nature

https://doi.org/10.1038/s41586-025-08897-0

Accelerated Article Preview

End-to-end data-driven weather prediction

Received: 10 July 2024

Accepted: 12 March 2025

Accelerated Article Preview

Cite this article as: Allen, A. et al. End-to-end data-driven weather prediction. *Nature* https://doi.org/10.1038/s41586-025-08897-0 (2025)

Anna Allen, Stratis Markou, Will Tebbutt, James Requeima, Wessel P. Bruinsma, Tom R. Andersson, Michael Herzog, Nicholas D. Lane, Matthew Chantry, J. Scott Hosking & Richard E. Turner

End-to-end data-driven weather prediction

Anna Allen[1,11†], Stratis Markou[2,11†], Will Tebbutt[2,9], James Requeima[4], Wessel P. Bruinsma[5], Tom R. Andersson[8,10], Michael Herzog[6], Nicholas D. Lane[1], Matthew Chantry[7], J. Scott Hosking[3,8] and Richard E. Turner[2,3†]

[1]Department of Computer Science and Technology, University of Cambridge, Cambridge, UK
[2]Department of Engineering, University of Cambridge, Cambridge, UK
[3]The Alan Turing Institute, London, UK
[4]Vector Institute, University of Toronto, Ontario, Toronto, Canada
[5]Microsoft Research AI for Science, Cambridge, UK
[6]Department of Geography, University of Cambridge, Cambridge, UK
[7]European Centre for Medium-Range Weather Forecasts, Reading, UK
[8]British Antarctic Survey, Cambridge, UK
[9]Present address: The Alan Turing Institute, London, UK
[10]Present Adress: Google DeepMind, London, UK

[11]These authors contributed equally: Anna Allen, Stratis Markou
†Corresponding author. Email: av555@cam.ac.uk, em626@cam.ac.uk, ret26@cam.ac.uk.

Abstract

Weather prediction is critical for a range of human activities including transportation, agriculture and industry, as well as the safety of the general public. Machine learning is transforming numerical weather prediction (NWP) by replacing the numerical solver with neural networks, improving the speed and accuracy of the forecasting component of the prediction pipeline [1,2,3,4,5,6]. However, current models rely on numerical systems at initialisation and to produce local forecasts, limiting their achievable gains. Here we show that a single machine learning model can replace the entire NWP pipeline. Aardvark Weather, an end-to-end data-driven weather prediction system, ingests observations and produces global gridded forecasts and local station forecasts. The global forecasts outperform an operational NWP baseline for multiple variables and lead times. The local station forecasts are skillful up to ten days lead time, competing with a post-processed global NWP baseline and a state-of-the-art end-to-end forecasting system with input from human forecasters. End-to-end tuning further improves the accuracy of local forecasts. Our results show that skillful forecasting is possible without relying on NWP at deployment time, which will enable the full speed and accuracy benefits of data-driven models to be realised. We believe Aardvark Weather will be the starting point for a new generation of end-to-end models that will reduce computational costs by orders of magnitude, and enable rapid, affordable creation of customised models for a range of end-users.

Introduction

Numerical weather prediction (NWP) systems are vital for creating weather forecasts required by emergency agencies, transport providers, agriculture, energy providers and the general public. Since the first numerical forecasts were produced in the 1950s, which required 24 hours to compute a single-day single-variable forecast on a 700 km grid [7], NWP systems have undergone a remarkable transformation. Modern systems predict a wide range of variables at up to 15 days lead time, the theoretical limit of medium-range weather forecasting predictability [8]. These systems consist of an intricate series of models of different components of the Earth's atmosphere, building on decades of research in Earth observation, data assimilation, fluid dynamics and statistical post-processing and requiring purpose-built supercomputers to run.

Generating a modern weather forecast begins with the acquisition of observations from a multitude of sources, including remote sensing instruments, in-situ observations, radar systems, radiosondes and aircraft data [19]. Some of these data are processed to generate derived products such as atmospheric motion vectors and surface winds. Raw data and resulting processed products are fed into a *data assimilation* system which combines these with an initial guess from the previous forecast to generate a global approximation of the current state of the atmosphere. This approximation is then used as an initial state for a *forecasting* system that integrates equations of fluid mechanics and thermodynamics to output predictions at future lead times. Finally, the resulting predictions from the forecasting system are used for *downstream tasks*, for example to generate local forecasts. This step may consist of statistical post-processing and running further higher resolution regional NWP models. Each stage of this pipeline consists of multiple numerical models chained together, resulting in an intricate workflow[20] that is challenging to iterate on and improve and requires purpose-built supercomputers to run. This motivates the development of fast, lightweight, and customisable alternatives.

With end-to-end machine learning revolutionising multiple fields by replacing complex human designed workflows, it has been suggested that a data-driven model may one day replace the entire NWP pipeline [21]. This will be transformational for weather prediction, reducing computational costs, removing bias from inflexible aspects of NWP systems, and enabling fast prototyping and optimisation for specific tasks. However, this has not been attempted to date, with studies focusing on applying machine learning to the easiest components of the pipeline. For example, machine learning models have been shown to outperform their operational state-of-the-art counterparts for replacing the numerical solver in the forecasting component [1,2,4,5,6,22], deriving variables from raw satellite data in pre-processing [23,24,25], and post-processing forecast data in the downstream stages [26,27]. Work on replacing the most challenging component, the assimilation system, remains at the stage of developing initial prototypes [3,28,29,30,31,32,33]. The vision of an end-to-end data-driven solution therefore remains aspirational, with conventional NWP systems essential for all forms of operational forecasting.

In a recent article assessing the prospect of end-to-end deep learning weather prediction the verdict was that "a number of fundamental breakthroughs are needed before this goal comes into reach"[21]. Here we report that these breakthroughs are happening earlier than expected. We present Aardvark Weather, the first end-to-end data-driven weather forecasting system capable of generating predictions with no input from conventional NWP by instead learning a mapping from raw input observations to output forecasts. This allows Aardvark to tackle the complete weather prediction pipeline whilst being entirely independent from NWP products at prediction time, relying solely on observation data to generate forecasts. We demonstrate that, using an order of magnitude fewer observations than those available to operational baselines and orders of magnitude less compute, Aardvark is capable of producing forecasts on a global 1.50^{o} grid that achieve lower root mean squared error (RMSE) than operational NWP systems across multiple variables and lead times. Furthermore, we demonstrate that this system provides local forecasts that achieve lower errors than post-processed NWP and a full end-to-end operational forecasting system for multiple lead times, and can be optimised end-to-end to maximise performance over variables and regions of interest.

Results

Aardvark Weather

Aardvark Weather is a deep learning model which provides forecasts of eastward wind, northward wind, specific humidity, geopotential and temperature at 200, 500, 700 and 850hPa pressure levels, 10-metre eastward wind, 10-metre northward wind, 2-metre temperature and mean sea level pressure on a dense global grid, as well as station forecasts for 2-metre temperature and 10-metre wind speed. Aardvark consists of

three modules, and is designed to leverage high-quality reanalysis data during training while being entirely independent from NWP products at deployment time. Figure 1 (bottom) illustrates the operation of Aardvark, outlining the function of each of its three modules.

First, an *encoder* module takes in observational data from multiple sources, both on-the-grid and off-the-grid, and produces a gridded initial state. On-the-grid observations are data modalities on a regular grid, while off-the-grid modalities are available at a set of longitude-latitude locations. To achieve this we leverage recent advances from deep learning [34] in handling off-the-grid and missing data. This approach to state estimation differs from the data assimilation (DA) systems used in conventional NWP pipelines. Conventional DA systems use a recurrent update in which the previous forecast is adjusted in light of new observations, similar to the Kalman filter recursions in a Markov model. In principle, DA accumulates information from observations across all past time steps, however in practice it has been estimated that the effective window size is as little as four days[35]. Due to the complexities of training recurrent neural networks including the need for a spin up period and gradient instabilities [36], we therefore opt for a non-recurrent approach.

Once the initial atmospheric state has been estimated, it is used as input to a *processor* module, which produces a gridded forecast at a lead time of 24 hours. Forecasts at subsequent lead times are produced by autoregressively feeding the predictions of the processor module back to it as an input, similar to existing approaches in data-driven weather forecasting [1,6]. Finally, task-specific *decoder* modules ingest these forecasts and produce local predictions. While in this work we consider a decoder for a single downstream task, producing local station forecasts, this system is suitable for use with multiple separate decoders for different tasks. Together, the encoder, processor and decoder modules form a neural process [34], a machine learning system which naturally handles off-the-grid and missing data. A vision transformer (VIT) [37] forms the backbone of the encoder and processor modules, while decoder modules are implemented as a lightweight convolutional architecture. The full set of inputs and outputs for the modules is detailed in Extended Data Table 1.

A key challenge in designing machine learning systems for observational atmospheric data is that the record for many instruments is relatively short, limiting the data available for training. The modular design of Aardvark (figure 1) addresses this issue by enabling pre-training using high-fidelity historical reanalysis data before fine-tuning on the scarcer observational data. Specifically, we train the system in a way that mimics how it will be deployed. We start by pre-training the encoder module using raw observations as input and reanalysis data as targets. We note that an advantage of this machine learning approach is that the model can learn to correct for biases in the input observations during training, therefore no bias correction step is performed on the input data. We also pre-train the processor using reanalysis data for both inputs and targets, and then fine-tune on the output of the state-estimation module. In the processor module, inputs and outputs are both on a regular 1.50^{o} grid to match the reanalysis training data. We next train the decoder using the output of the processor as input and raw data as targets. This procedure ensures there is no mismatch between the training and deployment of the system. Finally, we fine-tune the encoder, processor and decoder modules jointly, to optimise the entire model for a specific variable and region. For all modules we train on data prior to 2018, and hold out 2018 and 2019 as test and validation years respectively.

Input variables

Accurately estimating the state of the atmosphere requires inputs from a variety of observation sources. Input variables are selected to capture dynamics both at the Earth's surface as well as at multiple different levels through the atmosphere. In-situ observations are taken from weather stations and ships at surface level, and radiosondes at upper levels. As coverage from these instruments is largely confined to the surface, as well as geographically skewed and sparse, remote sensing instruments provide a crucial complementary global data source. Motivated by gains observed in operational NWP systems[38, 39, 40], we select four primary sources of satellite data: scatterometer data to provide information about surface wind over the ocean, multi-spectral (≈10 channels) microwave and infrared sounders and hyper-spectral (≈10^5 channels) infrared sounders to provide information on upper atmosphere temperature and humidity profiles, and geostationary infrared sounder data to provide an instantaneous snapshot of the state of the atmosphere. These observations are taken with different time windows ranging from one to 24 hours prior to lead-time zero. In contrast to operational medium-range NWP systems, observations are only included in the input if they are taken prior to lead time zero[41]. Figure 1 (top) shows an example of a single time slice of input data to Aardvark for in-situ and remote sensing sources, with full details in Extended Data Table 2. These atmospheric observations are augmented by several temporal and orographic variables. We note that Aardvark only ingests approximately 8% of the observations[1] available to conventional NWP systems[42], more than an order of magnitude fewer input data.

Evaluation: global forecasting

For global gridded forecasts we compare Aardvark to four baselines. The simplest of these, persistence and hourly climatology, assess whether a forecasting system is skillful. A more challenging comparison is to the two most widely used deterministic operational global NWP systems: the Integrated Forecast System (IFS) in its high resolution (HRES) configuration from the European Centre for Medium Range Weather Forecasting (ECFMWF), and the Global Forecast System (GFS) from the National Centres for Environmental Prediction. While HRES typically outperforms GFS on global metrics, operational centres often use a selection of different models, including GFS, to create their local forecasts, so we include it our comparison. For each variable, pressure level, and lead time, we report the latitude weighted root mean squared error (RMSE), a common metric for assessing the performance of deterministic forecasting systems[43]. For all baselines we take ERA5 reanalysis as ground truth. This choice is made as this is standard practice for evaluation machine learning NWP models. We note that, while in the present day, HRES analysis is of higher quality than ERA5 reanalysis, since ERA5 was developed using cycle Cy41r2[18] which remained operational until 2017, therefore the discrepancies between the two are limited for the test year of 2018.

Figure 2 shows latitude weighted RMSE performance compared to the baselines for eight headline variables. Here Aardvark matches or outperforms GFS across most lead times, with the only exception being geopotential at 500hPa. In addition, for most variables, Aardvark approaches the performance of HRES. Overall, Aardvark's errors are larger at higher atmospheric levels and shorter lead times compared to the operational baselines. This is possibly due to the higher concentration of observations close to the surface. For longer lead times, a by-product of fine-tuning to minimise errors at future lead times (see Methods) is that forecasts tend to become spectrally blurred. This phenomenon is commonly observed in data-driven weather forecasting systems [1,6,44]. A full display of the latitude weighted RMSE of Aardvark across all variables and levels can be found in Figure 1 in the supplementary information. Further insights can be drawn from inspecting the power spectra, anomaly correlation coefficients and activities of Aardvark's forecasts, shown in Figure 2, Figure 3 and Figure 4 of the supplementary information. This analysis suggests that while forecast blurring plays a role, Aardvark produces skillful forecasts and maintains meaningful signals even at longer lead times.

Figure 3 shows an example of gridded global predictions at lead times of zero, one, two and four days for 10-metre eastward wind. Aardvark successfully captures large-scale features of the atmospheric state, both in the mid-latitudes and the tropics. Many details are well represented, for example the formation of a tropical cyclone in the Southern Indian Ocean closely matches that in the ERA5 reanalysis data. This example hints at the potential of Aardvark for forecasting mesoscale high-impact weather events. Although some spectral blurring of the higher spatial frequencies is evident, these results are of remarkably high fidelity given the limited resolution and range of observations provided to the model. A comprehensive set of spatial plots across all variables is provided in Figures 6 to 29 in the supplementary information.

Encoder module ablation

A central innovation of the Aardvark Weather system is the estimation of an initial state from disparate data sources using the encoder module. With the volume and diversity of observational modalities available, an important question to ask is: *which observational sources are most important for estimating each atmospheric variable, and how does each affect predictive performance?* To investigate this we conduct an ablation experiment quantifying the significance of each observational source in our encoder module. We remove different observational sources from the set of encoder inputs, retrain the encoder with this reduced set, and evaluate it on the same test set as our original configuration, marked "ALL" (Figure 4). For example, the rows "no in-situ" and "no satellites" correspond to removing in-situ data and all satellite data respectively from the "ALL" configuration. We report the fractional increase in the latitude-weighted RMSE relative to the "ALL" configuration across all atmospheric variables for the initial condition generated at $t = 0$.

These results demonstrate that remote sensing data are of crucial importance in constraining the initial atmospheric state. Removing these data (no satellite in Figure 4), and training with in-situ observations only, leads to large skill reductions across all variables. Within different satellite modalities, the low earth orbit (LEO) sounder data are the most important. For example, removing these sounder modalities (no LEO) results in larger skill deterioration compared to, for example, removing scatterometer data (no ASCAT) or geostationary satellite data (no GEO). In-situ observations are most important for the surface variables, however they also play a surprisingly large role in predicting geopotential, particularly at lower levels. These results indicate that for future improvement of this system and development of other end-to-end data-driven systems, LEO sounder data are the most important source to include, with in-situ data providing an important complementary source to improve surface variable and geopotential forecasts. We provide full details of this experiment in Supplementary Information A.

Evaluation: station forecasting

In the next stage of the weather prediction pipeline, global gridded forecasts are used as input to downstream models to produce a variety of products for end users. One such category of products is producing local forecasts. We focus on applying Aardvark Weather to predict 2-metre atmospheric temperature and 10-metre wind speed at off-the-grid station locations. Accurate local predictions of temperature are vital for protection of public health during heatwaves and cold waves, in addition to agriculture and other use cases. Similarly, wind speed forecasts have a variety of end users for example in wind energy, marine forecasting and fire weather forecasting. We note that modules for any desired downstream task could be substituted for this station forecasting module.

There are significant differences in how agencies in different countries produce forecasts for end users. In well-resourced countries, station forecasts are produced using global models followed by higher resolution

regional models out to a few days lead time and statistical post-processing[46]. In contrast, in less well-resourced areas, while agencies have access to global products they often do not have access to comparable infrastructure to run high-resolution local NWP or post-process forecasts to a comparable degree[47]. With these considerations in mind, we report Aardvark's performance across all stations globally, but also break it down over four regions of particular interest: the contiguous United States (CONUS), Europe, West Africa and the Pacific (Figure 5; k). The US and most European countries run both local NWP for shorter lead times, as well as sophisticated post-processing of both global and local products. In contrast, West Africa and the Pacific are regions in which many centres are less well equipped. Although some agencies in these regions run sophisticated NWP pipelines, others utilise solely raw HRES forecasts and issue operational forecasts for very short lead times [47]. We compare Aardvark against per-station persistence and climatology, as well as against two challenging baselines: downscaled HRES and a full operational end-to-end baseline, the National Digital Forecast Database (NDFD) from the National Weather Service[46]. For a detailed description of baselines see Methods.

Figure 5 shows the mean absolute error (MAE) performance of Aardvark, reported by variable and region. Globally Aardvark generates skillful forecasts for both temperature and wind speed up to a lead time of 10 days, performing competitively with station-corrected HRES. For temperature, Aardvark is competitive with station-corrected HRES over both CONUS and Europe. In addition, remarkably, Aardvark matches the performance of the full operational NDFD baseline over CONUS. For lower resource areas in West Africa and the Pacific Aardvark outperforms station-corrected HRES at all lead times. For 10-metre wind speed, Aardvark has higher errors than station-corrected HRES over CONUS, and significantly outperforms the NDFD baseline. Over Europe, Aardvark has similar errors with station-corrected HRES up to four days lead time, and outperforms it thereafter. Finally, Aardvark generally outperforms station-corrected HRES over West Africa, while performing slightly worse over the Pacific. In addition to these results, we compare Aardvark's performance to HRES for a set of held out stations globally, demonstrating competitive performance on both variables (see figure 5 in the Supplementary Information).

End-to-end tuning

End users of NWP products typically have a particular region and set of applications that are of interest. A powerful capability of Aardvark is the ability to tune the entire pipeline end-to-end to directly optimise for any desired quantity and region of interest. Optimising the performance for a particular end-user product would be challenging and expensive in a conventional NWP system. To explore this capability, we fine-tune Aardvark to optimise predictions of 2-metre temperature and 10-metre wind speed at one day lead time globally and for each of the four regions. Although here we focus on only these two variables, this is a powerful paradigm able to be applied anywhere there is uncertainty in the reanalysis training data, for example clouds and precipitation.

We observe that fine-tuning Aardvark yields improvements both globally, as well in the specific regions of CONUS, Europe, West Africa and the Pacific (figure 5; bottom). For temperature, fine-tuning Aardvark results in large reductions in MAE of 6% over Europe, West Africa, the Pacific, and globally, and an improvement of 3% over CONUS. For 10-metre wind speed, small but statistically significant improvements of 1-2% are observed for all regions except the Pacific. To put these improvements into context, the last cycle update of the IFS improved surface variable scores in the range of two to six percent and took over a year of development by a large team of scientists.

Discussion

We have introduced Aardvark Weather, an end-to-end weather forecasting system which is the first data-driven system to tackle the entire NWP pipeline. Aardvark provides accurate forecasts that are orders of magnitude quicker to generate than existing systems, without any reliance on NWP products at deployment time. Generating a full forecast from observational data takes approximately one second on four NVIDIA A100 GPUs, compared to the approximately 1,000 node-hours required by HRES to perform data assimilation and forecasting [48] alone, before accounting for downstream local models and processing. In downstream tasks generating station forecasts of 2-metre temperature and 10-metre wind speed, Aardvark shows strong performance against operational NWP systems. Learning an end-to-end model offers the additional capability to optimise the system to maximise performance over an arbitrary variable or region of interest, opening the door for the creation of inexpensive, individually tailored models for any region globally, in an automated and streamlined fashion.

End-to-end forecasting has significant potential for real world impact. Compared to conventional NWP systems, machine learning systems are not only faster and computationally cheaper, but are also significantly easier to improve and maintain. In conventional NWP a new module, for example for a novel parameterisation or micro-physics scheme, may take a team considerable time to build and integrate into the model. End-to-end data-driven systems such as Aardvark elegantly bypass this issue using a single model in place of this complex pipeline. The simplicity of this system both makes it both easier to deploy and maintain for users already running NWP, and also opens the potential for wider access to running bespoke NWP in areas of the developing world where agencies often lack the resources and expertise to run conventional systems. There is also significant potential in the demonstrated ability to fine-tune bespoke models to maximise predictive skill for specific regions and variables. This capability is of interest to many end users in areas as diverse as agriculture, renewable energy, insurance and finance.

To envisage how an end-to-end data-driven model such as Aardvark could be deployed operationally, it is necessary to consider the limitations of the current model and a concrete set of steps required to turn it into a fully-fledged NWP system. As with all current AI-NWP systems[1, 6], Aardvark does not yet run at the resolution of IFS. Further work is required both to increase grid resolution and to produce forecast ensembles through, e.g. diffusion [2]. Other limitations centre around the use of observations. Further observational modalities will likely increase forecast skill. It is also important to consider how data from new instruments for which there are no training data available can be usefully integrated into the system. This could be accomplished by, for example, training on simulated data[49]. A further consideration is dealing with observation drift and other changes in data over time, which could be mitigated by regularly finetuning all modules with the most recent few months of data to adapt to changes in instrument characteristics.

The results presented in this study only scratch the surface of the potential of Aardvark Weather and end-to-end data-driven weather forecasting systems more broadly. Further capabilities could also be added by extending Aardvark to support multiple other forecast variables, both in its gridded forecasts as well as via its decoder module. For example, Aardvark could support a diverse range of decoder modules, to provide different types of end user forecasts such as hurricane, flood, severe convection, fire weather and other extreme weather warnings. A further exciting avenue for future research would be utilizing end-to-end systems at longer lead times to generate seasonal forecast products. Furthermore, additional observational modalities would allow for modelling of other components of the earth system, such as atmospheric chemistry for air quality forecasts and ocean parameters for marine forecasts. We envision that Aardvark Weather will be the first of a new generation of end-to-end weather forecasting systems tackling these diverse tasks.

References

1. Lam, R. *et al.* Learning skillful medium-range global weather forecasting. *Science* **382,** 1416–1421. eprint: https://www. science.org/doi/pdf/10.1126/science.adi2336. https://www.science.org/doi/abs/10.1126/science.adi2336 (2023).
2. Price, I. *et al.* GenCast: Diffusion-based ensemble forecasting for medium-range weather. *arXiv preprint arXiv:2312.15796* (2023).
3. Xu, X. *et al.* Fuxi-DA: A Generalized Deep Learning Data Assimilation Framework for Assimilating Satellite Observa- tions. *arXiv preprint arXiv:2404.08522* (2024).
4. Chen, K. *et al.* Fengwu: Pushing the skillful global medium-range weather forecast beyond 10 days lead. *arXiv preprint arXiv:2304.02948* (2023).
5. Keisler, R. Forecasting global weather with graph neural networks. *arXiv preprint arXiv:2202.07575* (2022).
6. Bi, K. *et al.* Accurate medium-range global weather forecasting with 3D neural networks. *Nature* **619,** 533–538 (2023).
7. Lynch, P. The origins of computer weather prediction and climate modeling. *Journal of computational physics* **227,** 3431– 3444 (2008).
8. Zhang, F. *et al.* What is the predictability limit of midlatitude weather? *Journal of the Atmospheric Sciences* **76,** 1077– 1091 (2019).
9. EUMETSAT. *Metop ASCAT Level 1B SZF Product* https://navigator. eumetsat. int/ product/ EO: EUM: DAT: METOP: ASCSZF1B. Accessed: 2024-10-20. 2024.
10. Zou, C.-Z., Wang, W. & Program, N. C. *NOAA Fundamental Climate Data Record (FCDR) of AMSU-A Level 1c Bright- ness Temperature, Version 1.0* https://doi.org/10.7289/V5X63JT2. NOAA National Climatic Data Center, Accessed: 22/10/2024. 2013.
11. Ferraro, R. R., Meng, H. & Program, N. C. *NOAA Climate Data Record (CDR) of Advanced Microwave Sounding Unit (AMSU)-B, Version 1.0* https://doi.org/10.7289/V500004W. NOAA National Climatic Data Center. 2016.
12. EUMETSAT. *HIRS Level 1C Fundamental Data Record Release 1 - Multimission - Global* http://doi.org/10.15770/ EUM_SEC_CLM_0026. 2022.
13. EUMETSAT. *IASI Principal Components Scores Fundamental Data Record Release 1 - Metop-A and -B* http://doi.org/ 10.15770/EUM_SEC_CLM_0084. 2022.
14. NOAA National Centers for Environmental Information (NCEI). *Gridded Geostationary Brightness Temperature Data* https://www.ncei.noaa.gov/products/gridded-geostationary-brightness-temperature. Accessed: 2024-10-20. 2024.
15. UK Met Office. *HadISD: Met Office Hadley Centre Integrated Surface Dataset* https://www.metoffice.gov.uk/hadobs/ hadisd/. Accessed: 2024-10-20. 2024.
16. NOAA National Centers for Environmental Information (NCEI). *International Comprehensive Ocean-Atmosphere Data Set (ICOADS)* https://icoads.noaa.gov. Accessed: 2024-10-20. 2024.
17. NOAA National Centers for Environmental Information (NCEI). *Integrated Global Radiosonde Archive (IGRA)* https://www.ncei.noaa.gov/products/weather-balloon/integrated-global-radiosonde-archive. Accessed: 2024-10-20. 2024.
18. Hersbach, H. *et al.* The ERA5 global reanalysis. *Quarterly Journal of the Royal Meteorological Society* **146,** 1999–2049 (2020).
19. ECMWF. in. 1 (ECMWF, 2023).
20. Dueben, P. D. & Bauer, P. Challenges and design choices for global weather and climate models based on machine learning. *Geoscientific Model Development* **11,** 3999–4009 (2018).

21. Schultz, M. G. *et al.* Can deep learning beat numerical weather prediction? *Philosophical Transactions of the Royal Society A* **379,** 20200097 (2021).
22. Chen, L. *et al.* FuXi: a cascade machine learning forecasting system for 15-day global weather forecast. *npj Climate and Atmospheric Science* **6,** 190 (2023).
23. Shao, W., Zhou, Y., Zhang, Q. & Jiang, X. Machine Learning-Based Wind Direction Retrieval From Quad-Polarized Gaofen-3 SAR Images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **17,** 808–816 (2023).
24. Yan, X. *et al.* A deep learning approach to improve the retrieval of temperature and humidity profiles from a ground-based microwave radiometer. *IEEE transactions on geoscience and remote sensing* **58,** 8427–8437 (2020).
25. Zhang, Z., Dong, X., Liu, L. & He, J. Retrieval of barometric pressure from satellite passive microwave observations over the oceans. *Journal of Geophysical Research: Oceans* **123,** 4360–4372 (2018).
26. Kirkwood, C., Economou, T., Odbert, H. & Pugeault, N. A framework for probabilistic weather forecast post-processing across models and lead times using machine learning. *Philosophical Transactions of the Royal Society A* **379,** 20200099 (2021).
27. Grönquist, P. *et al.* Deep learning for post-processing ensemble weather forecasts. *Philosophical Transactions of the Royal Society A* **379,** 20200092 (2021).
28. Chen, K. *et al.* Towards an end-to-end artificial intelligence driven global weather forecasting system. *arXiv preprint arXiv:2312.12462* (2023).
29. Huang, L., Gianinazzi, L., Yu, Y., Dueben, P. D. & Hoefler, T. DiffDA: a diffusion model for weather-scale data assimilation. *arXiv preprint arXiv:2401.05932* (2024).
30. McNally, A. *et al.* Data driven weather forecasts trained and initialised directly from observations. *arXiv preprint arXiv:2407.15586* (2024).
31. Manshausen, P. *et al.* Generative Data Assimilation of Sparse Weather Station Observations at Kilometer Scales. *arXiv preprint arXiv:2406.16947* (2024).
32. Keller, J. D. & Potthast, R. AI-based data assimilation: Learning the functional of analysis estimation. *arXiv preprint arXiv:2406.00390* (2024).
33. Cheng, S., Min, J., Liu, C. & Arcucci, R. TorchDA: A Python package for performing data assimilation with deep learning forward and transformation functions. *Computer Physics Communications* **306,** 109359 (2025).
34. Gordon, J. *et al. Convolutional Conditional Neural Processes* in *International Conference on Learning Representations* (2019).
35. Berre, L. Simulation and diagnosis of observation, model and background error contributions in data assimilation cycling. *Quarterly Journal of the Royal Meteorological Society* **145,** 597–608. (2019).
36. Pascanu, R., Mikolov, T. & Bengio, Y. *On the difficulty of training recurrent neural networks* in *Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28* (JMLR.org, Atlanta, GA, USA, 2013), III–1310–III–1318.
37. Dosovitskiy, A. *et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale* in *International Conference on Learning Representations* (2020).
38. Laloyaux, P., Thépaut, J.-N. & Dee, D. Impact of scatterometer surface wind data in the ECMWF coupled assimilation system. *Monthly Weather Review* **144,** 1203–1217 (2016).
39. Isaksen, L. & Janssen, P. A. Impact of ERS scatterometer winds in ECMWF's assimilation system. *Quarterly Journal of the Royal Meteorological Society: A journal of the atmospheric sciences, applied meteorology and physical oceanography* **130,** 1793–1814 (2004).

40. Eyre, J. *et al.* Assimilation of satellite data in numerical weather prediction. Part II: Recent years. *Quarterly Journal of the Royal Meteorological Society* **148,** 521–556 (2022).
41. European Centre for Medium-Range Weather Forecasts. Continuous long-window data assimilation. *ECMWF Newsletter.* https://www.ecmwf.int/en/newsletter/163/news/continuous-long- window- data- assimilation, Accessed: 2024-07-07 (2020).
42. Healy, S. *et al. Methods for assessing the impact of current and future components of the global observing system* Reading, Apr. 2024. .
43. Rasp, S. *et al.* Weatherbench 2: A benchmark for the next generation of data-driven global weather models. *arXiv preprint arXiv:2308.15560* (2023).
44. European Centre for Medium-Range Weather Forecasts. *A new ML model in the ECMWF web charts* https://www.ecmwf.int/en/ about/media-centre/aifs-blog/2023/new-ml-model-ecmwf-web-charts, Accessed: 2024-06-26. 2023.
45. National Centers for Environmental Prediction, National Weather Service, NOAA, U.S. Department of Commerce. *NCEP GFS 0.25 Degree Global Forecast Grids Historical Archive* Boulder CO, 2015. https://rda.ucar.edu/datasets/dsd084001/.
46. Glahn, H. R. & Ruth, D. P. The new digital forecast database of the National Weather Service. *Bulletin of the American Meteorological Society* **84,** 195–202 (2003).
47. WMO. *WMO Integrated Processing and Prediction System (WIPPS) Dashboard* Accessed: 2024-07-05. 2024. https://community.wmo.int/en/activity-areas/wmo-integrated-processing-and-prediction-system-wipps.
48. Buizza, R. *et al.* The development and evaluation process followed at ECMWF to upgrade the Integrated Forecasting System (IFS). eng. *ECMWF Technical Memoranda.* https://www.ecmwf.int/node/18658 (Oct. 2018).
49. Kaspar, M., Osorio, J. D. M. & Bock, J. *Sim2real transfer for reinforcement learning without dynamics randomization* in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (2020), 4383–4388.

Figure 1. Illustration of the data and operation of Aardvark Weather. *(a) Illustration of the different data sources leveraged in Aardvark. The input data consist of observations from remote sensing instruments (top row of a) which we pre-grid before passing to the model, as well as in-situ observations from land and marine observation platforms and radiosondes (bottom row of a). Each of these data modalities contain several observational variables, of which we select a subset here for the purposes of illustration. Here, we show remote sensing data [9,10,11,12,13,14] after performing our gridding step, and raw in-situ data [15,16,17]. Note that the colours in all six plots are meant for illustration purposes. The remote sensing data also include a range of meta-data about the measurements, omitted here for simplicity. White areas indicate regions of missing data which must be handled by the encoder module of Aardvark. (b) Illustration of of Aardvark at deployment time. First, an encoder module uses raw observations as input to estimate the initial state of the atmosphere across key variables at* $t = 0$*. Next, a processor module ingests the estimated state to produce a forecast at the next lead time* $t = \delta t$*. Forecasts at subsequent lead times are produced autoregressively. Finally, a decoder module is applied to the on-the-grid states to produce off-the-grid predictions. The modular design of Aardvark allows for pre-training on large, high-quality ERA5 reanalysis data[18]. In this figure, the data displayed are the training data used in to train each module of Aardvark, from the aforementioned sources. Circular line drawings generated using DALLE.*

Figure 2. Gridded global forecast performance for selected variables. *Latitude-weighted RMSE using ERA5 [18] reanalysis data as the ground truth, on the held out test year (2018), for the four surface variables: 2-metre temperature (a; T2M), 10-metre eastward wind (b; U10), 10-metre northward wind (c; V10) and mean sea level pressure (d; MSLP) and four headline upper-atmosphere variables: temperature at 850hPa (e; T850), eastward wind at 700hPa (f; U700) specific humidity at 700hPa (g; Q700) and geopotential at 500hPa (h; Z500), as a function of lead time* t*. At lead time* $t = 0$*, Aardvark predicts the initial atmospheric state from from observational data alone. The error at* $t = 0$ *corresponds to the error in the initial state. Note that HRES has non-zero error at* $t = 0$*, as it is compared to ERA5 reanalysis. The HRES forecasts [43] we use have been conservatively re-gridded to prevent aliasing, and we performed the same operation on the GFS forecasts [45]. We report the mean performance of each system together with 98% confidence intervals in our estimate of the mean performance.*

Figure 3. Example of Aardvark's gridded forecasts for the U10 wind component. *Plots of the initial condition (a-c) and subsequent forecasts (d-l) for 10-metre eastward wind (U10), showing Aardvark's prediction (a, d, g, j), the ERA5 ground truth[18] (b, e, h, k), and the difference between the two (c, f, i, l). Lead time* $t = 0$ *corresponds 00:00 on the 11th January 2018. Aardvark correctly predicts large-scale features for this variable, and correctly predicts the formation and positioning of the tropical cyclone Berguitta (highlighted in the magenta boxes), which reached peak intensity on the 15th of January 2018 off the coast of Madagascar. We emphasise that the model makes these predictions entirely from raw observations [9,10,11,12,13,14,15,16,17], without any NWP products as input.*

Figure 4. Encoder ablation experiments quantifying the impact of each data modality. *Results of ablation experiments comparing the LW-RMSE of the encoder trained with all data sources, both remote sensing [9, 10, 11, 12, 13, 14] as well as in-situ sources[15, 16, 17] (ALL) to other encoder configurations including: removing the scatterometer data (no ASCAT), removing the geostationary sounder data (no GEO), removing all in-situ data (no in-situ), removing all low Earth orbit sounder data (no sounder), or removing all satellite data (no satellite). We report the fraction of increase in LW-RMSE of each configuration relative to ALL.*

Figure 5. Station forecast performance and end-to-end fine-tuning improvements. *Top (a-j): Results for station forecasting for the held-out test set (2018) of HadISD data[15]. Here, Aardvark makes predictions at spatial locations observed during training, on temporally held out data, but it can also generate predictions at any arbitrary station location. We compare Aardvark's forecasts to two state-of-the-art NWP baselines, the National Digital Forecast Database (NDFD) [46] for CONUS. We also compare against a version of HRES [43] that we post-process using a separate scale and bias term for each station. We report the mean performance of each system together with 98% confidence intervals in our estimate of the mean performance. Bottom (k-m): Improvements from fine-tuning. Here, we compare the predictions of Aardvark for lead* $t = 1$ *day to those of its end-to-end fine-tuned counterpart for 2-metre temperature (T2M) and 10-metre wind speed (WS). We report the mean % improvement in each variable by region (k) with 98% confidence intervals. "Global" includes all stations (black and coloured). We emphasize Aardvark produces its predictions entirely from raw remote sensing [9,10,11,12,13,14] and in-situ [15,16,17] observations without any NWP products as input during test time.*

Methods

Datasets: state estimation inputs

We select multiple remote sensing and in-situ observations for input to the atmospheric state estimation module. To ensure that no NWP system is required for operational deployment of Aardvark, we select only data that are available at either level 1B or 1C processing level[50]. Level 1B satellite data refers to calibrated and geolocated data, meaning the raw sensor measurements have been processed to correct for sensor and instrument biases but are still in the form of physical measurements, while Level 1C satellite data are further processed to include radiometric and geometric corrections, making it ready for analysis with accurate geolocation and radiance values[50]. Other requirements for inclusion of datasets are that they are available from 2007-2020 and are available in near real time to facilitate anticipated operational deployment. Where available for remote sensing products, we utilise fundamental climate data records, where data from earlier generation sensors are homogenised to match the characteristics of current sensors, creating a consistent data record for training. Extended Data Table 1, provides a summary of all datasets that are used as inputs to the encoder module, including the type of instrument, orbit and platform (if applicable), as well as the data provider and data selection window that we use. We note that for satellite instruments in low Earth orbit it is necessary to include a longer window of observations to attain full global coverage. In contrast station observations for all locations are available at $t = 0$. Adding extra data would therefore be useful but is not necessary to achieve global coverage. As the data record is relatively short and over-fitting is a concern, we made the decision to limit the data to the shortest window possible whilst retaining global coverage.

In-situ observations are included from land stations, marine platforms and radiosondes. In-situ land station observations measuring surface temperature (8719 stations), pressure (8016 stations), wind (8721 stations) and dew point temperature (8617 stations) at six hourly intervals are taken from the HadISD dataset[51, 52], provided by the UK Met Office. Marine in-situ observations are taken from the International Comprehensive Ocean-Atmosphere Data Set (ICOADS)[53] provided by The National Oceanic and Atmospheric Administration. This dataset consists of observations from ships and buoys globally, from which five variables are included, namely 2-metre air temperature, 10-metre northward and eastward wind, sea surface temperature and mean sea level pressure. As observations are not taken precisely on the hour, all observations from $t = -1$ hours to $t = 0$ are included in the input. Upper atmosphere observations of humidity, wind, geopotential and temperature are taken from The Integrated Global Radiosonde Archive (IGRA)[54], provided by the National Centers for Environmental Information. This dataset consists of radiosonde observations at 1375 sites globally. Each record contains observations at multiple levels, of which we select observations at the surface and 200, 500, 700 and 850hPa pressure levels. All profiles retrieved within the past six hours, from $t = -6$ hours to $t = 0$, are included in the input.

As in-situ observations are limited in geographic coverage, remote sensing observations are included from scatterometers and microwave and infrared sounders. Input data from satellites are ingested in the form of level one granules each containing a six minute slice of observations or orbits. Although in principle the Aardvark Weather system can handle these data in their raw form, for simplicity data were first transferred to a regular one degree grid by nearest neighbour interpolation where the most recent observation is maintained in cases where multiple observations are available for the same gridpoint.

Several scatterometers are currently operational worldwide, of which we use the Advanced Scatterometer (ASCAT)[55] instrument aboard MetOp-A, B and C. Data for this instrument are provided by the European Organisation for the Exploitation of Meteorological Satellites (EUMETSAT). ASCAT provides a triplet of three measurements of backscatter (σ_0) from which operational centres retrieve the wind speed and direction, using a geophysical model function which solves for the two unknowns as a function of the σ_0 triplet together

with satellite metadata[56]. In contrast to this approach we opt to simply include the raw σ_0 values together with the metadata as channels to the encoder module, eliminating the complexity of the retrieval process. As all MetOp satellites are in Low Earth Orbit (LEO), with a revisit time of approximately 24 hours, the input to the state estimation module comprises of the latest ASCAT observations available within the grid box from any of the three platforms on a regular 1.50° longitude-latitude grid from $t = -1$ days to $t = 0$ days.

In operational NWP, temperature and humidity profiles in the upper atmosphere are retrieved using infrared and microwave sounder instruments[57]. For this purpose we include the Advanced Microwave Sounding Units A and B (AMSUA & AMSUB), and the Microwave Humidity Sounder (MHS) instruments for microwave observations and the High Resolution Infrared Radiation Sounder (HIRS/4) for infrared observations. Together, these instruments comprise the Advanced TIROS Operational Vertical Sounder (ATOVS) system used operationally to retrieve temperature and moisture profiles[58]. Data for these instruments is provided by The National Centers for Environmental Information (NCEI). Observations for AMSU-A, AMSU-B, MHS and HIRS are taken from the NOAA 15 through 19, Aqua and MetOp-A satellites. In operational NWP systems, both retrieved profiles and raw radiances are assimilated. Similar to ASCAT, profiles of the target variable are retrieved using a geophysical model function taking in the raw radiances and satellite metadata and solving for the desired observational profiles. We again opt to input the raw radiances together with the satellite metadata directly into the state estimation module without relying on higher level retrievals. As for ASCAT, the dataset consists of the latest observations from $t = -1$ to $t = 0$ days, taken within a grid box of a regular 1.50° longitude-latitude grid.

We augment the ATOVS observations with data from a hyperspectral infrared sounder, the Infrared Atmospheric Sounding Interferometer (IASI)[59]. Data for this instrument is provided by The National Centers for Environmental Information (NCEI). IASI captures data at a much higher spectral resolution than HIRS/4, with a total of 8461 channels across three bands. To limit input data volume, we take the leading 15 principal components across these channels, a technique demonstrated to lead to limited performance degradation in operational NWP systems. We note that data from IASI is available from October 2007 as opposed to January 2007 for the rest of the training set.

While platforms carrying scatterometer and passive microwave sounder instruments in LEO provide high-resolution observations, they have the disadvantage of lower temporal resolution. In contrast, geostationary satellites provide very high temporal resolution though with more limited instrumentation. As the available channels on geostationary satellites vary geographically and with time, we opt to use a composite product, the Gridded Satellite dataset (GridSat)[60], which provides homogenised retrievals of infrared and vapour window channels over standard geostationary platforms. Data for this instrument is provided by the National Climatic Data Center. For this data source we include the image taken at $t = 0$.

To account for diurnal, seasonal and longer term variations in the data, we include temporal information as input both to the encoder and forecasting modules. These channels consist of $\sin\left(\frac{2\pi d}{366}\right)$, $\cos\left(\frac{2\pi d}{366}\right)$, $\sin\left(\frac{2\pi h}{24}\right)$, and $\cos\left(\frac{2\pi h}{24}\right)$ where d is the day of year and h the hour of day. The absolute year is also included to account for any changes in data characteristics over the training record. In order to account for the effects of orography on the weather system, we include several sources of orographic information taken from the ERA5 dataset[18] as static fields. These data are provided by the European Center for Medium Range Weather Forecasting. These are the geopotential at surface level, angle of sub-grid scale orography, anisotropy of sub-grid scale orography, slope of sub-grid scale orography and standard deviation of orography.

Datasets: pre-training

The modular structure of Aardvark leverages ERA5 reanalysis data during the training phase to increase the length of the data record available. ERA5, or the Fifth Generation of the European Centre for Medium-Range Weather Forecasts (ECMWF) Reanalysis[18], is a state-of-the-art global atmospheric reanalysis dataset. It provides comprehensive information on various meteorological parameters such as temperature, humidity, wind, and geopotential, covering the period from 1940 to present. These data are provided by the European Center for Medium Range Weather Forecasting. From this we elect to train on data from 1979 onwards, coinciding with the beginning of widely available remote sensing observations which significantly improves the quality of the atmospheric reanalysis product.

Datasets: baselines

For the global gridded forecast experiments we compare the performance of Aardvark against four baselines: persistence, climatology, HRES and GFS. Persistence and climatology provide simple baselines for assessing whether a forecasting system is skillful. In persistence forecasting, it is assumed that the weather will remain unchanged from $t = 0$ at all future lead times. For the climatology baseline we utilise the climatology product from WeatherBench 2 [43]. Here, the predicted state is obtained by taking the mean value of all ERA5 observations from 1990-2017 for a given day of the year and hour using a sliding window of length 61 days.

The IFS and GFS are the two most widely used global operational NWP systems. As the focus of this study is on deterministic forecasting, we choose to compare our results to the HRES and GFS, deterministic runs at a resolution of 0.10◦ degrees and 0.25° degrees respectively. These constitute challenging baselines for comparison to Aardvark Weather which operates at a 1.50° resolution with just five vertical levels. For comparison to Aardvark, HRES and GFS outputs are conservatively regridded to 1.50° resolution. In particular, we use HRES forecast data and ERA5 target data as provided by WeatherBench 2 [43], in which both datasets have been coarsened to 1.50° resolution using first order conservative re-gridding [61]. This procedure reduces the effects of aliasing, ensuring that Aardvark does not get an unfair competitive advantage due to distortions in the power spectrum that would occur from naive sub-sampling. To ensure the GFS forecasts are compared fairly against Aardvark and HRES, we also apply conservative re-gridding to GFS. See Supplementary Information for further details on aliasing and its effects on signal spectra.

For station forecasts we consider four baselines. Persistence and climatology are calculated based on station observations. For 2-metre temperature we calculate daily climatology, and for 10-metre wind speed monthly. We further consider two more challenging baselines: station-corrected HRES and NDFD over the contiguous United States. As HRES is a gridded product, sub-grid scale processes are not resolved. We therefore learn a bias correction individually for each station on the 2007-2017 training set, and use this to correct the station forecasts on the 2018 test set. NDFD is produced by the National Weather Service in the United States of America, and is a state-of-the art local forecasting system [62]. Forecasts in the NDFD are created from an ensemble of over 30 models [63], including the IFS and GFS together with high resolution regional models at shorter lead times. The data from these systems is the shown to human forecasters at different National Weather Service (NWS) offices who create the final forecast. Our station forecasts are taken as the nearest gridbox forecast from the final NDFD forecast which is at approximately 2km resolution. NDFD therefore constitutes an extremely challenging baseline, capturing the full complexity of operational forecasting pipeline.

Evaluation metrics

For the global gridded forecasting experiments we compare models on latitude weighted root mean squared error. Given arrays of gridded target forecasts y and gridded target predictions $\hat{y}$, the latitude weighted RMSE of variable v is calculated as

$$\text{LW-RMSE}(y, \hat{y}, v) = \frac{1}{B}\sum_{b=1}^{B}\sqrt{\frac{1}{HW}\sum_{h=1}^{H}\sum_{w=1}^{W}\alpha_h(y_{bhwv} - \hat{y}_{bhvw})^2} \quad (1)$$

where b indexes batch elements, v indexes atmospheric variables, h and w index latitude and longitude coordinates and α_h are the latitude weights, defined as

$$\alpha_h = \frac{\cos\theta_h}{\frac{1}{H}\sum_{h=1}^{H}\cos\theta_h} \quad (2)$$

where θ_h is the latitude along the latitude-wise index h, so that their average is equal to one. In machine learning, a (mini-)batch refers to a subset of the training dataset, typically used to compute a stochastic estimate of a model's parameter gradients when performing gradient-based optimisation. For the station forecasting experiments we compare methods on mean absolute error. Given arrays of station target temperatures y and predictions $\hat{y}$, the MAE is calculated as

$$\text{MAE}(y, \hat{y}) = \frac{1}{BN}\sum_{b=1}^{B}\sum_{n=1}^{N}|y_{bn} - \hat{y}_{bn}| \quad (3)$$

where b indexes batch elements, and n indexes the stations in the forecast.

Training objectives

Separate training objectives are utilised for each of the three modules. For all three modules, we normalise the targets by calculating the mean and standard deviation for each variable and level, aggregating across all grid points. We note that the encoder and processor modules, which involve multiple target variables, this normalisation has the effect of implicitly weighting the variables, due to the scaling applied during normalisation. For the encoder module we determine an additional weighting by first training the model with using a latitude weighted RMSE objective of the form

$$\text{SUM-LW-RMSE}(y, \hat{y}) = \frac{1}{V}\sum_{v=1}^{V}\text{LW-RMSE}(y, \hat{y}, v) \quad (4)$$

In this initial run, all variables are therefore weighted equally. Next, weights β_v are produced for each variable by taking the reciprocal of the latitude weighted RMSE for each variable multiplied by a factor of three to generate weights within the range of approximately 0 to 1. The training objective for the encoder uses these weights, giving the variable and latitude weighted RMSE

$$\text{VLW-RMSE}(y, \hat{y}) = \frac{1}{V}\sum_{v=1}^{V} \beta_v \, \text{LW-RMSE}(y, \hat{y}, v) \quad (5)$$

For the processor module the training objective is the SUM-LW-RMSE (eq. 4). However, the processor module is trained to predict residuals (see "Processor module" below). We found the implicit weighting that is applied via normalisation works well, and we did not further weight the variables individually. Finally, for the decoder module the training objective is the same as for evaluation, that is eq. (3).

Model architecture

Aardvark Weather is a neural process model[64]. Neural processes are a family of deep learning models that provide a flexible framework capable of learning with off-the-grid data, missing and sparse data and providing probabilistic predictions at arbitrary locations at test time. These characteristics are ideally suited to working with complex environmental data for example in climate downscaling and sensor placement [65, 66, 67, 68, 69].

Our specific architecture is a novel member of the neural process family combining SetConv layers developed for the convolutional conditional neural process[34], which handle off-the-grid and sparse data modalities and produce off-the-grid predictions, together with a vision transformer backbone currently used in state-of-the-art AI-NWP forecasting systems[70]. This provides scalability not currently attainable with standard transformer neural process models with attention based encoders[71], whilst still retaining the flexibility to handle diverse data modalities. Here we give details for the architectures of these modules, how they are trained and fine-tuned, and how they are deployed. In the discussion that follows, note that the encoder, processor and decoder modules all receive auxiliary channels such as temporal embeddings and orographic information as input. For simplicity, we suppress these channels in our exposition, but it should be understood that all three modules receive them as input. We provide a complete listing of all inputs and outputs to our models in Extended Data Table 2.

Encoder module

The encoder module $\mathcal{E}$ takes raw observations as input, and outputs a gridded estimate of the initial state of each variable for the processor module. Let $o_\tau = \{o_{\tau,1}, \ldots, o_{\tau,N}\}$ be the set of observations corresponding to time τ, where each $o_{\tau,n}$, corresponds to the observations and corresponding meta-data (e.g. viewing angle, solar elevation angle and observation time) of a single data modality. Each $o_{\tau,n} = (x_{\tau,n}, y_{\tau,n})$ consists of a set of observations $y_{\tau,n}$ and their corresponding longitude and latitude coordinates $x_{\tau,n}$. Each data modality is either on-the-grid or off-the-grid, and has a corresponding function ψ_n to transform $o_{\tau,n}$ into a gridded representation of fixed dimensionality. For gridded observations, ψ_n consists of the addition of a masking channel to distinguish missing data from observed data in the grid. For off-the-grid observations, each ψ_n consists of a SetConv layer[34] with a learnable length scale. The SetConv layer produces a gridded representation of the data, as well as an accompanying density channel which carries information about the presence or absence of data, to handle irregularly sampled observations. The regular gridded representations of the modalities are concatenated to give a single gridded representation of dimension $C \times H \times W$, where C is the number of resulting channels, H is the number of latitude points and W is the number of longitude points. This representation of the input data are fed into the backbone of the module, consisting of a vision transformer V_e with patch size three, eight transformer blocks and latent dimension 512. Embeddings for each patch use an MLP following prior work [37]. The encoder outputs the initial state estimate $\hat{s}_{\tau,0}$ at time τ

with dimension $24 \times W \times H$ where 24 is the number of variables modelled in the forecasting module. Putting this together, we have

$$\hat{s}_{\tau,0} = E(o_\tau) = V_e\left(\odot_{n=1}^{N} \psi_n(o_{\tau,n})\right) \tag{6}$$

where $\hat{s}_{\tau,0}$ is the estimated initial state corresponding to time τ and $\odot$ denotes concatenation. The encoder module is trained to predict ERA5 reanalysis targets using the VLW-RMSE (eq. 5) as its loss function. We train the module for 150 epochs using AdamW with early stopping and a cosine learning rate scheduler starting at an initial learning rate of 5×10^{-4} and decaying to zero at the final epoch.

Processor Module

The processor module P takes the initial state estimate $\hat{s}_{\tau,0}$ as input and outputs forecasts for lead-times of one to ten days. This module consists of ten separate vision transformers, $V_p^{(1)}, \ldots, V_p^{(10)}$, which are composed to produce gridded global forecasts at each of the ten lead times we consider. Here each $V_p^{(i)}$, is designed to provide a one day forecast conditioned on the forecast of $V_p^{(i-1)}$. This 24-hour timestep is a common configuration in AI-NWP models [71,72], and is used here to avoid inconsistencies in assimilation procedures at the 06:00 and 18:00 UTC runs of IFS which may disadvantage this baseline in the comparison[2], and for computational tractability. All vision transformers have a patch size 5, latent dimension 512 and 16 transformer blocks. To improve modelling of interactions between variables we add cross-attention between variables at the start of the network, as suggested by[73]. The processor is trained using a pre-training phase followed by a fine-tuning phase. Let $\hat{s}_{\tau,t}$ be the ERA5 state corresponding to time t and lead time τ. During pre-training, the first vision transformer, $V_p^{(1)}$, is trained to ingest $s_{\tau,0}$ as input and predict the residual $s_{\tau,1} - s_{\tau,0}$ using the SUM-LW-RMSE loss (eq. 4). We pre-train, $V_p^{(1)}$ for 100 epochs using AdamW with a cosine learning rate scheduler starting at an initial learning rate of 5×10^{-4} and decaying to zero at epoch 100. During the fine-tuning phase, we train each vision transformer $V_p^{(i)}$ to work with the estimated state produced by the previous transformer $V_p^{(i-1)}$ as follows. Recall that $\hat{s}_{\tau,0}$ is the estimated state produced by the encoder module. We start by training $V^{(1)}$ to predict $s_{\tau,1} - \hat{s}_{\tau,0}$ using the initial state $\hat{s}_{\tau,0}$ as input. Once $V_p^{(1)}$ has been fine-tuned, we compute $\hat{s}_{\tau,1} = \hat{s}_{\tau,0} + V^{(1)}(\hat{s}_{\tau,0})$, and initialise the network $V_p^{(2)}$ using the weights of $V_p^{(1)}$. We then fine-tune $V_p^{(2)}$ to predict $s_{\tau,2} - \hat{s}_{\tau,1}$ using $\hat{s}_{\tau,1}$ the previously estimated initial state as input. We proceed sequentially in this fashion, until all networks have been initialised and fine-tuned. We note that this procedure can be regarded as an instance of the *pushforward trick* [74]. At deployment time, we compose the transformers to obtain a forecast for the desired lead time, that is

$$s_{\tau,t} = P(s_{\tau,0}, t) = \tilde{V}_p^{(t)} \circ \ldots \circ \tilde{V}_p^{(1)}(s_{\tau,0}) \tag{7}$$

where $\tilde{V}_p^{(t)}(\cdot) = \cdot + V_p^{(t)}(\cdot)$ and $s_{\tau,0} = E(o_\tau)$ is the initial state produced by the encoder.

Decoder module

The final step in the forecasting pipeline is the decoder module. For each lead-time t, we train a lightweight convolutional station forecasting module D_t, which takes the gridded estimated state $s_{\tau,t}$, as well as target

longitude-latitude coordinates x and auxiliary orographic information as inputs, and produces predictions for the corresponding station temperature measurements $y_{\tau,t}$. Each D_t consists of a UNet architecture[75], followed be a SetConv layer which maps on-grid predictions to predictions at arbitrary station locations, followed by an MLP which incorporates the auxiliary orographic information, to produce local forecasts $\hat{y}_{\tau,t}$. The UNet consists of four encoder blocks (which consist of 2D convolutions, BatchNorm layers, ReLU activations and MaxPool operations) followed by four decoder blocks (which consist of transpose 2D convolutions, BatchNorm layers, ReLU activations and MaxPool operations). The encoder and decoder blocks have skip connections, and channel dimensions (16, 32, 64, 128, 64, 32, 16, 1). We train each D_t for 10 epochs, using AdamW, with a learning rate of 1×10^{-3} and the RMSE loss (eq. 3). To produce local forecasts at coordinates x, we compute

$$\hat{y}_{\tau,t} = D_t\left(s_{\tau,t}, x\right) \tag{8}$$

where $s_{\tau,t}$ is the global forecast defined in eq. (7).

End-to-end deployment

At deployment time, no ERA5 input is required to run the system. To obtain global forecasts, we compose the encoder and processor together, and compute

$$\hat{s}_{\tau,t} = P_t \circ E(o_\tau) \tag{9}$$

where $P_t(\cdot) = P(\cdot, t)$. If we want to produce local station forecasts, we compose the encoder, processor as well as decoder modules, and compute

$$\hat{y}_{\tau,t} = D_t(P_t \circ E(o_\tau), x) \tag{10}$$

Station forecasting baselines

We compare Aardvark against per-station persistence and climatology, as well as against two challenging baselines. The first of these is a downscaled version of HRES: for each station, we select the nearest gridpoint from the HRES 0.25° forecast, and learn an affine correction (a scale and a constant bias) on a per-station basis to correct for systematic biases, which is a common and highly effective downscaling method[76]. We note that further, region-specific, downscaling refinements are possible, e.g. using local nested NWP. These could potentially further improve the performance of NWP systems, so the station-corrected HRES results we present should not necessarily be interpreted as the state-of-the-art in downscaling performance, but rather as a strong and globally applicable baseline. Second, over CONUS, we also compare against a full operational end-to-end baseline, the National Digital Forecast Database (NDFD) from the National Weather Service. NDFD forecasts are an archive of data from National Weather Service offices produced by combining the output of multiple global and regional forecasting models, post-processing these and incorporating input from human forecasters[46].

End-to-end fine-tuning

In order to perform end-to-end fine-tuning, we compose the encoder together with the lead time $t = 1$ processor and decoder modules, producing local station forecasts for lead time $t = 1$ given by

$$\hat{y}_{\tau,1} = D_1(P_1 \circ E(o_\tau), x) \tag{11}$$

This composition produces a single machine learning model, whose inputs consist of all raw observational sources of the encoder module, and whose outputs consist of the predictions of the decoder module. We then fine-tune this composite mode, i.e. all three networks, jointly with either 2-metre temperature or 10-metre windspeed station observations $y_{t,1}$ as the only targets, using the RMSE loss. Specifically, the fine-tuning procedure consists of loading the pre-trained weights of the encoder, processor and decoder modules, and performing stochastic gradient descent on the parameters of the three modules E, P_1 and D, to minimize the RMSE loss between the station forecast $\hat{y}_{\tau,1}$ and its corresponding target $y_{t,1}$. We use AdamW and optimise all parameters of the modules for 25,000 gradient steps with a constant learning rate of 5×10^{-5} and early stopping, as described by the following procedure.

During training, we store checkpoints of our models, in order to perform region-based model selection during evaluation. Specifically, every 1,000 fine-tuning gradient steps, we store a copy of the model weights at that point in training, commonly referred to as a checkpoint. We then use the checkpoints to perform model selection based on performance on a held-out validation set. Specifically, we evaluate each of the model checkpoints generated during fine-tuning on the validation data on the data from each of the regions we consider, namely Global, CONUS, Europe, West Africa and the Pacific. For each region, we then select the best checkpoint, as measured by performance on the validation set for that region, and evaluate this on the test data corresponding to the given region.

Model size and training costs

All model training for this paper was performed on a single virtual machine with four NVIDIA A100 GPUs. The encoder module contains approximately 31 million parameters and requires 13 hours to train. The processor module contains approximately 54 million parameters, and requires 8 hours to train on ERA5 and 3 hours to fine-tune using the output of the encoder module as input. Each of the eleven decoder modules contains approximately 2 million parameters and takes approximately 30 minutes to train. End-to-end fine-tuning of the encoder, processor and decoder modules takes 2 hours. The total time to train the model is therefore approximately 100 GPU hours.

Data availability. The dataset to run Aardvark Weather will be made available at 10.57967/hf/4274 on the completion of peer review. All figures have been generated using a combination of the Latex tikz package and the matplotlib Python package[77]. All coastlines and borders drawn in the spatial plots in the main text and supplement of the paper come from the border and coastline functionality of the matplotlib package.

Code availability. The code used for training the models, the trained models themselves, example test data, and notebook examples for how to apply the models to make predictions is made available here via a Zenodo link DOI 10.5281/zenodo.13158382.

Methods references

50. *Earthdata, NASA* https://www.earthdata.nasa.gov/learn/earth-observation-data-basics/data-processing-levels. Accessed: 2024-10-26.
51. Dunn, R. J. *et al.* HadISD: A quality-controlled global synoptic report database for selected variables at long-term stations from 1973–2011. *Climate of the Past* 8, 1649–1679 (2012).
52. Dunn, R. J., Willett, K. M., Parker, D. E. & Mitchell, L. Expanding HadISD: Quality-controlled, sub-daily station data from 1931. *Geoscientific Instrumentation, Methods and Data Systems* 5, 473–491 (2016).
53. Freeman, E. *et al.* ICOADS Release 3.0: a major update to the historical marine climate record. *International Journal of Climatology* 37, 2211–2232 (2017).
54. Durre, I., Vose, R. S. & Wuertz, D. B. Overview of the integrated global radiosonde archive. *Journal of Climate* 19, 53–68 (2006).
55. Gelsthorpe, R., Schied, E. & Wilson, J. ASCAT-Metop's advanced scatterometer. *ESA bulletin* 102, 19–27 (2000).
56. Stoffelen, A., Verspeek, J. A., Vogelzang, J. & Verhoef, A. The CMOD7 geophysical model function for ASCAT and ERS wind retrievals. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 10, 2123–2134 (2017).
57. Rosenkranz, P. W. Retrieval of temperature and moisture profiles from AMSU-A and AMSU-B measurements. *IEEE Transactions on Geoscience and Remote Sensing* 39, 2429–2435 (2001).
58. Li, J. *et al.* Global soundings of the atmosphere from ATOVS measurements: The algorithm and validation. *Journal of Applied Meteorology and Climatology* 39, 1248–1268 (2000).
59. Blumstein, D. *et al.* IASI instrument: Technical overview and measured performances. *Infrared Spaceborne Remote Sensing XII* 5543, 196–207 (2004).
60. Knapp, K. R. & Wilkins, S. L. Gridded satellite (GridSat) GOES and CONUS data. *Earth System Science Data* 10, 1417– 1425 (2018).
61. Jones, P. W. First-and second-order conservative remapping schemes for grids in spherical coordinates. *Monthly Weather Review* 127, 2204–2210 (1999).
62. Service, N. W. *National Digital Forecast Database: Short Range Guidance for TAF Sites* 2024. https://www.weather.gov/ media/mdl/ndfd/pd01002001curr.pdf.
63. Service, N. W. *How Do We Use Models in Our Forecasting?* 2024. https://www.weather.gov/ilx/about_models.
64. Garnelo, M. *et al. Conditional neural processes* in *International conference on machine learning* (2018), 1704–1713.
65. Andersson, T. R. *et al.* Environmental sensor placement with convolutional Gaussian neural processes. *Environmental Data Science* **2,** e32 (2023).
66. Markou, S., Requeima, J., Bruinsma, W., Vaughan, A. & Turner, R. E. *Practical Conditional Neural Process Via Tractable Dependent Predictions* in *International Conference on Learning Representations* (2022).
67. Vaughan, A., Tebbutt, W., Hosking, J. S. & Turner, R. E. Convolutional conditional neural processes for local climate downscaling. *Geoscientific Model Development* **15,** 251–268 (2022).
68. Vaughan, A., Lane, N. D. & Herzog, M. *Multivariate climate downscaling with latent neural processes* in *Tackling Climate Change with Machine Learning ICML Workshop* (2021).
69. Bruinsma, W. *et al. Autoregressive Conditional Neural Processes* in *The Eleventh International Conference on Learning Representations* (2023).

70. Bodnar, C. *et al.* Aurora: A foundation model of the atmosphere. *arXiv preprint arXiv:2405.13063* (2024).
71. Nguyen, T. *et al.* Scaling transformer neural networks for skillful and reliable medium-range weather forecasting. *arXiv preprint arXiv:2312.03876* (2023).
72. Couairon, G., Lessig, C., Charantonis, A. & Monteleoni, C. ArchesWeather: An efficient AI weather forecasting model at 1.5 degree resolution. *arXiv preprint arXiv:2405.14527* (2024).
73. Nguyen, T., Brandstetter, J., Kapoor, A., Gupta, J. K. & Grover, A. Climax: A foundation model for weather and climate. *arXiv preprint arXiv:2301.10343* (2023).
74. Brandstetter, J., Worrall, D. & Welling, M. *Message Passing Neural PDE Solvers* 2023. arXiv: 2202.03376 [cs.LG].
75. Ronneberger, O., Fischer, P. & Brox, T. *U-Net: Convolutional Networks for Biomedical Image Segmentation* 2015. arXiv: 1505.04597 [cs.CV].
76. Bouallègue, Z. B. *et al.* Statistical modeling of 2-m temperature and 10-m wind speed forecast errors. *Monthly Weather Review* **151,** 897–911 (2023).
77. Hunter, J. D. Matplotlib: A 2D graphics environment. *Computing in Science & Engineering* **9,** 90–95 (2007).
78. Scholz, J., Andersson, T. R., Vaughan, A., Requeima, J. & Turner, R. E. Sim2real for environmental neural processes. *arXiv preprint arXiv:2310.19932* (2023).
79. Chai, J., Zeng, H., Li, A. & Ngai, E. W. Deep learning in computer vision: A critical review of emerging techniques and application scenarios. *Machine Learning with Applications* **6,** 100134. ISSN: 2666-8270. https://www.sciencedirect.com/ science/article/pii/S2666827021000670 (2021).
80. Deshmukh, A. M. Comparison of Hidden Markov Model and Recurrent Neural Network in Automatic Speech Recogni- tion. *European Journal of Engineering and Technology Research* **5,** 958–965. https://www.ej-eng.org/index.php/ejeng/ article/view/2077 (Aug. 2020).
81. Gordon, J., Bronskill, J., Bauer, M., Nowozin, S. & Turner, R. *Meta-Learning Probabilistic Inference for Prediction* in *International Conference on Learning Representations* (2019). https://openreview.net/forum?id=HkxStoC5F7.
82. Cao, Y., Fang, Z., Wu, Y., Zhou, D.-X. & Gu, Q. Towards understanding the spectral bias of deep learning. *arXiv preprint arXiv:1912.01198* (2019).
83. Rahaman, N. *et al. On the Spectral Bias of Neural Networks* in *Proceedings of the 36th International Conference on Machine Learning* (eds Chaudhuri, K. & Salakhutdinov, R.) **97** (PMLR, Sept. 2019), 5301–5310.

Acknowledgements. We gratefully acknowledge the agencies whose efforts in collecting, curating, and distributing datasets made this study possible. This work stands on the foundation of decades of contributions from the meteorological community and their commitment to sharing data. Specifically, we thank The European Organisation for the Exploitation of Meteorological Satellites, The UK MetOffice, The National Environmental Satellite, Data, and Information Service, The National Centers for Environmental Information, The National Oceanic and Atmospheric Administration, The National Climatic Data Center, NSF National Center for Atmospheric Research, The European Centre for Medium-Range Weather Forecasts. The JASMIN Environmental Data Service and WeatherBench2 project provided invaluable access to pre-processed data sources. This work was generously supported by The Alan Turing Institute, with funding and access to computational resources. Anna Allen acknowledges the UKRI Centre for Doctoral Training in the Application of Artificial Intelligence to the study of Environmental Risks (AI4ER), led by the University of Cambridge (EP/S022961/1), and studentship funding from Google DeepMind.

Stratis Markou acknowledges funding from the Vice Chancellor's and George & Marie Vergottis scholarship of the Cambridge Trust, and the Qualcomm Innovation Fellowship. Will Tebbutt acknowledges funding from Huawei and EPSRC grant (EP/W002965/1). James Requeima acknowledges funding from the Data Sciences Institute at the University of Toronto. J. Scott Hosking is supported by the Alan Turing Institute's Turing Research and Innovation Cluster in Digital Twins (TRICDT) and the Environment and Sustainability Grand Challenge, and EPSRC grant EP/Y028880/1. Richard E. Turner is supported an EPSRC Prosperity Partnership grant EP/T005386/1 between the University of Cambridge and Microsoft. We would like to thank Tomas Lazauskas for Cloud engineering support in setting up the compute platform, John Bronskill for technical advice on both compute and machine learning techniques, Peter Dueben for advice on baselines and Peter Lean for advice on counting the number of observations input to the IFS.

Competing Interests. The authors do not have any competing interests to declare.

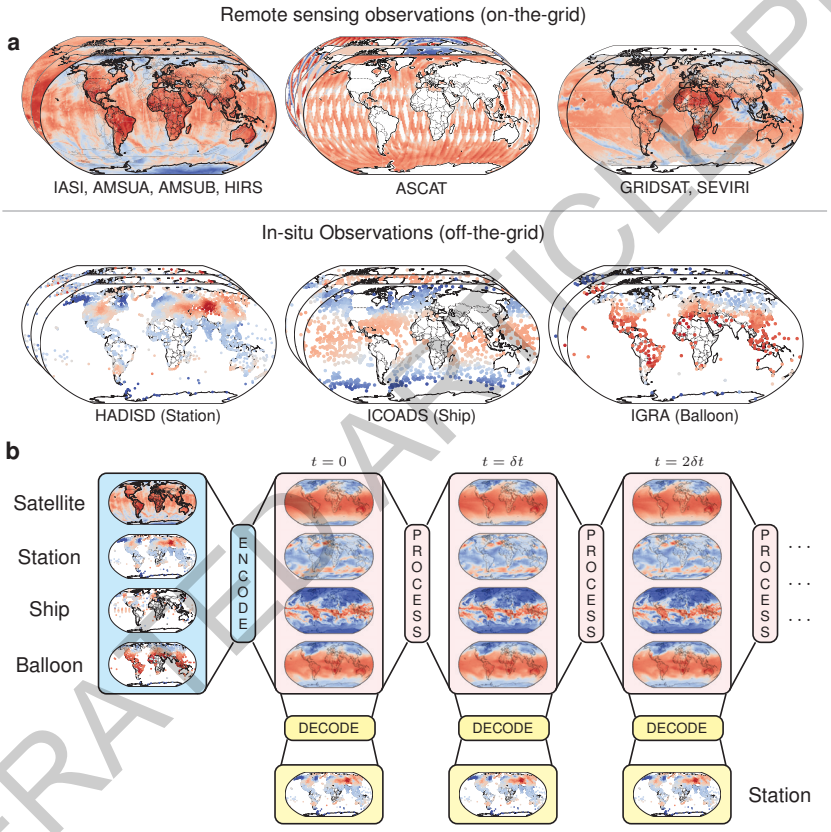
Author contributions. A.A and R.T conceptualised the project. A.A, S.M, W.T, J.R, W.B and R.T designed the experiments. A.A selected and collected all data and designed the end-to-end system. A.A, S.M and W.T implemented the codebase. A.A, S.M, W.T and R.T wrote the initial draft of the paper and S.M produced all figures. T.A, M.H, N.L, M.C, S.H and all aforementioned authors provided feedback on results at various stages of the project and contributed to the final version of the manuscript.

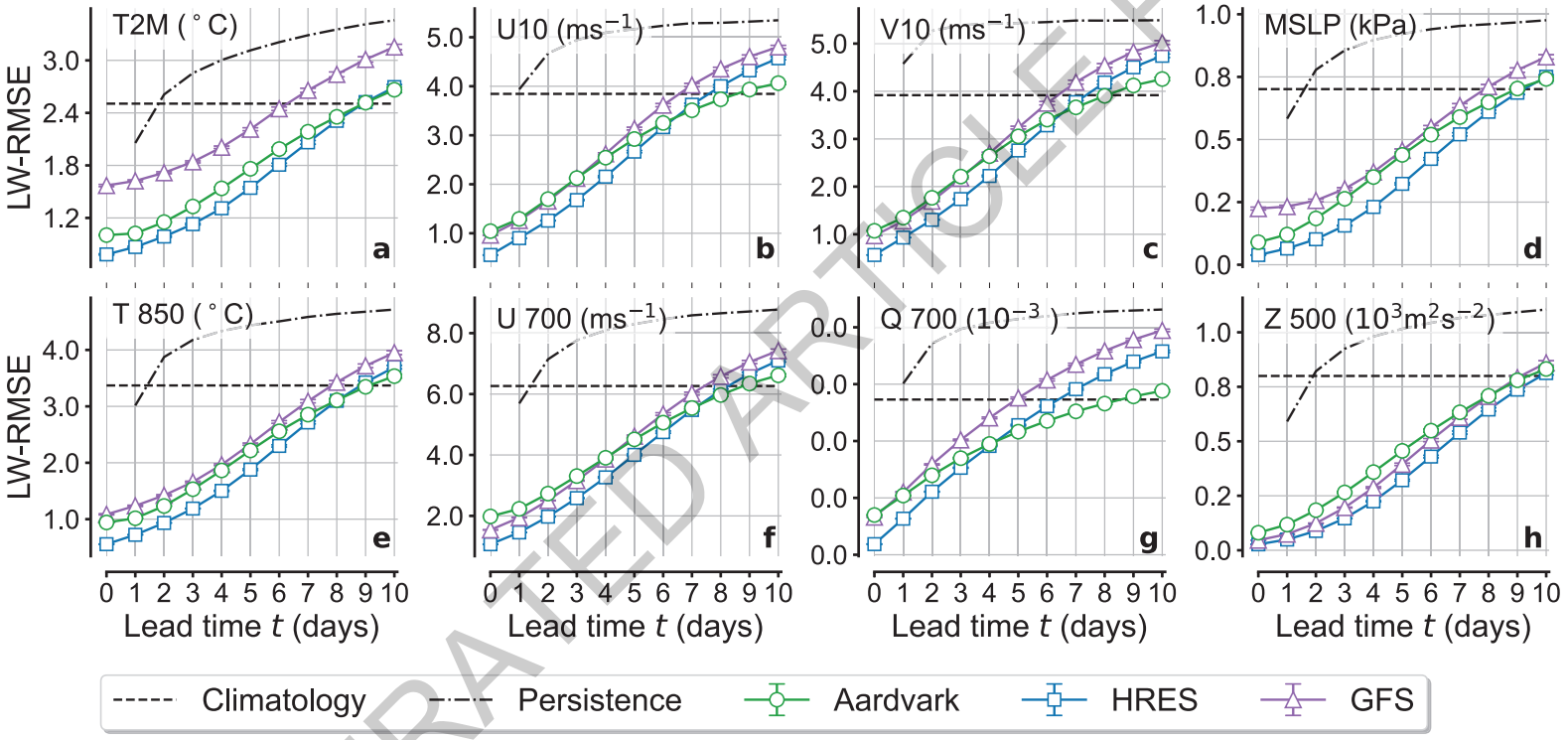
Supplementary information. Additional details on several aspects of this work, including supplementary figures and further discussion, are available in the supplementary information section, and rely on supplementary references [78,79,80,81,82,83].

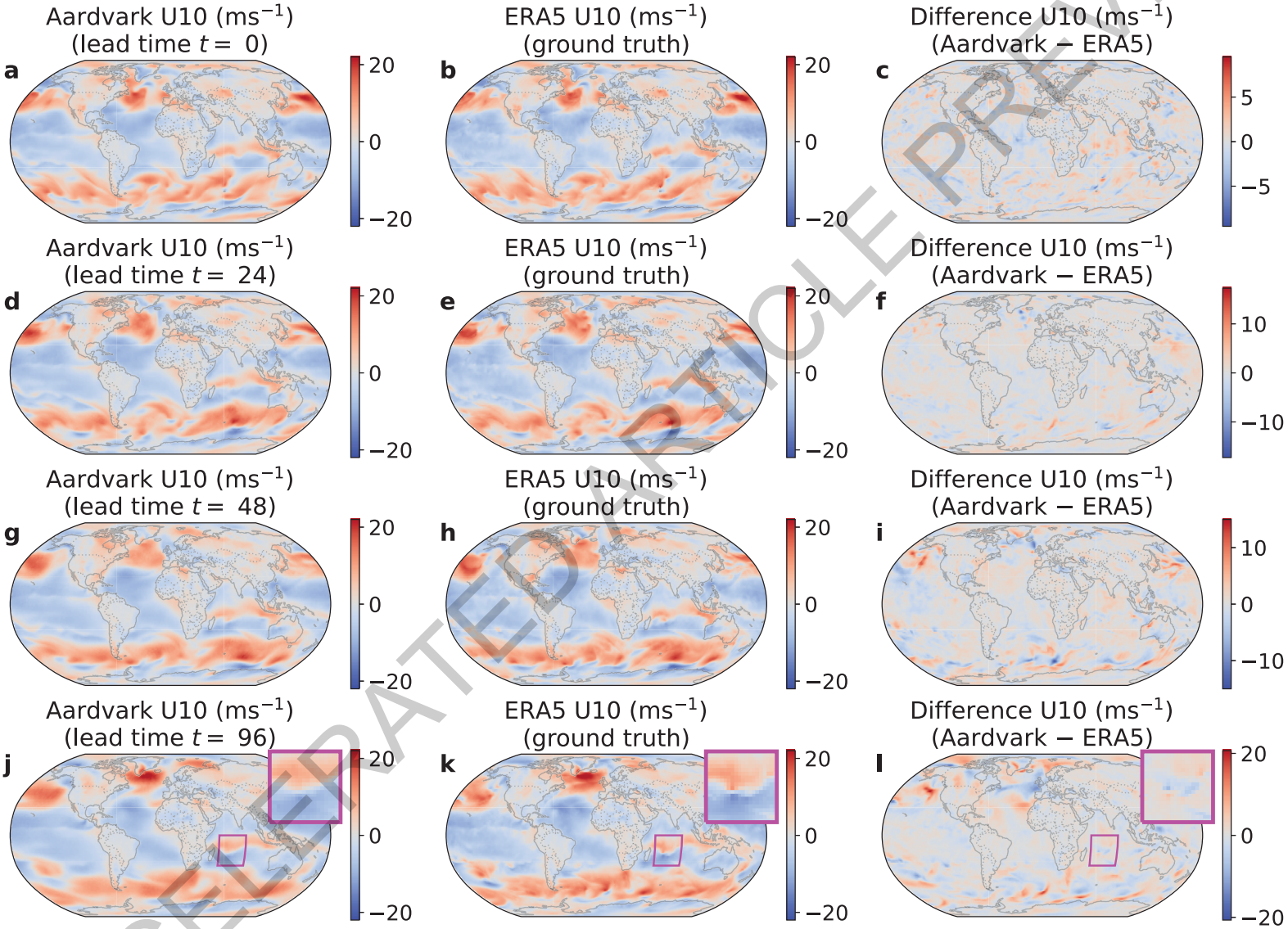
Extended display legends

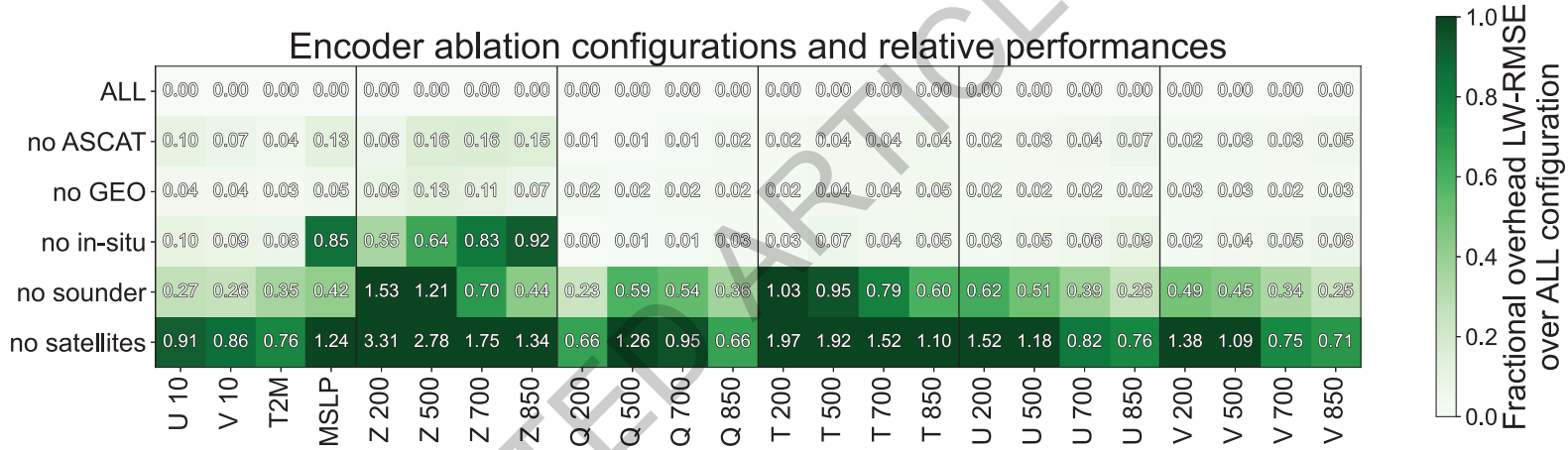
Table 1. Listing of the inputs and outputs of each module. *Raw data are passed to the encoder module which outputs predictions of the 24 prognostic variables on a global* 1.50^{o} *grid at* $t = 0$*. This initial state is then input to the processor module to produce predictions for each of the prognostic variables at lead times of one to ten days on the same grid. Finally, the decoder module takes these global predictions to local predictions at station locations.*

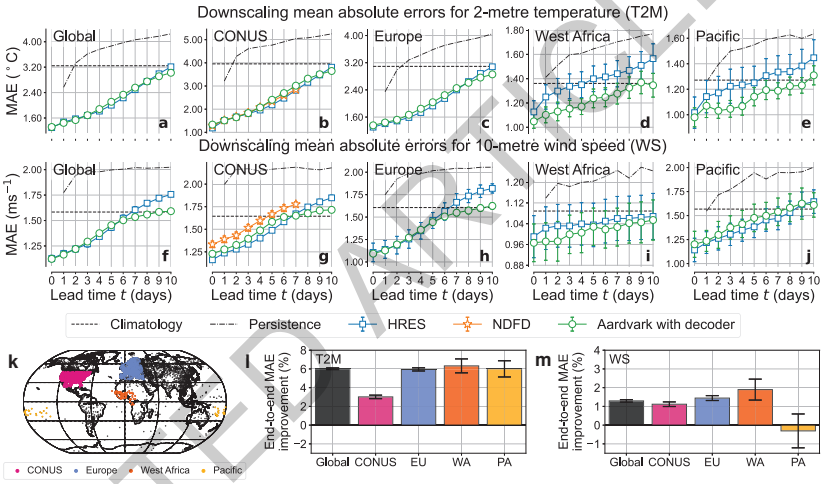
Table 2. Summary of the observational datasets used to train Aardvark. *Summary of the datasets, including the temporal window used in Aardvark. The acronyms "IR" and "MW" stand for infrared and microwave respectively.*











Module	Inputs		Outputs	
Encoder	in-situ (time 0)	HadISD, ICOADS, IGRA	surface (time 0)	T2M, U10, V10, MSLP
	remote sensing (time 0)	ASCAT, AMSU-A & B, HIRS, IASI, GRIDSAT		
	aux. varying (time 0)	ToY, ToD, Clim.	upper level (time 0) @ 200, 500, 700, 850 hPa	Q, Z, T, U, V
	aux. static	Orography		
Processor	surface (time t)	T2M, U10, V10, MSLP	surface (time t + 1)	T2M, U10, V10, MSLP
	upper level (time t) @ 200, 500, 700, 850 hPa	Q, Z, T, U, V		
	aux. varying (time t)	ToY, ToD	upper level (time t + 1) @ 200, 500, 700, 850 hPa	Q, Z, T, U, V
	aux. static	Orography		
Decoder	surface (time t)	T2M, U10, V10, MSLP	in-situ, surface (time t)	HadISD T2M HadISD WS
	upper level (time t)	Q, Z, T, U, V		
	aux. varying (time t)	ToY, ToD		
	aux. static	Orography		

Extended Data Table 1

Dataset	Instrument	Orbit	Platform	Provider	Window
ASCAT	Scatterometer	LEO	MetOp-A, B, C	EUMETSAT [9]	24 hours
AMSU-A	MW sounder	LEO	NOAA-15 to 18, MetOp-A, Aqua	NOAA [10]	24 hours
AMSU-B / MHS	MW sounder	LEO	NOAA-15 to 19, MetOp-A	NOAA [11]	24 hours
HIRS	IR sounder	LEO	NOAA-18 & 19 MetOp-A & B	EUMETSAT [12]	24 hours
IASI	IR sounder	LEO	MetOp-A	EUMETSAT [13]	24 hours
GRIDSAT	IR sounder	GEO	GOES, MSG, FengYun, Himawari	NOAA[14]	*t = 0*
HadISD	Land station	-	-	UK Met Office [15]	*t = 0*
ICOADS	Ship report	-	-	NOAA [16]	6 hours
IGRA	Radiosonde	-	-	NOAA [17]	6 hours

Extended Data Table 2