing the patch size, as the model is able to capture better details.

C.4. Ablation studies

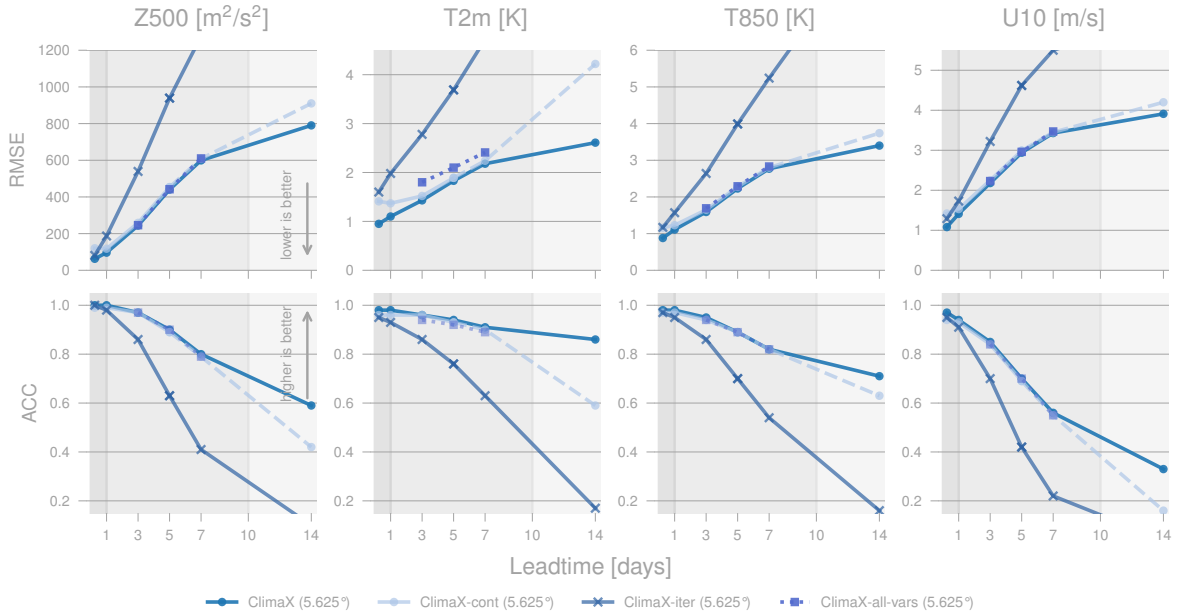

Figure 8. Performance of ClimaX and its variations on weather forecasting. ClimaX-cont is a lead-time-conditioned model that we finetune to make predictions at 6 hours to 7 days. ClimaX-iter forecasts at a 6-hour lead time and rolls out the predictions to forecast at longer horizons. ClimaX-all-vars predicts the future conditions of all variables in the input at particular lead-times.

In the main forecasting results, we finetune a separate ClimaX model for each target variable at each lead time, as we found this protocol led to the best performance. However, this can be computationally expensive, as finetuning cost scales linearly with respect to the number of target variables and lead times. In this section, we consider different finetuning alternatives to investigate the trade-off between computation and performance.

C.4.1. Should we finetune ClimaX for each variable separately or all at once?

Instead of finetuning ClimaX for each target variable separately, we could alternatively finetune once to predict all variables in the input simultaneously, which we denote as ClimaX-all-vars. Figure 8 shows that ClimaX-all-vars achieves comparable performance to ClimaX in most of the tasks and only underperforms for forecasting T2m. This suggests that with a limited budget, one can finetune ClimaX to predict all target variables at the same time without losing much performance.

C.4.2. Should we do iterative forecast or direct forecast?

To avoid finetuning a different model for each lead time, we can finetune ClimaX to make predictions at a short horizon such as 6 hours, and roll out the predictions during inference to make forecasts at longer horizons. We call this model ClimaX-iter, where *iter* stands for iterative prediction (Rasp et al., 2020). We note that in order to roll out more than one step, ClimaX-iter must predict for all input variables, or in other words. This provides the benefit of finetuning a single model that can predict for any target variable at any lead time. Figure 8 shows that ClimaX-iter works reasonably well up to 1-day prediction, but the performance degrades significantly at longer lead times. This is not surprising, because ClimaX-iter is not finetuned to predict multiple steps into the future, leading to quick error accumulation. One can employ a multi-step objective for finetuning as in Pathak et al. (2022) to achieve better results.

C.4.3. CAN WE FINETUNE CLIMAX TO WORK FOR ALL LEAD TIMES?

Another way to avoid finetuning for each lead time separately is to finetune a lead-time-conditioned model. Specifically, during finetuning, we randomize the lead time from 6 hours to 7 days, resembling the pretraining setting. Note that unlike ClimaX-iter, we still have to finetune a separate model for each target variable. We call this model ClimaX-cont, wherein *cont* stands for *continuous*, a standard term used in previous works (Rasp et al., 2020). Figure 8 shows that ClimaX-cont performs competitively on 6-hour to 7-day forecasting, but fails to extrapolate to 2 weeks and 1 month lead times that are unseen during training. One can also randomize the lead time from 6 hours to 1 month, but that means the model sees much fewer data points for each target lead time, potentially hurting the performance.

The cost for finetuning each set of weights is a constant C, which is about 15 hours on an $8 \times \text{V100}s$. Among different finetuning protocols, ClimaX is the most expensive, whose total cost is $C \times \#variables \times \#lead_times$, scaling linearly with the number of target variables and lead times. Following ClimaX are ClimaX-all-vars and ClimaX-cont, whose total costs are $C \times \#lead_times$ and $C \times \#variables$, respectively. Finally, ClimaX-iter is the cheapest finetuning protocol, where we only have to finetune a single model that works for all target variables and at all lead times. The performance is proportional to the computational cost, as ClimaX is the best performing model, while ClimaX-iter is the worst.

D. Model details

This section presents the implementation details and hyperparameters of ClimaX and the two CNN baselines UNet and ResNet.

D.1. ClimaX

D.1.1. IMPLEMENTATION DETAILS

ClimaX receives a tensor of shape $V \times H \times W$ and outputs a tensor of shape $V' \times H \times W$, where the number of input and output variables V and V' can vary between different datasets[2]. To do that, we assume a set $\mathcal{V}$ that contains all possible variables we could encounter during pretraining and finetuning. Each variable in $\mathcal{V}$ has a separate token embedding layer.

The variable tokenization module tokenizes the input to a sequence of $V \times h \times w$ tokens, with each token being a vector of size p^2. After that, for each token, we extract the corresponding embedding layer that transforms the token to a vector of dimension D. Each embedding layer is a single convolution layer with $in_channels = 1, out_channels = D, kernel_size = p, stride = p$. This results in a tensor of shape $V \times h \times w \times D$.

To differentiate between tokens of different input variables, we add the sequence with a *variable positional embedding*, which is a tensor of shape $|\mathcal{V}| \times D$. For each input variable, we extract the corresponding variable positional embedding to add to its tokens. After that, all tokens go through the variable aggregation module, which outputs a tensor of shape $h \times w \times D$.

The tokens are then fed to the attention layers, which output a tensor of the same shape $h \times w \times D$. The prediction head takes each token of dimension D and maps it to a vector of dimension $|\mathcal{V}| \times p^2$, and the output is reshaped to $|\mathcal{V}| \times H \times W$. Finally, we extract predictions of V' target variables and compute the loss.

D.1.2. HYPERPARAMETERS

D.2. CNN Baselines

D.2.1. RESNET HYPERPARAMETERS

We use the following hyperparameters for ResNet in all of our experiments.

D.2.2. UNET HYPERPARAMETERS

We borrow our UNet implementation from PDEArena (Gupta & Brandstetter, 2022). We use the following hyperparameters for UNet in all of our experiments.

[2]The spatial resolution $H \times W$ can also vary. In that case, we employ the common practice of interpolating the positional embedding, and everything else remains the same (Dosovitskiy et al., 2020; Touvron et al., 2021).

Table 4. Default hyperparameters of ClimaX

Hyperparameter	Meaning	Value
$\mathcal{V}$	Default variables	All ERA5 variables in Table 9
$\lvert\mathcal{V}\rvert$	Number of default variables	48
p	Patch size	2 for $5.625°$ 4 for $1.40625°$
D	Embedding dimension	1024
Depth	Number of ViT blocks	8
# heads	Number of attention heads	16
MLP ratio	Determine the hidden dimension of the MLP layer in a ViT block	4
Prediction depth	Number of layers of the prediction head	2
Hidden dimension	Hidden dimension of the prediction head	1024
Drop path	For stochastic depth (Huang et al., 2016)	0.1
Dropout	Dropout rate	0.1

Table 5. Default hyperparameters of ResNet

Hyperparameter	Meaning	Value
Padding size	Padding size of each convolution layer	1
Kernel size	Kernel size of each convolution layer	3
Stride	Stride of each convolution layer	1
Hidden dimension	Number of output channels of each residual block	128
Residual blocks	Number of residual blocks	28
Dropout	Dropout rate	0.1

Table 6. Default hyperparameters of UNet

Hyperparameter	Meaning	Value
Padding size	Padding size of each convolution layer	1
Kernel size	Kernel size of each convolution layer	3
Stride	Stride of each convolution layer	1
Channel multiplications	Determine the number of output channels for Down and Up blocks	$[1, 2, 2, 4]$
Blocks	Number of blocks	2
Use attention	If use attention in Down and Up blocks	False
Dropout	Dropout rate	0.1

D.2.3. Other implementation details

Following the implementation of ResNet in Rasp et al. (2020); Rasp & Thuerey (2021); Ernst (2021), we found the following details important for the performance of both CNN baselines:

- Use Batch normalization
- Use Leakyrelu with a slope of 0.3 as the activation function
- Postnorm instead of Prenorm
- Use periodic convolutions in the longitude direction but not the latitude direction.
- Use a kernel size of 7 in the first CNN layer.

E. Training details

Data normalization We normalized all inputs during pre-training as well as fine-tuning. For each variable, at each pressure level (for atmospheric variables), we compute the mean and standard deviation to normalize them to zero mean and unit variance. We de-normalize the predictions to get back to the original range before computing evaluation metrics.

Software and hardware stack We use PyTorch (Paszke et al., 2019), `timm` (Wightman, 2019), `numpy` (Harris et al., 2020) and `xarray` (Hoyer & Hamman, 2017) to manage our data and model training. We used 32GB NVIDIA V100 devices for training. For pretraining we distribute the batch across 80 V100s on AzureML. We leverage `fp16` floating point precision in our model.

E.1. Pretraining

E.1.1. Objective

We use the loss function in Equation (1) for pretraining. Given the prediction $\tilde{X}_{t+\Delta t}$ and the ground truth $X_{t+\Delta t}$, the loss is computed as:

$$\mathcal{L} = \frac{1}{V \times H \times W} \sum_{v=1}^{V} \sum_{i=1}^{H} \sum_{j=1}^{W} L(i)(\tilde{X}^{v,i,j}_{t+\Delta t} - X^{v,i,j}_{t+\Delta t})^2, \tag{1}$$

in which $L(i)$ is the latitude weighting factor:

$$L(i) = \frac{\cos(\text{lat}(i))}{\frac{1}{H}\sum_{i'=1}^{H} \cos(\text{lat}(i'))}, \tag{2}$$

where $\text{lat}(i)$ is the latitude of the corresponding ith row of the grid.

E.1.2. Optimization

We used the AdamW optimizer (Kingma & Ba, 2014; Loshchilov & Hutter, 2017) with parameters ($\beta_1 = 0.9, \beta_2 = 0.95$). We used weight decay of $1e-5$ for all parameters except for the positional embedding. We used a learning rate of $5e-4$, with a linear warmup schedule for 10000 steps (5 epochs), followed by a cosine-annealing schedule for 190000 steps (95 epochs).

E.2. Finetuning

E.2.1. Objective

We use lat-weighted MSE in Equation (1) for finetuning ClimaX in temporal forecasting and downscaling tasks. In ClimateBench, we finetune using standard MSE without the weighting term, as this led to better results and was suggested by (Watson-Parris et al., 2022).

E.2.2. Optimization

For all tasks, we used AdamW with parameters ($\beta_1 = 0.9, \beta_2 = 0.999$). We used weight decay of $1e-5$ for all parameters except for the positional embedding. We used a linear warmup schedule for 10000 steps (5 epochs), followed by a cosine-annealing schedule for 90000 steps (45 epochs). The learning rate for each task is as follows:

Table 7. Learning rate for finetuning ClimaX in different downstream tasks

Task	Learning rate
Weather forecasting	$5e-7$
Climate projection	$5e-4$
Climate downscaling	$5e-5$

We used a small learning rate for weather forecasting as the task resembles pretraining. For downscaling, we used a larger learning rate, as the nature of the task is different from pretraining, even though the input variables are similar. In climate

projection, we needed to initialize new weights for the embedding layers and prediction heads, and thus used a similar learning rate to training from scratch.

E.2.3. FINETUNING CLIMAX FOR REGIONAL FORECASTING

Figure 9 illustrates the finetuning process of ClimaX on this task, where the only difference from global forecasting is the input now only contains tokens that belong to North America.

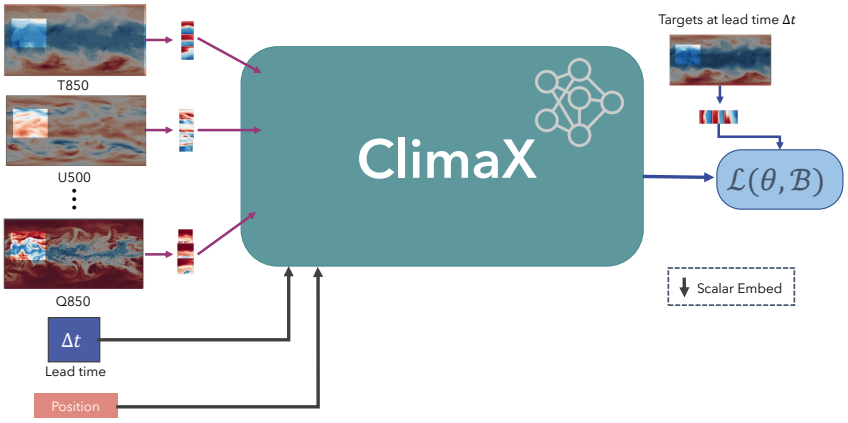

Figure 9. Finetuning setup for Regional Forecasting in North America.

E.2.4. FINETUNING CLIMAX FOR CLIMATE PROJECTION

Figure 10 illustrates the finetuning pipeline of ClimaX for ClimateBench. We introduce two components to the pipeline in Figure 3. We use a history of the preceding ten years of the forcing factors to make predictions for a particular year, creating an input of shape $T \times V \times H \times W$. Each time slice of the input goes through variable tokenization, variable aggregation, and the attention layers as usual, which output a feature tensor of shape $T \times h \times w \times D$, where D is the embedding size. The feature tensor then goes through a global average pooling layer, reducing the dimension to $T \times D$. Finally, the 10-year history is aggregated using a cross-attention layer before being fed to the prediction head, which linearly transforms the D-dimensional feature vector to a $H \times W$ map. The history aggregation and the global pooling modules are the two additions to the original ClimaX architecture. These architectural designs are inspired by the neural network baseline in (Watson-Parris et al., 2022).

F. Datasets

F.1. CMIP6-ClimaX

We created CMIP6-ClimaX for pretraining ClimaX, which consists of 5 datasets from the CMIP6 project. We downloaded the datasets from the official CMIP6 search interface at https://esgf-data.dkrz.de/search/cmip6-dkrz/. These datasets share the following attributes:

- Experiment ID: historical
- Table ID: 6hrPlevPt, i.e., 6-hourly data on pressure levels.
- Variant label: r1i1p1f1. The variant label distinguishes among closely related simulations by a single model, in which "r" specifies the initial condition, "i" specifies the observational dataset and initialization method used for determining the initial condition, "p" specifies the perturbed physics version of the model, and "f" specifies the forcing index.

All datasets have a temporal coverage from 1850 to 2015 and a temporal resolution of 6 hours. We chose these datasets as they contain similar climate variables at similar vertical levels to ERA5. We note that there are more than 5 datasets from CMIP6 that suit our selection criteria, but we were not able to download others due to some issues on the data servers. We

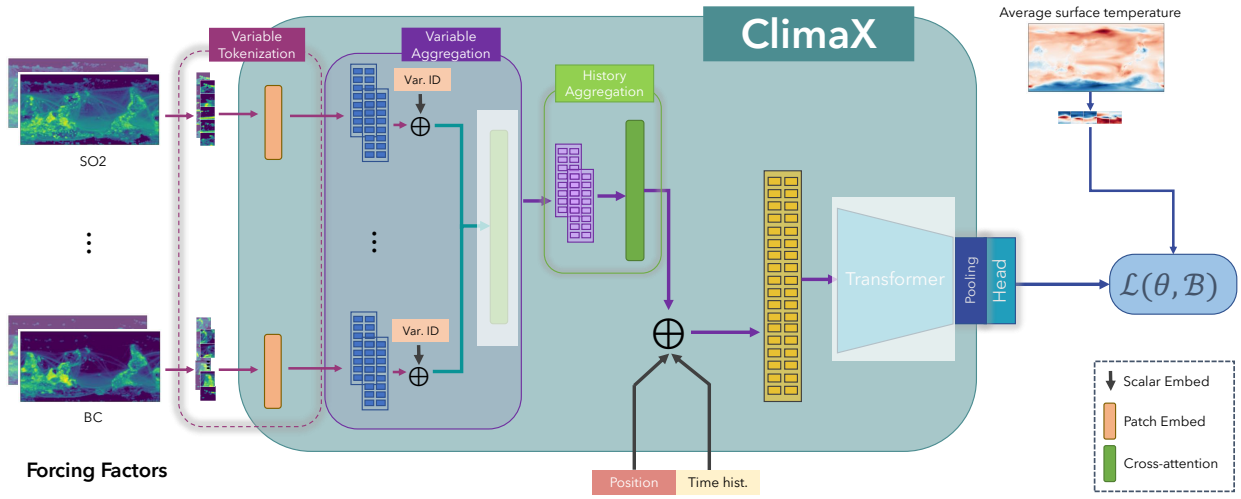

Figure 10. Finetuning pipeline for ClimateBench. A different set of input and output variables requires different embedding layers and prediction heads. Attention layers can be frozen or finetuned.

regridded these datasets to $5.625°$ and $1.40625°$ using the xesmf Python package (Zhuang, 2018) using bilinear interpolation. We provide a detailed description of these 5 data sources and the available variables we used to construct CMIP6-ClimaX in Table 8.

We note that AWI and HAMMOZ are not the best data sources for higher resolution $1.40625°$ training, because their original resolution at 250 km is lower than $1.40625°$, which is about 156 km. We wanted to use other higher-resolution datasets but were not able to download them. We believe pretraining on other high-resolution datasets would lead to better performance.

F.2. ERA5

We use the preprocessed version of ERA5 from WeatherBench (Rasp et al., 2020) for finetuning ClimaX. WeatherBench was created as a standard benchmark data and evaluation framework for comparing data-driven weather forecasting models. WeatherBench regridded the original ERA5 at $0.25°$ to three lower resolutions: $5.625°$, $2.8125°$, and $1.40625°$. See `https://confluence.ecmwf.int/display/CKB/ERA5%3A+data+documentation` for more details of the raw ERA5 data. Table 9 summarizes the variables we use for finetuning ClimaX.

F.2.1. ERA5-NA

We constructed ERA5-NA from ERA5 to evaluate ClimaX and the baselines on regional forecasting. ERA-NA has the same set of variables as in Table 9, but only contains data that belongs to the North America region. To do this, we first identified the latitude and longitude range to form a rectangular area that encapsulates North America, using the standard CORDEX domains `https://cordex.org/wp-content/uploads/2012/11/CORDEX-domain-description_231015.pdf`. For each data sample, we then extracted the spatial positions that fall into this range, forming in ERA5-NA.

F.2.2. ERA-S2S

We built ERA5-S2S from ERA5 to serve as a benchmark dataset for sub-seasonal to seasonal prediction. ERA5-S2S consists of two sub-datasets, whose the goals are to predict the biweekly average statistics of target variables in weeks 3 and 4, and weeks 5 and 6, respectively. The input includes all variables in Table 9, while the output variables are are averaged over two weeks, starting from the start of week 3 (5) and to the end of week 4 (6).

F.3. ClimateBench

We refer to Watson-Parris et al. (2022) for complete details of ClimateBench.

Table 8. Resolution and variables of CMIP6-ClimaX dataset used for pretraining. *Static* represents variables don't depend on time, *Single* represents surface variables, and *Atmospheric* represents time-varying atmospheric properties at the chosen altitudes.

Data Source	Original resolution	Variables		
		Type	Abbrev.	Levels
MPI	100km	Single	t2m	
		Single	u10	
		Single	v10	
		Atmospheric	z	50, 250, 500, 600, 700, 850, 925
		Atmospheric	u	50, 250, 500, 600, 700, 850, 925
		Atmospheric	v	50, 250, 500, 600, 700, 850, 925
		Atmospheric	t	50, 250, 500, 600, 700, 850, 925
		Atmospheric	q	50, 250, 500, 600, 700, 850, 925
Tai	100km	Single	t2m	
		Atmospheric	z	250, 500, 600, 700, 850, 925
		Atmospheric	u	250, 500, 850
		Atmospheric	v	250, 500, 850
		Atmospheric	t	250, 500, 850
		Atmopheric	q	250, 500, 600, 700, 850, 925
AWI	250km	Single	t2m	
		Single	u10	
		Single	v10	
		Atmospheric	z	50, 250, 500, 600, 700, 850, 925
		Atmospheric	u	50, 250, 500, 600, 700, 850, 925
		Atmospheric	v	50, 250, 500, 600, 700, 850, 925
		Atmospheric	t	50, 250, 500, 600, 700, 850, 925
		Atmospheric	q	50, 250, 500, 600, 700, 850, 925
HAMMOZ	250km	Single	t2m	
		Single	u10	
		Single	v10	
		Atmospheric	z	50, 250, 500, 600, 700, 850, 925
		Atmospheric	u	50, 250, 500, 600, 700, 850, 925
		Atmospheric	v	50, 250, 500, 600, 700, 850, 925
		Atmospheric	t	50, 250, 500, 600, 700, 850, 925
		Atmospheric	q	50, 250, 500, 600, 700, 850, 925
CMCC	100km	Atmospheric	z	50, 250, 500, 600, 700, 850, 925
		Atmospheric	u	50, 250, 500, 600, 700, 850, 925
		Atmospheric	v	50, 250, 500, 600, 700, 850, 925
		Atmospheric	t	250, 500, 850

G. Quantitative evaluation

G.1. Metrics

This section presents all evaluation metrics we use in Section 3. For all metrics, we denote $\tilde{X}$ and X as the prediction and ground truth, which have a shape of $N \times H \times W$, where N is the number of forecasts, or the number of test samples, $H \times W$ is the spatial resolution. $L(i)$ is the latitude weighting term to account for the non-uniformity in areas of the grid cells. We have removed the time notation for simplicity.

Table 9. ECMWF variables used in our ERA5 dataset. *Static* represents variables don't depend on time, *Single* represents surface variables, and *Atmospheric* represents time-varying atmospheric properties at the chosen altitudes.

Type	Variable name	Abbrev.	ECMWF ID	Levels
Static	Land-sea mask	LSM	172	
Static	Orography			
Single	2 metre temperature	T2m	167	
Single	10 metre U wind component	U10	165	
Single	10 metre V wind component	V10	166	
Atmospheric	Geopotential	Z	129	50, 250, 500, 600, 700, 850, 925
Atmospheric	U wind component	U	131	50, 250, 500, 600, 700, 850, 925
Atmospheric	V wind component	V	132	50, 250, 500, 600, 700, 850, 925
Atmospheric	Temperature	T	130	50, 250, 500, 600, 700, 850, 925
Atmospheric	Specific humidity	Q	133	50, 250, 500, 600, 700, 850, 925
Atmospheric	Relative humidity	R	157	50, 250, 500, 600, 700, 850, 925

G.1.1. Weather forecasting metrics

Root mean square error (RMSE)

$$\text{RMSE} = \frac{1}{N}\sum_{k=1}^{N}\sqrt{\frac{1}{H\times W}\sum_{i=1}^{H}\sum_{j=1}^{W} L(i)(\tilde{X}_{k,i,j} - X_{k,i,j})^2}. \tag{3}$$

Anomaly correlation coefficient (ACC) Anomaly correlation coefficient (ACC) is the spatial correlation between prediction anomalies $\tilde{X}^{'}$ relative to climatology and ground truth anomalies $X^{'}$ relative to climatology:

$$\text{ACC} = \frac{\sum_{k,i,j} L(i)\tilde{X}^{'}_{k,i,j}X^{'}_{k,i,j}}{\sqrt{\sum_{k,i,j} L(i)\tilde{X}'^{2}_{k,i,j}\sum_{k,i,j} L(i)X'^{2}_{k,i,j}}}, \tag{4}$$

$$\tilde{X}^{'} = \tilde{X}^{'} - C, X^{'} = X^{'} - C, \tag{5}$$

in which climatology C is the temporal mean of the ground truth data over the entire test set $C = \frac{1}{N}\sum_k X$.

G.1.2. Climate projection metrics

Normalized spatial root mean square error (NRMSE$_s$) Normalized spatial root mean square error (NRMSE$_s$) measures the spatial discrepancy between the temporal mean of the prediction and the temporal mean of the ground truth:

$$\text{NRMSE}_s = \sqrt{\left\langle\left(\frac{1}{N}\sum_{k=1}^{N}\tilde{X} - \frac{1}{N}\sum_{k=1}^{N}X\right)^2\right\rangle} \Big/ \frac{1}{N}\sum_{k=1}^{N}\langle X\rangle\,, \tag{6}$$

in which $\langle A\rangle$ is the global mean of A:

$$\langle A\rangle = \frac{1}{H\times W}\sum_{i=1}^{H}\sum_{j=1}^{W} L(i)A_{i,j} \tag{7}$$

Normalized global root mean square error (NRMSE$_g$) Normalized global root mean square error (NRMSE$_g$) measures the discrepancy between the global mean of the prediction and the global mean of the ground truth:

$$\text{NRMSE}_g = \sqrt{\frac{1}{N}\sum_{k=1}^{N}\left(\langle\tilde{X}\rangle - \langle X\rangle\right)^2} \Big/ \frac{1}{N}\sum_{k=1}^{N}\langle X\rangle\,. \tag{8}$$

Total normalized root mean square error (TRMSE) Total normalized root mean square error (TRMSE) is the weighted sum of NRMSE_s and NRMSE_g:

$$\text{TRMSE} = \text{NRMSE}_s + \alpha \cdot \text{NRMSE}_g, \tag{9}$$

where α is chosen to be 5 as suggested by Watson-Parris et al. (2022).

G.1.3. Climate downscaling metrics

Root mean square error (RMSE) This is the same as Equation (3).

Mean bias Mean bias measures the difference between the spatial mean of the prediction and the spatial mean of the ground truth. A positive mean bias shows an overestimation, while a negative mean bias shows an underestimation of the mean value.

$$\text{Mean bias} = \frac{1}{N \times H \times W} \sum_{k=1}^{N} \sum_{i=1}^{H} \sum_{j=1}^{W} \tilde{X} - \frac{1}{N \times H \times W} \sum_{k=1}^{N} \sum_{i=1}^{H} \sum_{j=1}^{W} X \tag{10}$$

Pearson coefficient Pearson coefficient measures the correlation between the prediction and the ground truth. We first flatten the prediction and ground truth, and compute the metric as follows:

$$\rho_{\tilde{X},X} = \frac{\text{cov}(\tilde{X}, X)}{\sigma_{\tilde{X}} \sigma_X} \tag{11}$$

G.2. Results summary

Table 10 and 11 summarize the global forecasting results of ClimaX and the baselines for all target variables and at all lead times. In addition to IFS and the two CNN-based baselines in the main text, we include FourCastNet (Pathak et al., 2022), PanguWeather (Bi et al., 2022), and GraphCast (Lam et al., 2022) for comprehensiveness. We want to emphasize that the results obtained by these methods are not comparable with ClimaX, as they were trained on ERA5 at $0.25°$, a much higher resolution compared to $5.625°$ and $1.40625°$ data used to train ClimaX. In Appendix C.3, we had a discussion on how the performance of ClimaX scales favorably with respect to data resolution. We hope this summary will provide future works with an easier comparison with existing baselines.

In spite of being trained on much lower resolutions, ClimaX outperforms FourCastNet in forecasting Z500, T850, and U10 at lead times from 3 days and beyond, in terms of both RMSE and ACC. For T2m, ClimaX achieves better results at horizons longer than 3 days. PanguWeather performs better than ClimaX on most of the tasks, but the gap between the two methods shrinks and becomes negligible as the lead time increases. ClimaX even outperforms PanguWeather in predicting U10 at 7 days lead times. This is because ClimaX is finetuned to perform direct prediction, which mitigates error accumulation for long horizon prediction. GraphCast achieves the lowest RMSE among all methods, but performs worse in terms of ACC compared to ClimaX and PanguWeather.

Table 10. RMSE on global forecasting for different target variables at different lead times. Lower is better.

VARIABLE	LEAD TIME	ClimaX		FCN[a]	PW[b]	GC[c]	HRES	IFS		ResNet	UNet
	[hr.]	5.625°	1.40625°	0.25°	0.25°	0.25°	0.1	5.625°	1.40625°	5.625°	5.625°
Z500	6	62.73	49.67	37.52	15.40	16.46	24.66	26.93	26.96	47.00	53.66
[m^2/s^2]	24	96.19	72.76	81.31	42.23	38.77	45.90	51.01	50.96	86.60	132.65
	72	244.08	201.00	251.96	133.12	125.78	146.37	152.15	152.20	305.22	458.84
	120	440.40	392.00	483.44	295.63	271.65	316.79	331.45	331.38	614.20	721.83
	168	599.43	566.00	680.00	504.90	466.53	535.93	549.01	548.96	806.59	819.39
	336	790.26	788.43	nan	nan	nan	nan	1011.72	1011.56	835.55	866.40
	720	815.25	817.52	nan	nan	nan	nan	nan	nan	858.98	880.34
T2m	6	0.95	1.11	0.72	0.59	0.50	0.35	0.97	0.97	0.76	0.77
[K]	24	1.10	1.19	0.95	0.72	0.62	0.66	1.02	1.02	0.91	1.11
	72	1.43	1.47	1.38	1.05	0.94	1.06	1.30	1.30	1.70	1.91
	120	1.83	1.83	1.99	1.53	1.36	1.52	1.72	1.71	2.22	2.49
	168	2.18	2.17	2.54	2.06	1.88	2.06	2.24	2.23	2.66	2.66
	336	2.61	2.67	nan	nan	nan	nan	3.31	3.30	2.86	2.79
	720	2.67	2.74	nan	nan	nan	nan	nan	nan	2.86	2.81
T850	6	0.88	0.84	0.52	0.42	0.28	0.33	0.69	0.69	0.70	0.80
[K]	24	1.11	1.02	0.81	0.72	0.58	0.70	0.87	0.87	1.26	1.25
	72	1.59	1.46	1.55	1.13	1.02	1.27	1.34	1.34	1.90	2.39
	120	2.23	2.08	2.47	1.78	1.63	1.96	2.01	2.01	2.86	3.23
	168	2.77	2.66	3.30	2.60	2.41	2.78	2.82	2.82	3.51	3.50
	336	3.40	3.41	nan	nan	nan	nan	4.43	4.43	3.65	3.65
	720	3.47	3.49	nan	nan	nan	nan	nan	nan	3.69	3.73
U10	6	1.08	1.04	0.55	0.46	0.37	0.58	0.80	0.79	0.86	1.02
[m/s]	24	1.41	1.31	0.99	0.90	0.80	1.15	1.11	1.11	1.27	1.68
	72	2.18	2.02	2.24	1.60	1.47	1.98	1.92	1.92	2.78	3.17
	120	2.94	2.79	3.41	2.52	2.36	2.95	2.89	2.89	3.63	3.93
	168	3.43	3.35	4.18	3.46	3.25	3.87	3.81	3.81	4.15	4.08
	336	3.91	3.92	nan	nan	nan	nan	5.24	5.23	4.23	4.16
	720	3.96	3.97	nan	nan	nan	nan	nan	nan	4.29	4.22

[a] FourCastNet (Pathak et al., 2022)
[b] PanguWeather (Bi et al., 2022)
[c] GraphCast (Lam et al., 2022)

Table 11. ACC on global forecasting for different target variables at different lead times. Higher is better.

VARIABLE	LEAD TIME	ClimaX		FCN[a]	PW[b]	GC[c]	HRES	IFS		ResNet	UNet
	[hr.]	5.625°	1.40625°	0.25°	0.25°	0.25°	0.1	5.625°	1.40625°	5.625°	5.625°
Z500	6	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
	24	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.99
	72	0.97	0.98	0.97	0.99	0.99	0.98	0.99	0.99	0.95	0.89
	120	0.90	0.92	0.89	0.96	0.94	0.92	0.95	0.95	0.79	0.69
	168	0.80	0.82	0.76	0.87	0.83	0.78	0.87	0.87	0.57	0.57
	336	0.59	0.59	nan	nan	nan	nan	0.55	0.55	0.53	0.51
	720	0.55	0.55	nan	nan	nan	nan	nan	nan	0.49	0.49
T2m	6	0.98	0.98	0.99	0.99	0.98	0.99	0.99	0.99	0.99	0.99
	24	0.98	0.97	0.98	0.99	0.98	0.98	0.99	0.99	0.98	0.98
	72	0.96	0.96	0.96	0.98	0.95	0.94	0.98	0.98	0.94	0.93
	120	0.94	0.94	0.92	0.95	0.90	0.88	0.96	0.96	0.90	0.88
	168	0.91	0.91	0.87	0.92	0.81	0.77	0.93	0.93	0.86	0.86
	336	0.86	0.85	nan	nan	nan	nan	0.85	0.85	0.83	0.84
	720	0.85	0.84	nan	nan	nan	nan	nan	nan	0.83	0.83
T850	6	0.98	0.99	0.99	1.00	1.00	0.99	0.99	0.99	0.99	0.99
	24	0.98	0.98	0.98	0.99	0.99	0.98	0.99	0.99	0.97	0.97
	72	0.95	0.96	0.95	0.98	0.96	0.93	0.97	0.97	0.92	0.88
	120	0.89	0.91	0.87	0.94	0.89	0.84	0.93	0.94	0.82	0.75
	168	0.82	0.84	0.77	0.87	0.75	0.68	0.87	0.87	0.68	0.69
	336	0.71	0.71	nan	nan	nan	nan	0.68	0.69	0.66	0.66
	720	0.69	0.68	nan	nan	nan	nan	nan	nan	0.64	0.64
U10	6	0.97	0.97	0.99	0.99	0.99	0.99	0.98	0.98	0.98	0.97
	24	0.94	0.95	0.97	0.97	0.98	0.96	0.97	0.97	0.95	0.91
	72	0.85	0.87	0.85	0.92	0.93	0.88	0.89	0.89	0.74	0.65
	120	0.70	0.74	0.64	0.80	0.82	0.74	0.76	0.76	0.52	0.37
	168	0.56	0.59	0.45	0.63	0.64	0.55	0.58	0.58	0.28	0.28
	336	0.33	0.32	nan	nan	nan	nan	0.21	0.21	0.19	0.22
	720	0.29	0.28	nan	nan	nan	nan	nan	nan	0.17	0.21

[a] FourCastNet (Pathak et al., 2022)
[b] PanguWeather (Bi et al., 2022)
[c] GraphCast (Lam et al., 2022)

H. Qualitative evaluation

We qualitatively evaluate the performance of CliMax on global forecasting tasks for all target variables and at all lead times. In each figure, the first column is the initial condition of the target variable, which serves as the input, the second column is the ground truth of the target variable at a particular lead time, the third column is the prediction of ClimaX, and the last column is the bias, which is the difference between the prediction and the ground truth.

H.1. Nowcasting

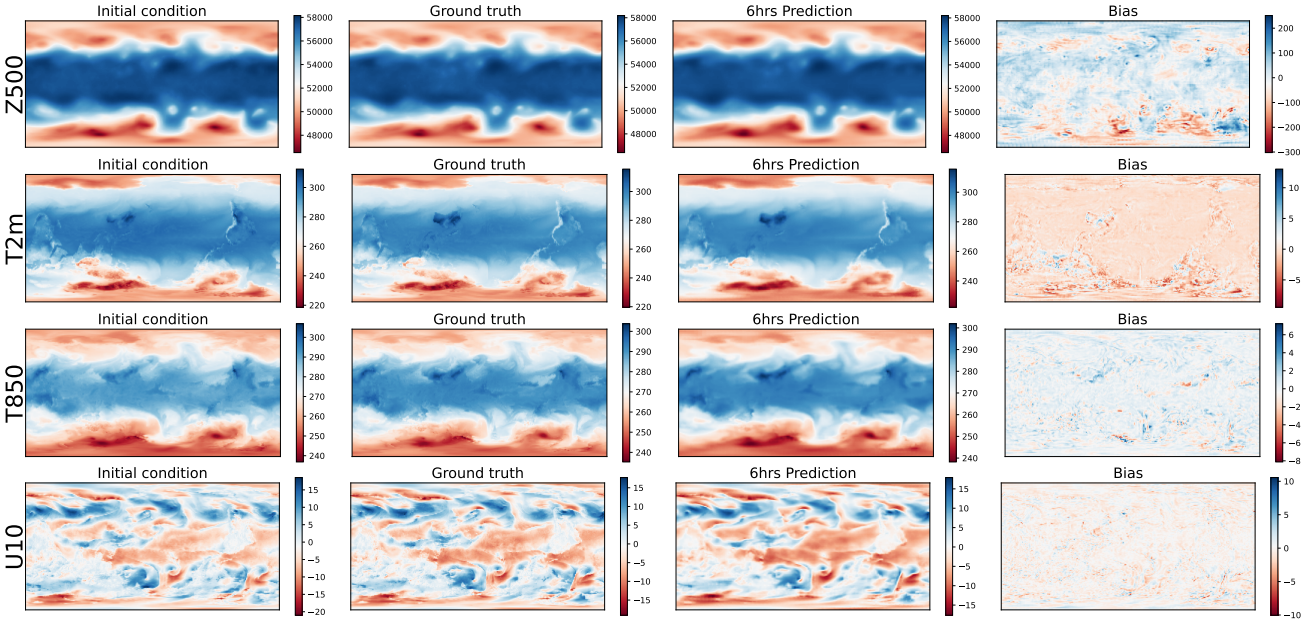

Figure 11. Example forecasts from ClimaX at 6-hour lead time compared to ground truth ERA5.

H.2. Short and medium-range weather forecasting

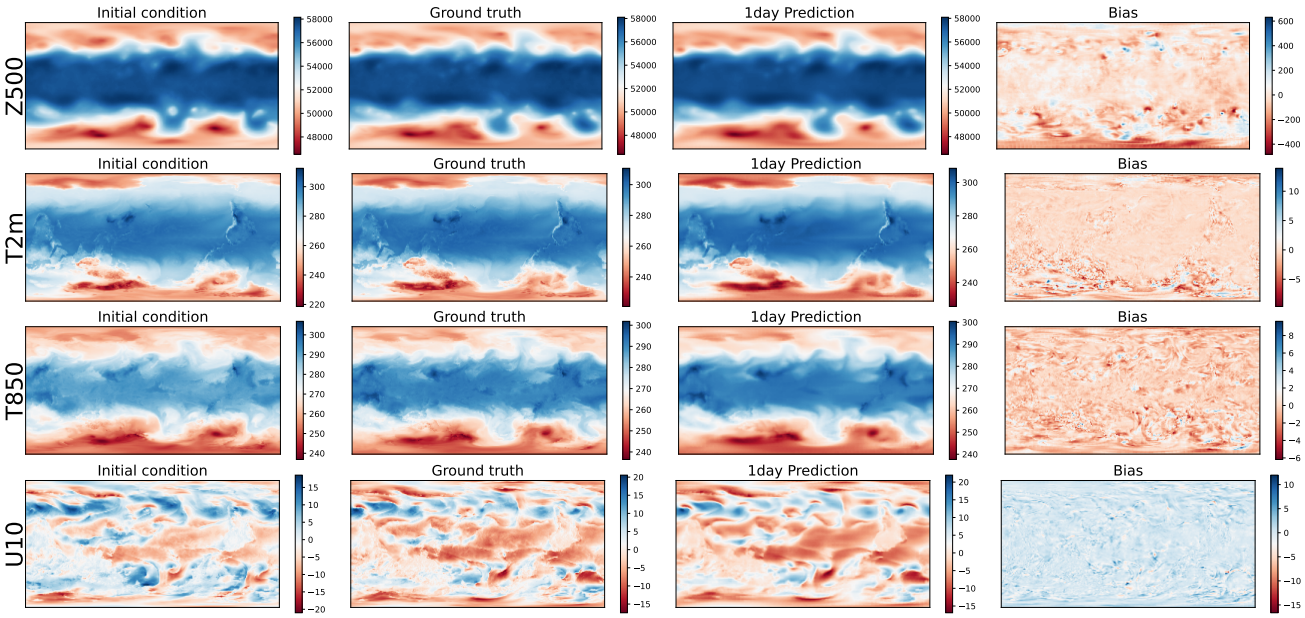

Figure 12. Example forecasts from ClimaX at 1-day lead time compared to ground truth ERA5.

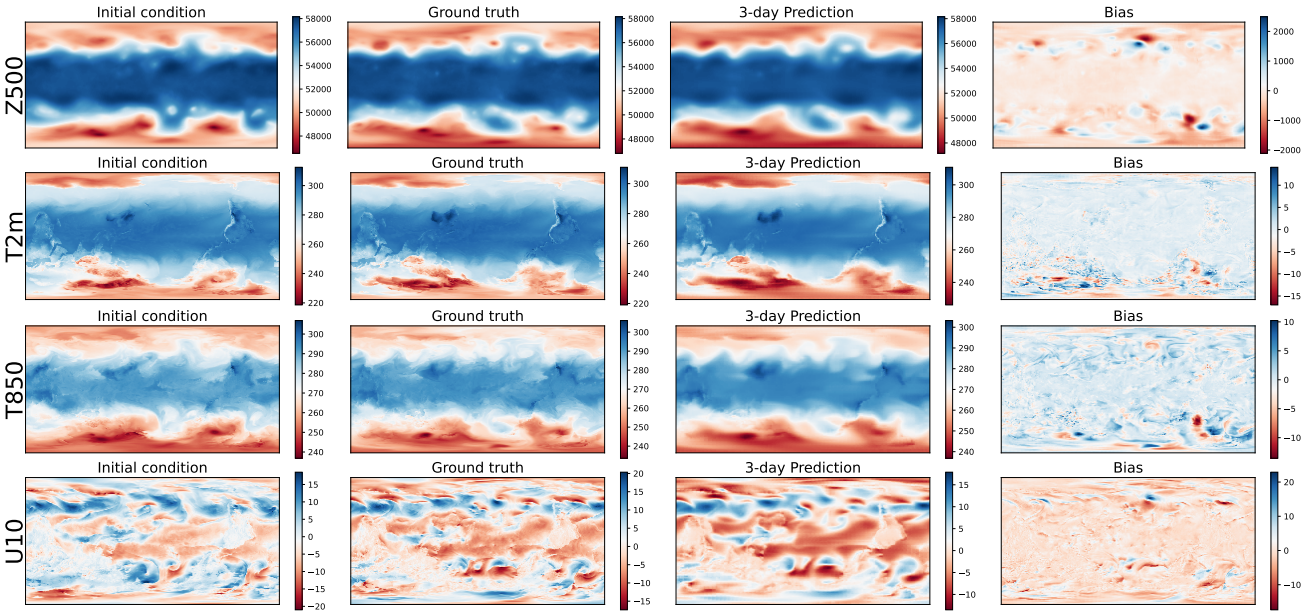

Figure 13. Example forecasts from ClimaX at 3-day lead time compared to ground truth ERA5.

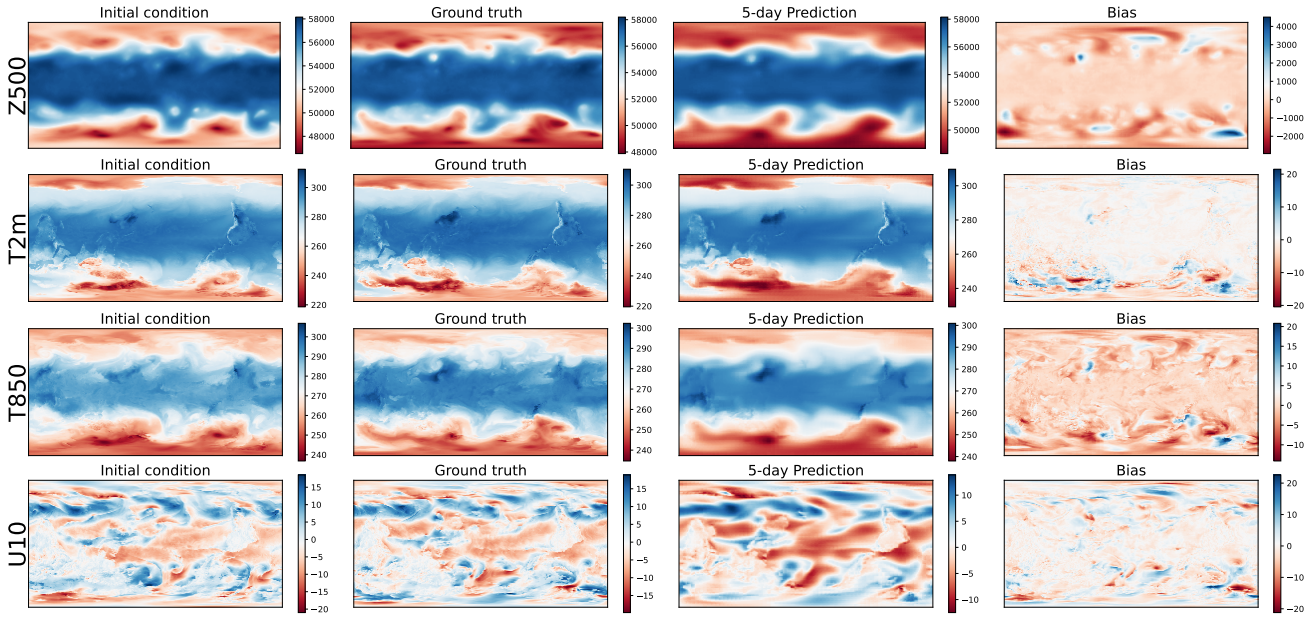

Figure 14. Example forecasts from ClimaX at 5-day lead time compared to ground truth ERA5.

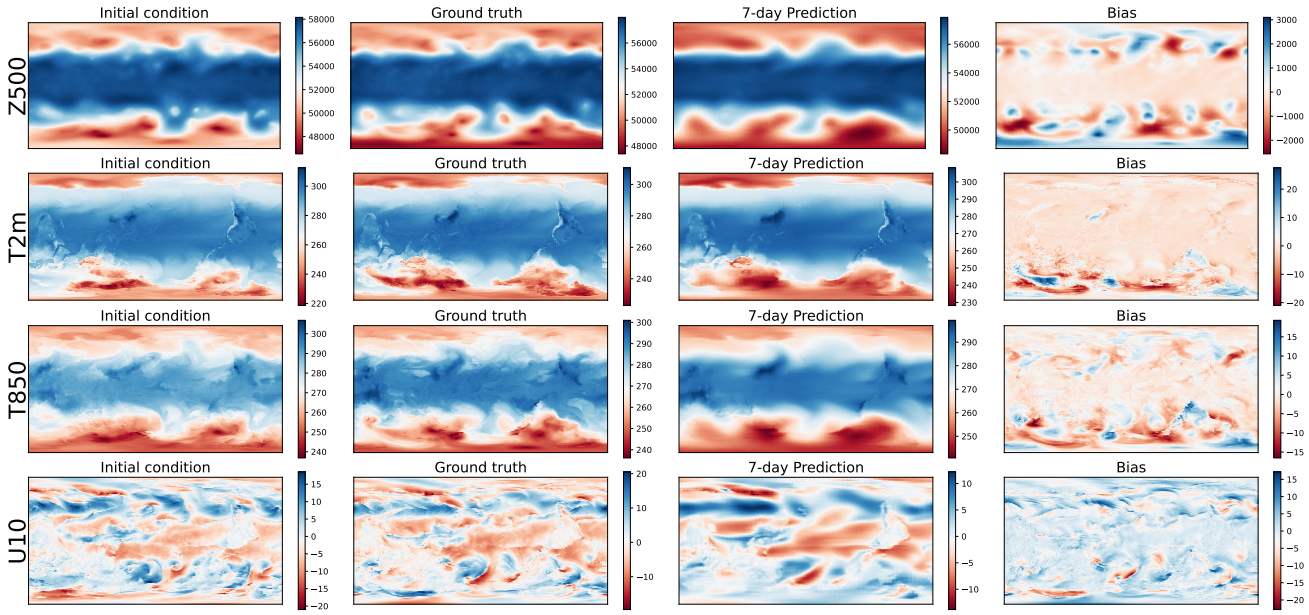

Figure 15. Example forecasts from ClimaX at 7-day lead time compared to ground truth ERA5.

H.3. Longer horizon instantaneous forecasting

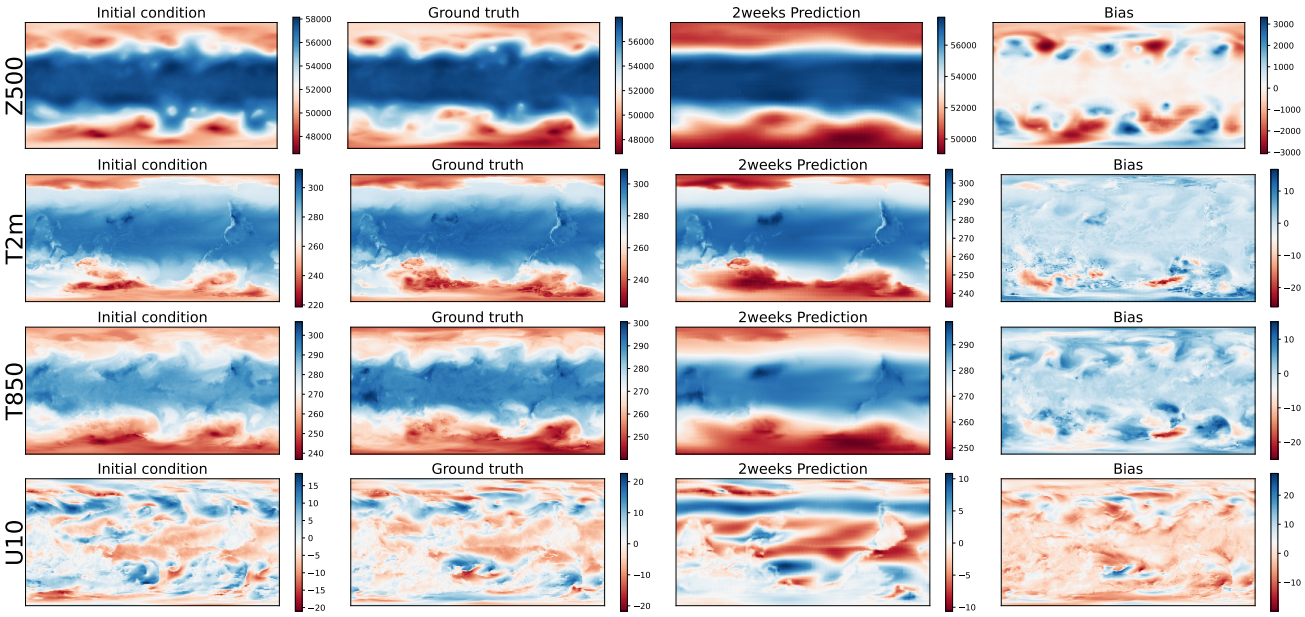

Figure 16. Example forecasts from ClimaX at 2-week lead time compared to ground truth ERA5.

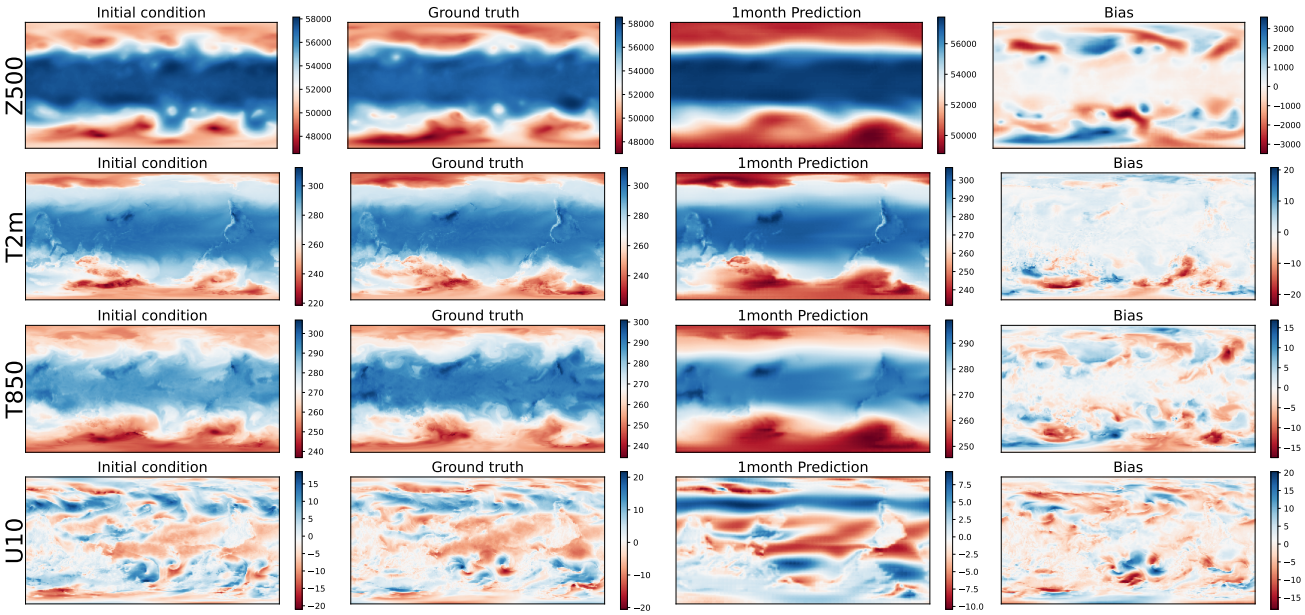

Figure 17. Example forecasts from ClimaX at 1-month lead time compared to ground truth ERA5.

H.4. Climate model downscaling

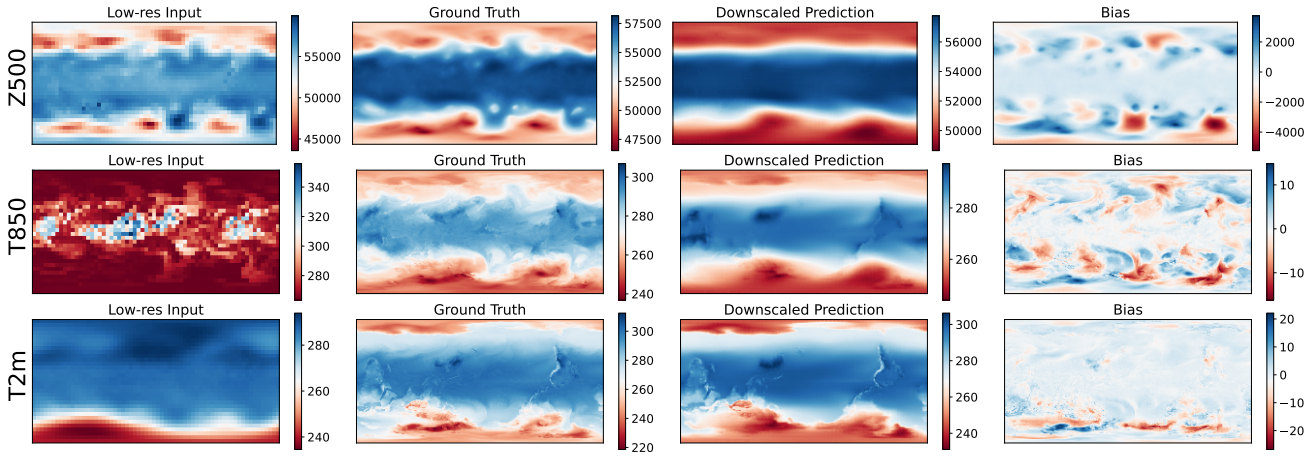

Figure 18. Example visualizations of downscaled prediction of key variables by ClimaX.