ArcticBench and SmartTransfer: Benchmarking and Enabling Continual Learning of Atmosphere Generative Foundation Model

Supplementary Material

Algorithm 1 SMARTTRANSFER: Structure-Aware Weight Transfer for Continual Learning of Generative Foundation Models

Require: Pretrained parameters Θ^{src} at level ℓ_{src} with C_{src} variables; target model Θ^{tgt} (randomly drawn) at level ℓ_{tgt} with C_{tgt} variables
Ensure: Adapted Θ^{tgt}

for each named parameter $(\text{name}, \theta^{\mathrm{src}}) \in \Theta^{\mathrm{src}}$ **do**
  if name $=$ `pos_embed` **then**
    $r \leftarrow 4^{(\ell_{\mathrm{src}} - \ell_{\mathrm{tgt}})}$
    $\theta^{\mathrm{tgt}} \leftarrow \mathrm{AvgPool}_r(\theta^{\mathrm{src}})$ {Main Eq. (8): spatial adaptation}
  else if name $=$ `enc_input_conv` **then**
    Transfer positional slice; keep variable slice random {Main Eq. (9)}
  else if name $\in \mathcal{K}_{\mathrm{decoder\_output}}$ **then**
    Keep random defaults {§1.1.2}
  else if $\mathrm{shape}(\theta^{\mathrm{src}}) = \mathrm{shape}(\theta^{\mathrm{tgt}})$ **then**
    $\theta^{\mathrm{tgt}} \leftarrow \theta^{\mathrm{src}}$ {Eq. (2): backbone copy}
  end if
end for

1. Additional SmartTransfer Details

For completeness, this appendix provides the SmartTransfer details omitted from the main paper for space. In particular, we make explicit the two transfer cases that are only summarized in the body text—backbone direct copy and output-head re-initialization—and we provide a compact algorithmic summary and the full continual-adaptation protocol.

1.1. Complete Structure-Aware Transfer Rule

SmartTransfer acts on four parameter components of the source checkpoint: (i) the HEALPix positional embedding, (ii) the positional slice of the first encoder convolution, (iii) the shared denoising backbone, and (iv) the variable-specific output head. The complementary variable-specific slice of the first encoder convolution is intentionally left at its target initialization, since its semantics depend directly on the target variable inventory.

Explicit encoder-slice offsets. For clarity, Eq. (9) can be written in offset form as

$$\mathbf{W}^{\mathrm{tgt}}_{\mathrm{enc}}[:,\, d_{\mathrm{tgt}} : d_{\mathrm{tgt}} + C_{\mathrm{pe}},\, :, :] \leftarrow \mathbf{W}^{\mathrm{src}}_{\mathrm{enc}}[:,\, d_{\mathrm{src}} : d_{\mathrm{src}} + C_{\mathrm{pe}},\, :, :]. \tag{1}$$

where $d_{\mathrm{src}} = 3C_{\mathrm{src}}$ and $d_{\mathrm{tgt}} = 3C_{\mathrm{tgt}}$ denote the offsets at which the positional channels begin in the source and target encoder inputs, respectively. This makes explicit that SmartTransfer copies only the domain-invariant positional slice of the input projection, while leaving the variable-conditioned slice free to specialize to the target domain.

1.1.1. Backbone Direct Copy

All intermediate U-Net parameters whose tensor shapes are shared by the source and target configurations are copied verbatim:

$$\theta^{\mathrm{tgt}}_k \leftarrow \theta^{\mathrm{src}}_k, \qquad \forall\, k \in \mathcal{K}_{\mathrm{backbone}}. \tag{2}$$

Here $\mathcal{K}_{\mathrm{backbone}}$ includes the residual blocks, normalization layers, attention modules, any learned skip-path projections, and the noise-level embedding network. These parameters encode domain-general denoising priors—multi-scale feature extraction, long-range spatial coupling, and noise-conditioned refinement—that transfer directly once the spatial and channel interfaces have been aligned.

1.1.2. Output-Layer Re-initialization

The final decoder projection maps internal features to the target variable channels. Because the source and target variable inventories differ both in dimensionality and in physical semantics, these layers are not meaningfully transferable and are therefore kept at their random target defaults:

$$\theta^{\mathrm{tgt}}_k \leftarrow \theta^{\mathrm{init}}_k, \qquad \forall\, k \in \mathcal{K}_{\mathrm{out}}, \tag{3}$$

where $\mathcal{K}_{\mathrm{out}}$ denotes the final decoder convolution(s) and associated bias terms. Re-initializing only this irreducibly variable-specific interface avoids negative transfer while allowing the shared denoising trunk to be reused almost entirely.

Taken together, SmartTransfer changes only the parts of the model that are structurally tied to resolution or variable semantics: the positional tensor is adapted to the target HEALPix level, the positional slice of the encoder input projection is copied, the shared backbone is copied exactly, and the variable-specific interfaces are left free to specialize to the target domain.

1.2. Algorithmic Summary

Algorithm 1 summarizes the full SmartTransfer procedure. The algorithm is agnostic to the particular source and target domains and applies whenever the source and target models share the same backbone architecture.

1.3. Implementation Details for Continual Adaptation

After SmartTransfer, the target model—**ArcticBottleSR** in our CARRA experiments—is fine-tuned end-to-end with the EDM objective (Eq. (3)). In all experiments, we use $P_{\text{mean}} = -1.2$, $P_{\text{std}} = 1.2$, and $\sigma_{\text{data}} = 0.5$.

Optimization uses SGD with momentum 0.9 and a two-group learning-rate schedule:

- **Non-positional parameters:** all copied backbone weights together with the newly initialized variable-specific encoder and decoder layers use a base learning rate of 10^{-7}, chosen to preserve transferred features while allowing gradual target-domain adaptation.
- **Positional embedding:** the HEALPix positional embedding adapted by Eq. (8) uses an elevated learning rate of 5×10^{-4} to enable rapid adaptation to the target grid geometry.

Both parameter groups follow a `StepLR` schedule with decay factor $\gamma = 0.6$ every 8k steps. We train with batch size 64 for 20k steps and clip the gradient norm at 10^6. For numerical stability under the EDM objective, the sampled noise level is clamped to a finite interval $[\sigma_{\min}, \sigma_{\max}]$ in both the noise-sampling procedure and the loss weighting $\lambda(\sigma)$. Model selection is performed by validation loss, and we retain the top-5 checkpoints for analysis.

This protocol intentionally separates fast adaptation of the resolution-dependent spatial prior from slow drift of the transferred denoising backbone. In practice, this separation is important: the positional embedding must re-specialize to the target HEALPix geometry much more quickly than the shared denoising trunk should be allowed to move.

2. Additional ArcticBench Details

This appendix collects the ArcticBench details omitted from the main paper for space, including the extended motivation, data curation pipeline, storage format, NaN-safe preprocessing, split definition, exact mask construction, metric formulas, and released artefacts.

2.1. Extended Motivation and Design Principles

The Arctic is undergoing rapid transformation. Temperatures in the region are rising at nearly four times the global average [10], amplifying sea-ice retreat, accelerating permafrost thaw, and reshaping ocean–atmosphere interactions with downstream consequences for mid-latitude weather [4]. The region is also becoming increasingly operationally important, increasing demand for accurate atmospheric state estimation and uncertainty-aware prediction. Yet the Arctic remains one of the most observation-sparse areas on Earth: the network of ground-based monitoring stations north of 60°N is roughly an order of magnitude sparser than at temperate latitudes [2], and satellite retrievals are additionally degraded by persistent cloud cover, low solar angles, and the radiometric ambiguity between snow, ice, and cloud surfaces [8]. These conditions make the Arctic a critical and under-served test bed for atmospheric generative models.

Existing benchmarks for learned weather and climate models (*e.g.*, WeatherBench, WeatherBench2, and ClimateBench [11, 12, 14]) mainly evaluate global or tropical–midlatitude domains. While indispensable, they underemphasize the Arctic because (i) area-weighted global metrics are dominated by the tropics and midlatitudes, (ii) latitude–longitude grids introduce strong geometric redundancy near the poles, and (iii) Arctic-specific regimes such as sea-ice variability, katabatic winds, and polar-vortex dynamics are not explicitly isolated. ArcticBench addresses these gaps through three design principles:

1. **Arctic-native supervision.** We anchor the benchmark on CARRA, a regional reanalysis produced with an Arctic-optimized data assimilation system, rather than on a polar subset of a global product. CARRA assimilates Arctic-relevant observations, including buoys, drifting stations, and high-latitude radiosondes, yielding a stronger regional reference than generic global reanalyses may provide [6, 13].
2. **Equal-area spherical evaluation.** All data and metrics are defined on the HEALPix NEST grid [5], which partitions the sphere into equal-area pixels irrespective of latitude. This removes the high-latitude over-sampling inherent in equirectangular grids and ensures that each evaluation pixel contributes equally to aggregate scores.
3. **Geophysically stratified reporting.** ArcticBench provides region-specific and tail-sensitive diagnostics in addition to domain-wide averages, enabling finer assessment of Arctic failure modes.

2.2. Data Curation Pipeline

Transforming the raw CARRA archive into a model-ready benchmark involves data acquisition, vertical harmonization, spherical projection, and quality control. We release the full pipeline as open-source, version-controlled scripts to ensure exact reproducibility.

2.2.1. Source Data Acquisition

We retrieve CARRA fields from the Copernicus Climate Data Store [3] via its programmatic API. Three complementary retrieval streams are used to capture the atmospheric state:

- **Pressure-level fields:** geopotential, temperature, specific humidity, and u/v wind components on nine standard isobaric surfaces (1000, 925, 850, 700, 500, 300, 200, 50, and 10 hPa).
- **Single-level fields:** 2 m temperature (`2t`), 10 m wind components (`10u`, `10v`), mean sea-level pressure (`msl`), surface pressure (`sp`), skin temperature (`skt`), total cloud cover (`tcc`), sea-surface temperature (`sst`), sea-ice area fraction (`sic`), and snow-depth water equivalent (`sd`).
- **Forecast-accumulated fields:** total precipitation, evaporation, surface net solar radiation, and surface latent heat flux, retrieved as 3-hour forecast products.

Data are downloaded in GRIB format at 3-hourly resolution (00, 03, 06, . . . , 21 UTC) over both the east and west CARRA domains, which together encompass the European Arctic, Greenland, Iceland, and the surrounding maritime region. Monthly files are written incrementally, enabling resumable downloads for the multi-terabyte archive.

2.2.2. Vertical Harmonization

CARRA variables are provided on heterogeneous vertical coordinates: (i) standard pressure levels, (ii) hybrid model levels requiring an accompanying pressure field for interpolation, and (iii) single levels (surface or column-integral variables). Our preprocessing pipeline detects the vertical mode of each input field and routes processing accordingly:

- *Pressure-level mode.* Fields already on isobaric surfaces are selected directly at the target pressure levels.
- *Model-level mode.* Fields on native hybrid-sigma coordinates are vertically interpolated to the target pressure levels using column-wise log-linear interpolation in pressure, yielding physically consistent profiles across the domain.
- *Surface mode.* Two-dimensional fields are stored as-is with a singleton vertical dimension for schema compatibility.

This multi-path design abstracts away coordinate-specific handling while preserving physical consistency.

2.2.3. Projection to the HEALPix NEST Grid

All fields are projected from the native CARRA Lambert conformal conic grid to a global HEALPix NEST grid at level 6 ($N_{\text{side}} = 64$, $N_{\text{pix}} = 12 \times 4^6 = 49{,}152$). At this resolution, each pixel spans approximately 0.92° ($\sim$55 km at the equator). HEALPix is the natural discretization for our setting: it provides equal-area pixels at all latitudes, supports efficient hierarchical coarsening for multi-resolution [9] conditioning, and is natively used by the cBottle generative architecture [1].

The projection is performed via angular binning: each native CARRA grid point is mapped to its enclosing HEALPix pixel, and when multiple native points fall into the same pixel their values are averaged. Pixels outside the CARRA domain remain as IEEE 754 NaN, providing an implicit regional validity mask.

The resulting tensors are stored in Zarr format with the schema below:

Array	Shape	Description
`time`	$(T,)$	datetime64[ns] timestamps
`levels`	$(L,)$	int32 pressure levels [hPa]
`<var_3d>`	(T, L, N_{pix})	pressure-level variables
`<var_2d>`	(T, N_{pix})	single-level variables

Each array is chunked as $(1, 1, N_{\text{pix}})$ for 3-D variables and $(1, N_{\text{pix}})$ for 2-D variables, enabling efficient random access to individual time steps during training. Zarr metadata are consolidated after writing, allowing schema inspection without scanning data chunks.

2.2.4. Quality Control and NaN-Safe Preprocessing

Because CARRA is a regional product, most of the global HEALPix grid lies outside its domain and is stored as NaN. Naïvely propagating these values through normalization, pooling, or convolution would corrupt the training signal. Our preprocessing therefore adopts a NaN-safe design throughout:

- *Normalization statistics* (per-channel mean and standard deviation) are computed only over finite-valued pixels in the training split using Welford's online algorithm [15], ensuring that out-of-domain NaNs do not bias the statistics.
- *After normalization*, NaN pixels are replaced with zero—the normalized population mean—so that pooling and convolution operators remain well defined everywhere. This is equivalent to a mean-valued background assumption and prevents NaN propagation through the pipeline.
- *Evaluation masks* are derived from the NaN pattern of a reference field, guaranteeing that metrics are computed strictly over the CARRA-valid region.

2.3. Benchmark Splits and Dataset Interface

We partition the curated time series into training (first 75%) and test (last 25%) splits via a deterministic temporal cutoff, ensuring strict temporal separation and preventing leakage. No shuffling is applied: contiguous training and test periods preserve the temporal autocorrelation structure of the atmosphere and avoid optimistically biased evaluations.

Within the training split, per-channel z-score normalization statistics are estimated from a subsample of up to 200 uniformly selected time steps to keep preprocessing tractable; these statistics are then frozen and reused during evaluation.

During training, each sample is a single time step yielding a tensor of shape $(C, 1, N_{\text{pix}})$, where the channel dimension C indexes the flattened variable–level representation. For a configuration with pressure-level variable set $\mathcal{V}_{\text{3D}}$ and

single-level variable set $\mathcal{V}_{\text{2D}}$, the channel count is

$$C = \sum_{v \in \mathcal{V}_{\text{3D}}} L_v + |\mathcal{V}_{\text{2D}}|, \tag{4}$$

where L_v denotes the number of retained pressure levels for variable v.

2.4. Versioned Masks and Arctic-Stratified Metrics

A key contribution of ArcticBench is a geophysically motivated, spatially stratified evaluation suite that goes beyond a single aggregate score. We define three complementary evaluation axes: regional stratification, tail diagnostics, and spatial realism assessment.

2.4.1. Regional Stratification

All metrics are computed over three nested spatial regions, defined on the HEALPix grid by latitude thresholds applied to CARRA-valid pixels:

- **Core Arctic** ($\phi \geq 66.5°\text{N}$): the region poleward of the Arctic Circle.
- **Boundary Arctic** ($60°\text{N} \leq \phi < 66.5°\text{N}$): the sub-Arctic transitional zone.
- **All Arctic**: the union of the two masks above.

The latitude thresholds and mask construction are versioned (currently v1.0) and deterministic given the HEALPix level and the CARRA Zarr store, ensuring exact reproducibility across studies. At HEALPix level 6, the CARRA-valid domain comprises 520 pixels, of which 446 (85.8%) fall in the core Arctic and 74 (14.2%) in the boundary Arctic.

2.4.2. Standard Metrics

For each variable and region, we report root mean squared error (RMSE) and mean absolute error (MAE):

$$\text{RMSE} = \sqrt{\frac{1}{|\mathcal{M}|} \sum_{i \in \mathcal{M}} (\hat{y}_i - y_i)^2}, \tag{5}$$

$$\text{MAE} = \frac{1}{|\mathcal{M}|} \sum_{i \in \mathcal{M}} |\hat{y}_i - y_i|, \tag{6}$$

where $\hat{y}$ denotes the model prediction, y the CARRA reference, and $\mathcal{M}$ the set of pixels in the spatial mask.

We additionally report macro-averaged scores that weight all variables equally regardless of physical scale:

$$\text{Macro-RMSE} = \frac{1}{|\mathcal{V}|} \sum_{v \in \mathcal{V}} \text{RMSE}_v, \tag{7}$$

$$\text{Macro-MAE} = \frac{1}{|\mathcal{V}|} \sum_{v \in \mathcal{V}} \text{MAE}_v, \tag{8}$$

where $\mathcal{V}$ is the set of evaluated variables.

2.4.3. Extreme-Quantile Metrics

Accurate representation of extreme events is especially important for Arctic applications. We therefore evaluate tail performance by restricting metrics to pixels where the absolute value of the ground-truth field exceeds a specified quantile threshold q:

$$\text{RMSE}_q = \sqrt{\frac{1}{|\mathcal{E}_q|} \sum_{i \in \mathcal{E}_q} (\hat{y}_i - y_i)^2}, \tag{9}$$

where

$$\mathcal{E}_q = \{i \in \mathcal{M} : |y_i| \geq Q_q(|y|)\}, \tag{10}$$

and Q_q denotes the q-th sample quantile. Computing quantiles over absolute values captures extremes in both tails for signed variables such as temperature and wind components. We report $\text{RMSE}_{0.95}$ and $\text{RMSE}_{0.99}$ by default.

2.4.4. Spatial Gradient Realism

Generative models can be globally accurate in L^2 error yet locally over-smoothed. We quantify spatial realism via a lightweight **gradient ratio**

$$\rho = \frac{\overline{|\nabla \hat{y}|}}{\overline{|\nabla y|}}, \tag{11}$$

where $\overline{|\nabla \cdot|}$ denotes the mean gradient magnitude over masked pixels, estimated from first-order differences over the HEALPix representation. A ratio $\rho \approx 1$ indicates that the prediction preserves the spatial texture of the reference, whereas $\rho < 1$ indicates over-smoothing.

2.5. Evaluation Protocol

We support two evaluation modes, trading off fidelity against computational cost:

1. **Diffusion sampling (default).** Full EDM [7] sampling with 18 steps, $\sigma_{\max} = 80$, and a fixed random seed for reproducibility.
2. **Single-step denoising (fast).** A small amount of noise ($\sigma = 0.002$) is added to the ground truth, and the model performs a single forward pass to reconstruct the clean field.

In both modes, the model receives low-resolution conditioning constructed by hierarchical average-pooling from HEALPix level 6 to level 4 ($16\times$ area coarsening) and bilinear regridding to a 128×128 latitude–longitude global context. All results are reported in physical (un-normalized) space.

For statistical reliability, we evaluate over three random seeds ($\{0, 1, 2\}$) and report mean $\pm$ standard deviation. The full evaluation code, mask definitions, and metric implementations are released alongside the dataset to enable exact reproduction.

2.6. Released Artefacts

We release ArcticBench as a benchmark package comprising four complementary artefacts:

1. **Data acquisition and preprocessing scripts** that download raw CARRA fields from CDS and produce analysis-ready HEALPix Zarr stores.
2. **A PyTorch `Dataset` implementation** that provides normalized, NaN-safe training samples with deterministic train/test splitting and automatic channel flattening.
3. **Versioned spatial masks and evaluation code** that produce Arctic-stratified metrics and qualitative map panels from any compatible checkpoint.
4. **Baseline results and checkpoints** for the transfer regimes studied in this paper, establishing reference performance numbers on the benchmark.

By standardizing the data format, evaluation geometry, metric definitions, and reporting protocol, ArcticBench reduces ambiguity in comparison across methods and institutions. We hope it can serve as a stable reference benchmark and be extended over time to additional Arctic-relevant variables, higher HEALPix resolutions, and temporal evaluation protocols.

References

[1] Noah D Brenowitz, Tao Ge, Akshay Subramaniam, Peter Manshausen, Aayush Gupta, David M Hall, Morteza Mardani, Arash Vahdat, Karthik Kashinath, and Michael S Pritchard. Climate in a bottle: Towards a generative foundation model for the kilometer-scale global atmosphere. *arXiv preprint arXiv:2505.06474*, 2025. 3

[2] David H Bromwich, Ryan L Fogt, Kevin I Hodges, and John E Walsh. A tropospheric assessment of the era-40, ncep, and jra-25 global reanalyses in the polar regions. *Journal of Geophysical Research: Atmospheres*, 112(D10), 2007. 2

[3] Carlo Buontempo, Samantha N Burgess, Dick Dee, Bernard Pinty, Jean-Noël Thépaut, Michel Rixen, Samuel Almond, David Armstrong, Anca Brookshaw, Angel Lopez Alos, et al. The copernicus climate change service: climate science in action. *Bulletin of the American Meteorological Society*, 103(12):E2669–E2687, 2022. 2

[4] Judah Cohen, James A Screen, Jason C Furtado, Mathew Barlow, David Whittleston, Dim Coumou, Jennifer Francis, Klaus Dethloff, Dara Entekhabi, James Overland, et al. Recent arctic amplification and extreme mid-latitude weather. *Nature geoscience*, 7(9):627–637, 2014. 2

[5] Krzysztof M Gorski, Eric Hivon, Anthony J Banday, Benjamin D Wandelt, Frode K Hansen, Mstvos Reinecke, and Matthia Bartelmann. Healpix: A framework for high-resolution discretization and fast analysis of data distributed on the sphere. *The Astrophysical Journal*, 622(2):759–771, 2005. 2

[6] Hans Hersbach, Bill Bell, Paul Berrisford, Shoji Hirahara, András Horányi, Joaquín Muñoz-Sabater, Julien Nicolas, Carole Peubey, Raluca Radu, Dinand Schepers, et al. The era5 global reanalysis. *Quarterly journal of the royal meteorological society*, 146(730):1999–2049, 2020. 2

[7] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in neural information processing systems*, 35:26565–26577, 2022. 4

[8] JE Kay, BR Hillman, SA Klein, Y Zhang, Brian Medeiros, R Pincus, Andrew Gettelman, B Eaton, J Boyle, R Marchand, et al. Exposing global cloud biases in the community atmosphere model (cam) using satellite observations and their corresponding instrument simulators. *Journal of Climate*, 25 (15):5190–5207, 2012. 2

[9] Puneet Mathur, Atula Neerkaje, Malika Chhibber, Ramit Sawhney, Fuming Guo, Franck Dernoncourt, Sanghamitra Dutta, and Dinesh Manocha. Monopoly: Financial prediction from monetary policy conference videos using multimodal cues. In *Proceedings of the 30th ACM international conference on multimedia*, pages 2276–2285, 2022. 3

[10] Mika Rantanen, Alexey Yu Karpechko, Antti Lipponen, Kalle Nordling, Otto Hyvärinen, Kimmo Ruosteenoja, Timo Vihma, and Ari Laaksonen. The arctic has warmed nearly four times faster than the globe since 1979. *Communications earth & environment*, 3(1):168, 2022. 2

[11] Stephan Rasp, Peter D Dueben, Sebastian Scher, Jonathan A Weyn, Soukayna Mouatadid, and Nils Thuerey. Weatherbench: a benchmark data set for data-driven weather forecasting. *Journal of Advances in Modeling Earth Systems*, 12 (11):e2020MS002203, 2020. 2

[12] Stephan Rasp, Stephan Hoyer, Alexander Merose, Ian Langmore, Peter Battaglia, Tyler Russell, Alvaro Sanchez-Gonzalez, Vivian Yang, Rob Carver, Shreya Agrawal, et al. Weatherbench 2: A benchmark for the next generation of data-driven global weather models. *Journal of Advances in Modeling Earth Systems*, 16(6):e2023MS004019, 2024. 2

[13] H. Schyberg, X. Yang, M.A.Ø. Køltzow, B. Amstrup, Å. Bakketun, E. Bazile, J. Bojarova, J.E. Box, P. Dahlgren, S. Hagelin, M. Homleid, A. Horányi, J. Høyer, Å. Johansson, M.A. Killie, H. Körnich, P. Le Moigne, M. Lindskog, T. Manninen, P. Nielsen Englyst, K.P. Nielsen, E. Olsson, B. Palmason, C. Peralta Aros, R. Randriamampianina, P. Samuelsson, R. Stappers, E. Støylen, S. Thorsteinsson, T. Valkonen, and Z.Q. Wang. Arctic regional reanalysis on pressure levels from 1991 to present, 2020. Accessed on DD-MMM-YYYY. 2

[14] Duncan Watson-Parris, Yuhan Rao, Dirk Olivié, Øyvind Seland, Peer Nowack, Gustau Camps-Valls, Philip Stier, Shahine Bouabid, Maura Dewey, Emilie Fons, et al. Climatebench v1. 0: A benchmark for data-driven climate projections. *Journal of Advances in Modeling Earth Systems*, 14 (10):e2021MS002954, 2022. 2

[15] Barry Payne Welford. Note on a method for calculating corrected sums of squares and products. *Technometrics*, 4(3): 419–420, 1962. 3

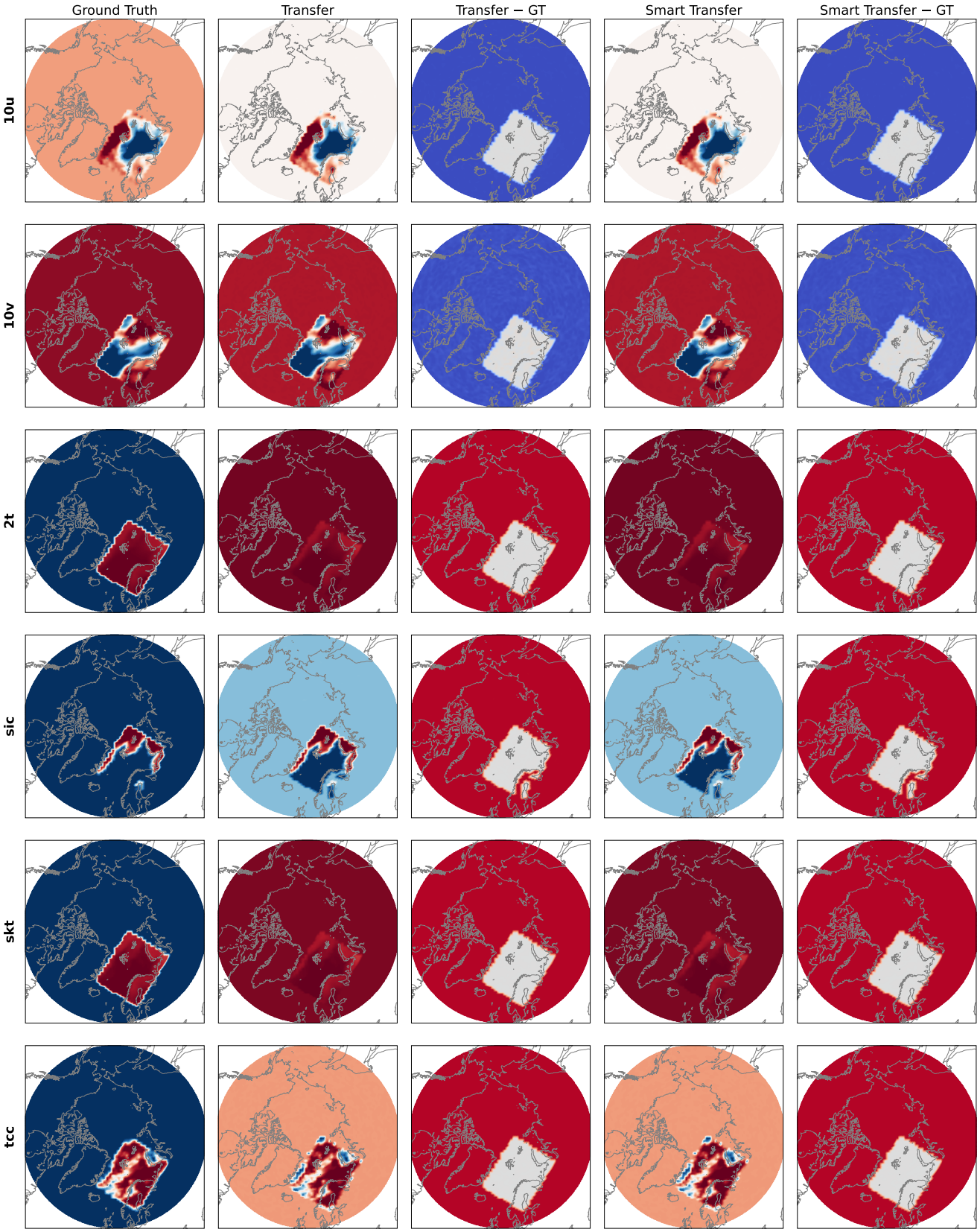

Figure 1. ArcticBench Evaluation of Transfer and Smart Transfer using Atmosphere Foundation Model