

Mechanistic Interpretability of Atmospheric Rivers in GraphCast

Madelyn Mathai
University of Virginia

Timothy B. Higgins
University of Virginia

Kevin M. Grise
University of Virginia

Chirag Agarwal
University of Virginia

Antonios Mamalakis
University of Virginia

Abstract

While AI weather models now rival operational forecasts, how they represent the atmosphere internally remains an open question: feature attribution reveals which input patterns matter, not what the model computes or how it combines information internally. We train sparse autoencoders (SAEs) on GraphCast to uncover its learned concepts, using atmospheric rivers as our phenomenon of focus. Both standard and Matryoshka SAEs show GraphCast computes atmospheric river intensity, measured by integrated vapor transport (IVT), as a stable internal variable, despite IVT being neither an input nor a target. In contrast to the unstructured concept retrieval of the standard SAE, the Matryoshka SAE orders concepts by importance and exposes their relations. Atmospheric river concepts persist across depth and direct interventions confirm causality. This method offers a way to find internal variables and determine which of them the model actually relies on, which is a prerequisite for asking whether those variables remain meaningful as the phenomenon changes under a warming climate. Code available at https://github.com/AikyamLab/climate_xai.

1 Introduction

As deep neural networks (DNNs) are increasingly integrated into forecasting operations [1, 2, 3], it is crucial to understand how their predictions are made not only for assessing model accuracy, but also for enabling scientific discovery [4]. While NWP systems are built on physical formulations and a shared theoretical foundation [5], DNNs offer little to no understanding behind their decisions, hindering their adoption and scientific credibility [6, 7, 8]. Prior work explains AI weather and climate models with post-hoc feature attribution, including relevance propagation, saliency, and perturbation methods [9, 10, 11, 12, 13, 14, 15], which reveal where a prediction is sensitive but not what the model has learned internally. Prior works that apply sparse autoencoders (SAEs) to climate science models only apply a standard SAE at a single depth [16], serving as an implementation rather than a framework.

In this study, we focus on interpreting how GraphCast, an AI weather model, has learned to represent atmospheric rivers (ARs). We investigate how coherent internal representations, namely *concepts*, can be identified in a network, how they are organized relative to each other, and how they can be traced across the network. To do this, we train two dictionary-learning architectures, Standard SAEs [17] and Matryoshka SAEs [18], at layers 0, 8, and 15 within the model (Appendix A).

We measure AR intensity with integrated vapor transport (IVT), a known function of GraphCast’s humidity and wind inputs that lets us ground SAE latents in physical quantities. Our analysis shows that, although trained only to forecast and never on IVT, GraphCast computes AR intensity as a stable internal variable. The Matryoshka dictionary makes this concept not only detectable but also traceable in its relationship to other concepts. Direct intervention on the forward pass confirms the concept is

Tackling Climate Change with Machine Learning: workshop at NeurIPS 2026. Corresponding author: Madelyn Mathai.

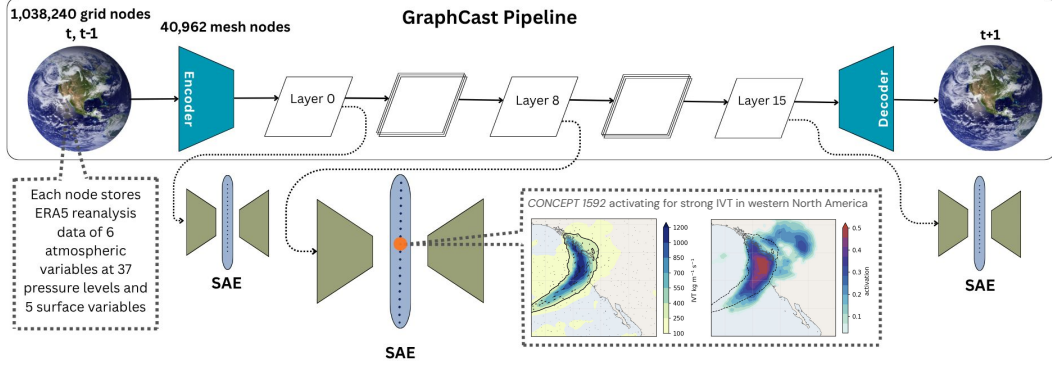


Figure 1: **Top:** GraphCast maps ERA5 grid inputs at times $t - 1$ and t (227 variables per node) onto an icosahedral mesh, where a 16-layer processor updates the 512-dimensional mesh-node activations before the decoder returns the forecast at $t + 1$. **Bottom:** We train SAEs post-hoc on the mesh-node activations at layers 0, 8, and 15, each encoding the 512-dimensional activation into a wider ($\mathbb{R}^{4096}$), sparsely active dictionary and reconstructing it. **Inset:** One of many AR concepts. The activation pattern of CONCEPT 1592 at layer 8, which responds to strong IVT over western North America.

causal, weakening real ARs across the globe when removed and intensifying their precipitation when written in. Because the approach extends to other phenomena and variables, this moves interpretability of weather models beyond input attribution toward a mechanistic account of what the model computes internally, how that computation is organized, and whether the forecast actually depends on it.

2 Methods

Fig. 1 shows GraphCast, a graph neural network (GNN) that autoregressively forecasts the global atmosphere at 0.25° resolution on a multiscale icosahedral mesh. GraphCast is a 16-layer GNN trained on ERA5 data from 1979 to 2017 at 6-hour time steps. The model predicts 227 fields spanning six atmospheric variables at 37 pressure levels and five surface variables, including 6-hourly accumulated precipitation.

The Matryoshka SAE’s ordinal dictionary allows us to explore how latents represent information relative to each other. The Matryoshka SAE prefix groups are $G_0=[0, 255]$, $G_1=[256, 511]$, $G_2=[512, 1023]$, $G_3=[1024, 2047]$, and $G_4=[2048, 4095]$, ordered from most to least important. In the Matryoshka SAE, each prefix must reconstruct the input on its own, so the core fills with the most frequently active latents and the outer shells with progressively rarer, more specific ones. A broad latent cannot cede its coverage to a narrower one without hurting the earlier prefix’s reconstruction [18]. Matryoshka construction leaves behind the parent–child ordering itself [19]: the latent from the earlier group can serve as the parent, and we can ask whether a later latent is a *nested* child within it. Where language SAEs must infer relationships from activation co-occurrence alone [20], GraphCast supplies a handle that text does not: each latent activation has a spatial footprint on the mesh at each timestep, so containment becomes a geometric, physical question. We measure nesting as *spatial containment*, $c = |\mathcal{F}^{\text{ch}} \cap \mathcal{F}^{\text{par}}|/|\mathcal{F}^{\text{ch}}|$, the fraction of the child’s (node, timestep) firings at which the parent also fires, equivalently the confidence of the rule “child $\Rightarrow$ parent” [21]. **Our work is the first exploration of Matryoshka nesting structure in a GNN setting.** This allows us to ask questions like *How do individual concepts work together to track atmospheric phenomena?* and *Does a broad concept refine into narrower children by region or by intensity?*, questions that a flat, unordered dictionary cannot pose and that ground the relationships between latents in physical space rather than in co-occurrence alone. See Appendix B for additional details on nesting.

3 Results

3.1 Matryoshka SAE shows nesting structure for global ARs

Does a broad AR concept have more specific children? CONCEPT 99 (core group G_0) is the dictionary’s strongest global AR concept, with a full-period mean correlation of $r = 0.51$ between

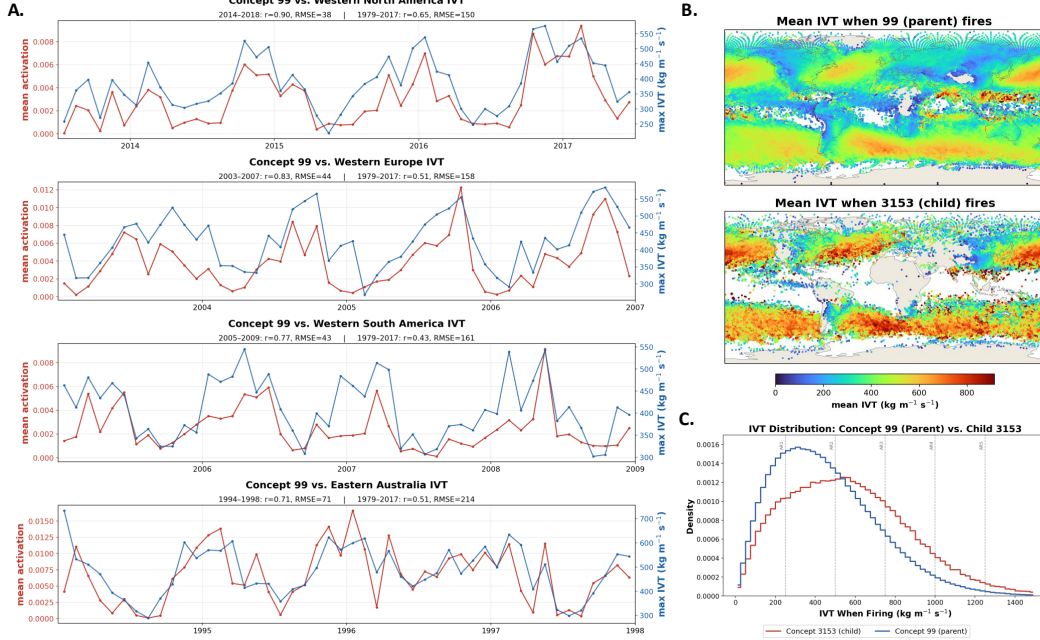


Figure 2: (A) CONCEPT 99 tracks regional AR intensity in four regions among the most frequently affected by ARs [22, 23, 24]: western North America (30–50°N, 115–130°W), western Europe (37–57°N, 8°W–7°E), western South America (30–50°S, 62–77°W), and eastern Australia (22–42°S, 147–162°E). Displayed are region-mean activation of Matryoshka-L8 CONCEPT 99 (red, left axis) against region-maximum IVT (blue, right axis) over AR timesteps, averaged into monthly means. Each panel displays the four-year window with the strongest monthly correlation. Full training-period correlation is printed above each panel. (B) Every mesh node where CONCEPT 99 and child CONCEPT 3153 fire across 8,000 sampled timesteps, colored by the mean IVT at that node over the timesteps it fires; blank regions are nodes where the concept never fires. (C) Distribution of node-level IVT over all sampled firing node–timesteps. Dashed lines indicate the AR category (Cat 1–5) thresholds based on magnitude alone; see Ralph et al. [25].

region-mean activation and region-max IVT across the four regions shown in Fig. 2A. Still, CONCEPT 99 fires diffusely, active at essentially every timestep and touching all but roughly 10% of mesh nodes over the record (Fig. 2B). CONCEPT 3153 is contained in 99 at $\hat{c} = 0.953$ (95% CI [0.949, 0.956], $n = 15,203$ firing events); 99’s next-most-contained latent reaches only 0.735 ([0.730, 0.739]). No other latent in G0 or G1 contains 3153, so CONCEPT 99 is unambiguously its single parent.

What distinguishes the child’s firings from the parent’s? Of 99’s firing events, 3153 co-fires on the more intense ones. As shown in Fig. 2C, its firing sites carry higher IVT on average (mean 570 versus 460 $\text{kg m}^{-1}\text{s}^{-1}$; $\text{Pr}[\text{IVT}_{3153} > \text{IVT}_{99}] = 0.61$), and the difference is sharpest in the extreme tail: 3153 fires on IVT above 750 $\text{kg m}^{-1}\text{s}^{-1}$ nearly twice as often as its parent (26% versus 14%). If the two distributions were completely non-overlapping, it would indicate *feature absorption* [26]: CONCEPT 99 would have stopped firing on extreme ARs and ceded them to its child, creating a “hole” in the monosemantic concept it fires for (ARs). Instead, CONCEPT 3153 co-fires during the highest-IVT events but does not take over for CONCEPT 99. The parent is present at 95% of the child’s firing sites, so 99 has not ceded these cases and is most active on them. It therefore nests by intensity: its child occupies the same geography but fires more on the strong end of the parent’s condition.

3.2 Steering shows causality of CONCEPT 99

To test causality, we scale a concept’s SAE code by a factor β at layer 8 and decode the change back into the mesh-node activations at every rollout step, so $\beta > 1$ amplifies the concept, $\beta = 1$ is the unedited baseline, and $\beta = 0$ removes it (Appendix C). As shown in Fig. 3, dialing CONCEPT 99 moves within-AR precipitation strongly and monotonically while leaving wind nearly unchanged, so the concept acts on AR intensity through a moisture rather than a dynamical channel.

IVT and column water vapor both increase with β at initialization, but under strong amplification the enhanced precipitation depletes the column faster than transport resupplies it, and both fall below baseline within a day. Removing the concept instead weakens the AR uniformly, lowering IVT, vapor, and precipitation together. In contrast, applying the same causality test to CONCEPT 3153 produced no effect (not shown). GraphCast therefore holds concepts that only detect ARs, like CONCEPT 3153, alongside those that drive them. Which of the many nested children are causal, and which merely detect, is a promising direction for future work.

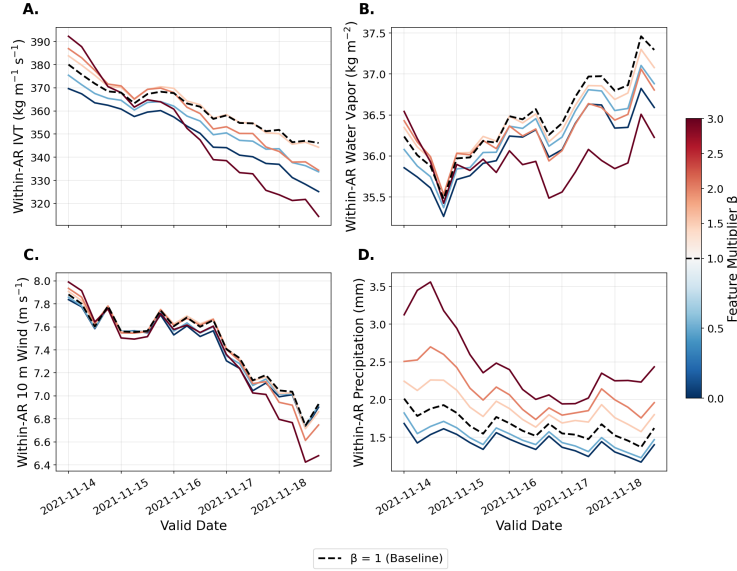


Figure 3: CONCEPT 99 dialed over a 5-day rollout of global ARs (initialized 13 Nov 2021, four years past the SAE training period). Panels show within-AR (A) IVT, (B) column water vapor, (C) 10 m wind, and (D) precipitation.

4 Conclusion and climate change impact

Atmospheric rivers drive much of the world’s extreme precipitation and flooding, and their intensity, the property warming is expected to change, is a quantity GraphCast was never given yet computes internally as a causal variable.

Across depth the Matryoshka dictionary starts broad and refines, while the Standard SAE starts narrow and generalizes (Fig. 4). The Matryoshka SAE gives that concept an address, holding it in the broad core at every depth while the standard SAE scatters the same function across arbitrary indices. This stable addressing allows us trace a single concept across depth.

We characterized only one parent–child pair, though the sweep returned many more. Since its child CONCEPT 3153 detects the strongest ARs without driving them, the full account of how GraphCast computes ARs lies in the circuit of related concepts rather than in any one of them. A model that computes a quantity it was never taught, is legible in a dictionary where scientists can locate it, and responds to intervention as the physics would suggest is no longer a black box but a system that can be audited as ARs intensify beyond its training record.

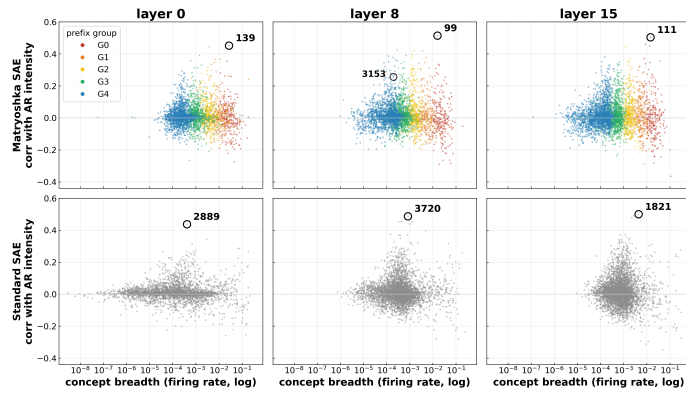


Figure 4: All 4,096 concepts from each SAE plotted by breadth (firing rate) vs. correlation with AR intensity; Matryoshka SAE (top) and Standard SAE (bottom) at layers 0, 8, 15. Colors correspond to Matryoshka prefix groups. Each panel’s top AR concept and child concept CONCEPT 3153 are labeled.

References

- [1] Zied Ben Bouallègue, Mariana C. A. Clare, Linus Magnusson, Estibaliz Gascon, Michael Maier-Gerber, Martin Janoušek, Mark Rodwell, Florian Pinault, Jesper S. Dramsch, Simon T. K. Lang, Baudouin Raoult, Florence Rabier, Matthew Chantry, Mihai Alexe, Peter Dueben, and Florian Pappenberger. The rise of data-driven weather forecasting: A first statistical assessment of machine learning-based weather forecasts in an operational-like context. *Bulletin of the American Meteorological Society*, 105(6):E864–E883, 2024. doi: 10.1175/BAMS-D-23-0162.1.
- [2] Stephan Rasp, Stephan Hoyer, Alexander Merose, Ian Langmore, Peter Battaglia, Tyler Russell, Alvaro Sanchez-Gonzalez, Vivian Yang, Rob Carver, Shreya Agrawal, Matthew Chantry, Zied Ben Bouallègue, Peter Dueben, Carla Bromberg, Jared Sisk, Luke Barrington, Aaron Bell, and Fei Sha. 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. doi: 10.1029/2023MS004019.
- [3] Remi Lam, Alvaro Sanchez-Gonzalez, Matthew Willson, Peter Wyrnsberger, Meire Fortunato, Ferran Alet, Suman Ravuri, Timo Ewalds, Zach Eaton-Rosen, Weihua Hu, Alexander Merose, Stephan Hoyer, George Holland, Oriol Vinyals, Jacklynn Stott, Alexander Pritzel, Shakir Mohamed, and Peter Battaglia. Learning skillful medium-range global weather forecasting. *Science*, 382(6677):1416–1421, 2023. doi: 10.1126/science.adi2336.
- [4] Kirsten J. Mayer, William E. Chapman, and William A. Manriquez. Exploring the relative importance of the MJO and ENSO to North Pacific subseasonal predictability. *Geophysical Research Letters*, 51(10):e2024GL108479, 2024. doi: 10.1029/2024GL108479.
- [5] Peter Bauer, Alan Thorpe, and Gilbert Brunet. The quiet revolution of numerical weather prediction. *Nature*, 525(7567):47–55, 2015. doi: 10.1038/nature14956.
- [6] Amy McGovern, Ryan Lagerquist, David John Gagne, G. Eli Jergensen, Kimberly L. Elmore, Cameron R. Homeyer, and Travis Smith. Making the black box more transparent: Understanding the physical implications of machine learning. *Bulletin of the American Meteorological Society*, 100(11):2175–2199, 2019. doi: 10.1175/BAMS-D-18-0195.1.
- [7] Imme Ebert-Uphoff and Kyle Hilburn. Evaluation, tuning, and interpretation of neural networks for working with images in meteorological applications. *Bulletin of the American Meteorological Society*, 101(12):E2149–E2170, 2020. doi: 10.1175/BAMS-D-20-0097.1.
- [8] Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*, 2017. doi: 10.48550/arXiv.1702.08608.
- [9] Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLoS ONE*, 10(7):e0130140, 2015. doi: 10.1371/journal.pone.0130140.
- [10] Benjamin A. Toms, Elizabeth A. Barnes, and Imme Ebert-Uphoff. Physically interpretable neural networks for the geosciences: Applications to Earth system variability. *Journal of Advances in Modeling Earth Systems*, 12(9):e2019MS002002, 2020. doi: 10.1029/2019MS002002.
- [11] Elizabeth A. Barnes, Benjamin Toms, James W. Hurrell, Imme Ebert-Uphoff, Chuck Anderson, and David Anderson. Indicator patterns of forced change learned by an artificial neural network. *Journal of Advances in Modeling Earth Systems*, 12(9):e2020MS002195, 2020. doi: 10.1029/2020MS002195.
- [12] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, volume 70 of *Proceedings of Machine Learning Research*, pages 3319–3328. PMLR, 2017.
- [13] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. SmoothGrad: Removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017. doi: 10.48550/arXiv.1706.03825.

- [14] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should I trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1135–1144, 2016. doi: 10.1145/2939672.2939778.
- [15] Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems 30 (NeurIPS)*, pages 4765–4774, 2017.
- [16] Theodore MacMillan and Nicholas T. Ouellette. Towards mechanistic understanding in a data-driven weather model: Internal activations reveal interpretable physical features. *arXiv preprint arXiv:2512.24440*, 2025. doi: 10.48550/arXiv.2512.24440.
- [17] Leo Gao, Tom Dupré la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. Scaling and evaluating sparse autoencoders. *arXiv preprint arXiv:2406.04093*, 2024. doi: 10.48550/arXiv.2406.04093.
- [18] Bart Bussmann, Patrick Leask, and Neel Nanda. Learning multi-level features with Matryoshka sparse autoencoders. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, Proceedings of Machine Learning Research. PMLR, 2025. arXiv:2503.17547.
- [19] Aditya Kusupati, Gantavya Bhatt, Aniket Rege, Matthew Wallingford, Aditya Sinha, Vivek Ramanujan, William Howard-Snyder, Kaifeng Chen, Sham Kakade, Prateek Jain, and Ali Farhadi. Matryoshka representation learning. In *Advances in Neural Information Processing Systems*, volume 35, pages 30233–30249, 2022.
- [20] Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Zac Hatfield-Dodds, Alex Tamkin, Karina Nguyen, Brayden McLean, Josiah E. Burke, Tristan Hume, Shan Carter, Tom Henighan, and Christopher Olah. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023. URL <https://transformer-circuits.pub/2023/monosemantic-features>.
- [21] Rakesh Agrawal, Tomasz Imieliński, and Arun Swami. Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, SIGMOD '93*, pages 207–216, Washington, D.C., USA, 1993. ACM. doi: 10.1145/170035.170072.
- [22] Bin Guan and Duane E. Waliser. Detection of atmospheric rivers: Evaluation and application of an algorithm for global studies. *Journal of Geophysical Research: Atmospheres*, 120(24): 12514–12535, 2015. doi: 10.1002/2015JD024257.
- [23] Duane Waliser and Bin Guan. Extreme winds and precipitation during landfall of atmospheric rivers. *Nature Geoscience*, 10(3):179–183, 2017. doi: 10.1038/ngeo2894.
- [24] Jonathan J. Rutz, Christine A. Shields, Juan M. Lora, Ashley E. Payne, Bin Guan, Paul Ullrich, Travis O'Brien, L. Ruby Leung, F. Martin Ralph, Michael Wehner, Swen Brands, Allison Collow, Naomi Goldenson, Irina Gorodetskaya, Helen Griffith, Karthik Kashinath, Brian Kawzenuk, Harinarayan Krishnan, Vitaliy Kurlin, David Lavers, Gudrun Magnusdottir, Kelly Mahoney, Elizabeth McClenny, Grzegorz Muszynski, Phu Dinh Nguyen, Mr Prabhat, Yun Qian, Alexandre M. Ramos, Chandan Sarangi, Scott Sellars, Tereza Shulgina, Ricardo Tome, Duane Waliser, Daniel Walton, Gary Wick, Anna M. Wilson, and Maximiliano Viale. The Atmospheric River Tracking Method Intercomparison Project (ARTMIP): Quantifying uncertainties in atmospheric river climatology. *Journal of Geophysical Research: Atmospheres*, 124(24):13777–13802, 2019. doi: 10.1029/2019JD030936.
- [25] F. Martin Ralph, Jonathan J. Rutz, Jason M. Cordeira, Michael Dettinger, Michael Anderson, David Reynolds, Lawrence J. Schick, and Chris Smallcomb. A scale to characterize the strength and impacts of atmospheric rivers. *Bulletin of the American Meteorological Society*, 100(2): 269–289, 2019. doi: 10.1175/BAMS-D-18-0023.1.
- [26] David Chanin, James Wilken-Smith, Tomáš Dulka, Hardik Bhatnagar, and Joseph Bloom. A is for absorption: Studying feature splitting and absorption in sparse autoencoders. *arXiv preprint arXiv:2409.14507*, 2024. doi: 10.48550/arXiv.2409.14507.

Hyperparameter	Standard Top- K	Matryoshka Top- K
Input dimension d	512	512
Dictionary size n	4096	4096
Active latents k	32 (per sample)	32 (per sample)
Auxiliary latents (AuxK)	512	512
Input normalization	mean-center, ℓ_2 row	min-max $[-1, 1]$, running-avg
Batch size	8192	4096
Training length	10 epochs	3×10^5 steps
Optimizer	Adam ($\beta=0.9, 0.999; \epsilon=6.25 \times 10^{-10}$)	Adam ($\beta=0.5, 0.9375$)
Learning rate	2×10^{-4}	2×10^{-4}

Table 1: **Training hyperparameters.** All three depths of a given architecture use identical settings; only the input activation record differs. Both architectures use per-sample Top- K selection and an AuxK dead-latent revival term.

A SAE Training

All six SAEs share the same width and sparsity budget and differ only in input normalization, reconstruction objective, and optimization schedule (Table 1). Each maps a 512-dimensional GraphCast mesh activation to a dictionary of 4,096 latents and keeps the $k = 32$ largest per-sample pre-activations, so sparsity is fixed by construction and no ℓ_1 penalty is used. Both architectures add an auxiliary Top- K term (AuxK) that routes reconstruction error through the 512 least-recently-active latents to revive dead latents. Optimization uses Adam at a learning rate of 2×10^{-4} in every run.

The two architectures are trained on different schedules because their objectives and input normalizations differ. The Matryoshka loss averages reconstruction error over five nested prefixes, so each update carries gradient signal from five reconstructions rather than one, and its min-max input scaling and running-norm rescaling place activations in a different range from the Standard SAE’s unit-normalized rows. Batch size, optimizer momentum, and schedule length were set separately for each so that each objective trains stably. As a result, the Standard SAE takes roughly 2.85×10^6 updates over 10 epochs, while the Matryoshka SAE takes 3×10^5 updates covering about half an epoch. We treat each schedule as sufficient rather than matched: in both architectures the training loss plateaus well before the schedule ends, so additional updates would not materially change either dictionary.

Figure 5 shows the reconstruction loss over training for all six SAEs, each in its own operating space. Each panel reports its architecture’s native training loss. The Standard SAE is trained on unit-normalized activations and the Matryoshka SAE on min-max-scaled activations, so absolute loss values are not comparable across panels; only the trend within a panel is meaningful. The active-latent count remains fixed at $k=32$ throughout Matryoshka training, and the slight late rise in its loss reflects optimization under a constant learning rate against a running input normalizer rather than any loss of reconstruction quality.

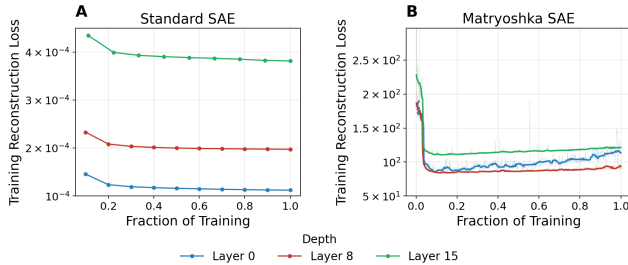


Figure 5: Training reconstruction loss for the six SAEs against fraction of training, one panel per architecture, colored by depth. Each loss is measured in that SAE’s own operating space, so the panels are not comparable in absolute value. The Matryoshka SAE logs every 500 steps, so its curve is drawn smoothed (bold) over the raw per-step values (faint) to show the trend.

B Matryoshka Nesting

As this is the first work to explore Matryoshka nesting in a GNN setting, we provide definitions that specify spatiotemporal relationships between concepts. Any two latents active somewhere on nearly every timestep will trivially co-fire at the timestep level, so timestep co-occurrence cannot distinguish nesting. Such latents are nonetheless active at only a small fraction of the nodes at any given moment. We therefore ask not whether parent and child fire on the same input, but whether they fire at the same nodes on that input. We take every (node, timestep) at which the child fires and ask how often the parent is firing there too. We call this fraction c the child’s *spatial containment* in the parent. A latent is counted as firing at a node when its activation exceeds zero, *i.e.*, it is one of the top- k active latents at that node. Spatial containment is the *confidence* of the association rule “child $\Rightarrow$ parent” [21], the set-inclusion coefficient

$$c = \frac{|\mathcal{F}^{\text{ch}} \cap \mathcal{F}^{\text{par}}|}{|\mathcal{F}^{\text{ch}}|},$$

where $\mathcal{F}^{\text{ch}}$ and $\mathcal{F}^{\text{par}}$ are the sets of (node, timestep) cells at which the child and parent fire. Dividing by the child’s own firing count is what makes the measure one-directional: the parent may fire at many nodes the child never touches without lowering c . Treating each of the child’s firing events as a Bernoulli trial, we report c as a proportion with a 95% Wilson score interval. This relation is one we measure, not one the architecture imposes: the Matryoshka objective orders latents into nested groups but never binds a later latent to a unique earlier one, so containment induces a partial order rather than a hierarchy. A latent may nest under zero, one, or several parents.

We form every candidate pair by taking each latent in the two innermost groups (G0 and G1, latents 0–511) with every later-group latent (1,835,008 pairs) and compute c . For the overwhelming majority containment is essentially zero (median 0.003, with 90% of pairs below 0.083). Every one of these 1.8 million unrelated pairs is in effect a random pairing, so the sweep is its own negative control. Genuine nesting is rare: just 7,995 pairs (0.44%) exceed $c = 0.5$ and only 1,436 (0.078%) reach $c = 0.9$. These pairs are asymmetric in the way nesting requires: the parent is nearly always present where the child fires (high c), yet the child covers only a few percent of the parent’s own footprint. The two form a part and a whole rather than feature splitting, which would be two copies of the same concept [26].

We call a later-group latent *nested* in an earlier-group parent when the lower end of its 95% Wilson interval exceeds 0.5, so the child fires inside the parent more often than not; *strongly nested* when the lower end of its 95% Wilson interval exceeds 0.9; and *exclusively nested* when it is strongly nested in that parent *and in no other latent in G0 or G1*. As containment is only a partial order, strong nesting need not be exclusive: a narrow latent lying inside the overlap of two broad concepts is strongly contained in both. The pair examined in Section 3.1 is both strong and exclusive.

C Activation Steering

Both the Standard SAE and Matryoshka SAE normalize model activations prior to SAE encoding. Each architecture does this slightly differently. The Standard SAE normalizes each activation vector separately. It centers it at the mean and rescales it to unit length, so the normalization factor is that vector’s own magnitude and is different for every activation. Because this factor is known only during the forecast, the edit is converted back to the activation’s original units, by multiplying by that same magnitude, inside the forward pass so the model can continue. The Matryoshka SAE instead normalizes each of the 512 activation dimensions by its fixed range over the training period, mapping each dimension’s minimum to -1 and maximum to $+1$. Since those ranges are fixed constants, converting the edit back to original units uses a fixed factor, $m^{\text{max}} - m^{\text{min}}$, so the whole edit can be precomputed.

With the understanding of how each normalization works, we can write a concept edit. A concept c contributes a fixed pattern to a node’s activation vector x : its activation strength z_c , the code value the SAE assigns the concept, times its decoder direction $W_{\text{dec}}[c]$, the pattern in activation space the concept stands for. To intervene, we change the concept’s code by an amount δz_c and add the decoded change $\delta z_c W_{\text{dec}}[c]$ back into the activation, either scaling an already-active concept to β times its value or writing a chosen level b into a silent one. Because $W_{\text{dec}}[c]$ is expressed in the

SAE's normalized space, we convert the change to the activation's raw units with the same factor that undoes the normalization.

For the Standard SAE that factor is the node's own magnitude, the length $\|x - \bar{x}\|$ of its mean-centered activation, where $\bar{x}$ is the mean over the 512 dimensions:

$$\Delta x_{sae} = \delta z_c W_{\text{dec}}[c] \|x - \bar{x}\|, \quad \delta z_c = \begin{cases} (\beta - 1) z_c & \text{scale an active concept by } \beta \\ b & \text{write a value } b \text{ into a concept} \end{cases} \quad (1)$$

Since $\|x - \bar{x}\|$ is known only once the forecast produces the activation, this edit is applied inside the forward pass. For the Matryoshka SAE the factor is a fixed constant set by the per-dimension bounds $m^{\min}$ and $m^{\max}$, the smallest and largest value each dimension takes over training, and the scalar $s = \sqrt{d}/\bar{n}$, where $d = 512$ is the activation dimension and $\bar{n}$ the running norm:

$$\Delta x_{matry} = \delta z_c W_{\text{dec}}[c] \frac{m^{\max} - m^{\min}}{2s}, \quad \delta z_c = \begin{cases} (\beta - 1) z_c & \text{scale an active concept by } \beta \\ b & \text{write a value } b \text{ into a concept} \end{cases} \quad (2)$$

so the whole edit is a fixed field, computed once over the training period and added to the activations during the forecast.