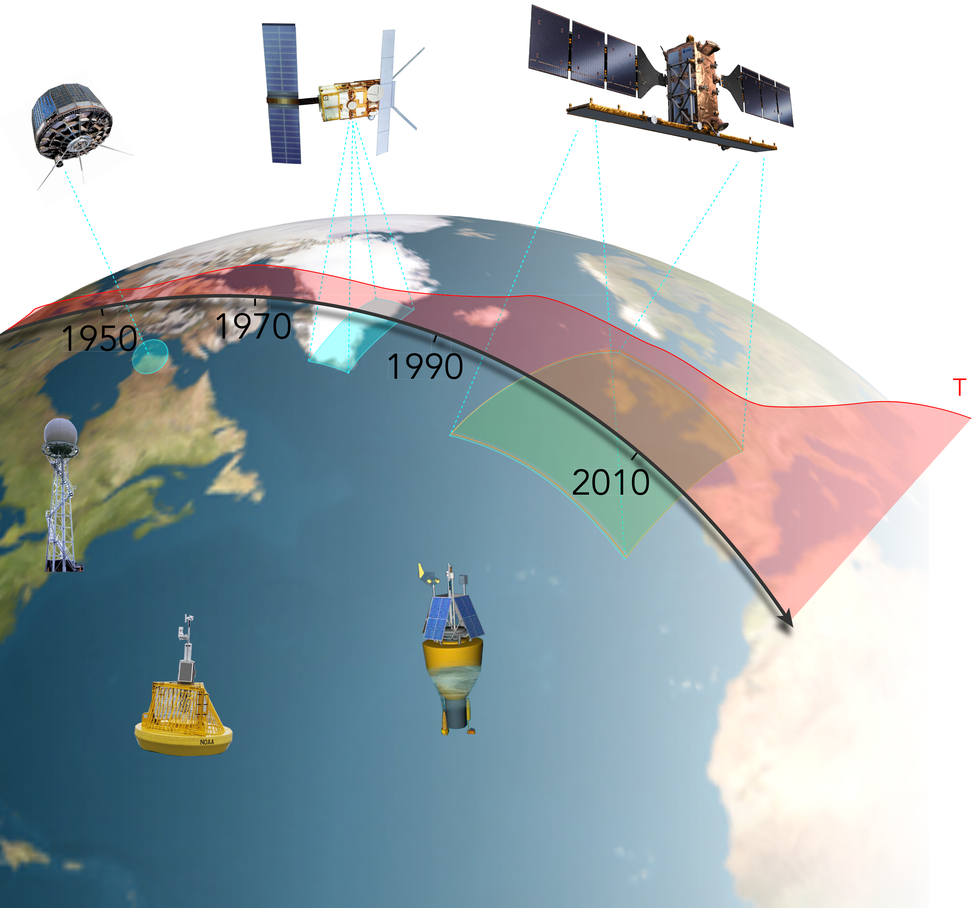
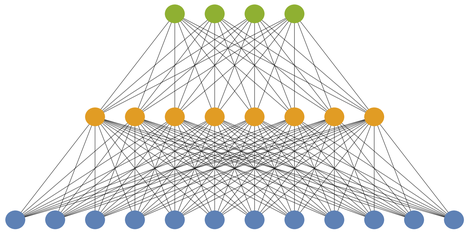


AtmoRep

Large scale representation learning of atmospheric dynamics

Christan Lessig, Ilaria Luise, Martin Schultz,
Maike Sonnewald, Annesh Subramanian



address climate change

large scale machine learning

scientific insight

Representation learning

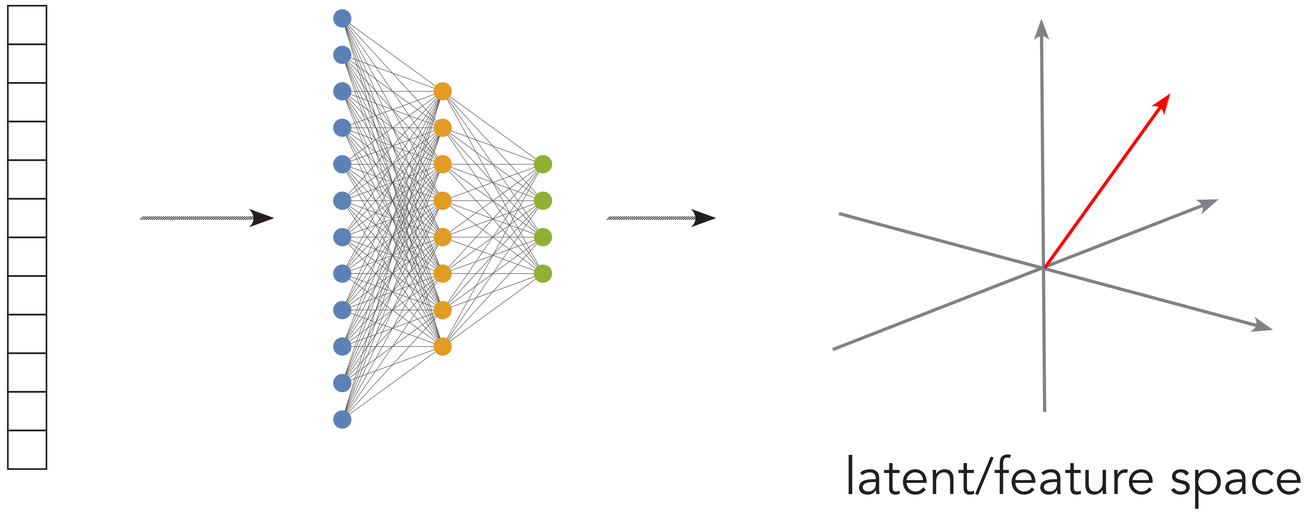
- Learn a domain-specific but task-independent neural network that is useful for a range of applications

Representation learning

- Learn a **domain-specific** but **task-independent** neural network that is useful for a range of applications

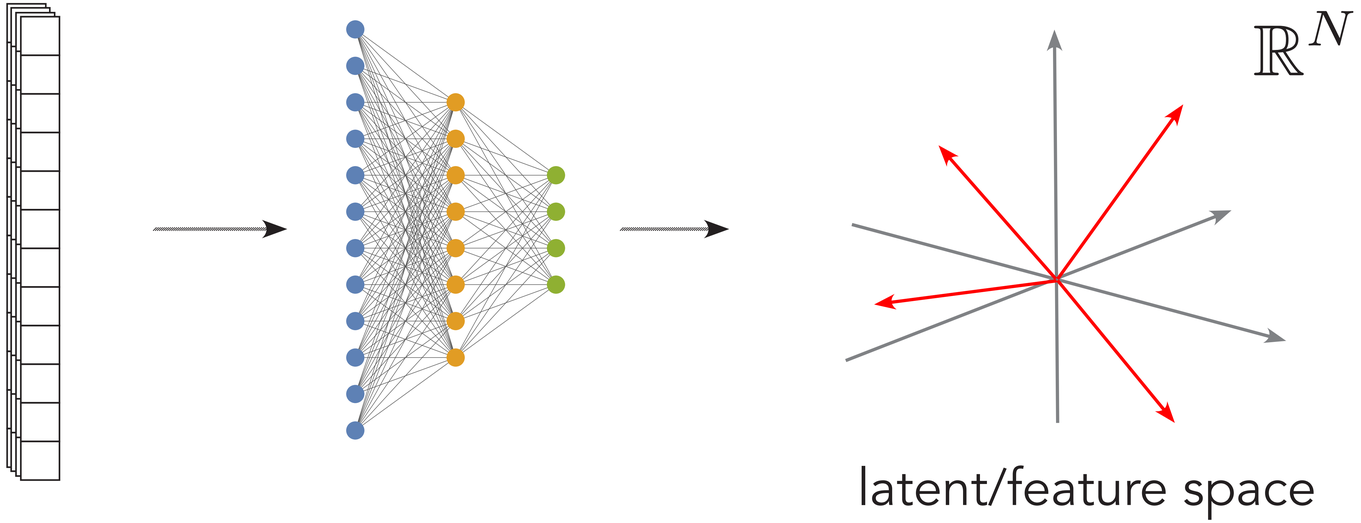
Representation learning

- Learn a **domain-specific** but **task-independent** neural network that is useful for a range of applications



Representation learning

- Learn a **domain-specific** but **task-independent** neural network that is useful for a range of applications



Representation learning

- Learn a **domain-specific** but **task-independent** neural network that is useful for a range of applications
  - Representation network provides transformation of network input to an effective feature space

Representation learning

- Learn a **domain-specific** but **task-independent** neural network that is useful for a range of applications
  - Representation network provides transformation of network input to an effective feature space
  - Self-supervised training on very large amounts of data (O(PB)) with very large networks (O(10^9) parameters)

Representation learning

- Learn a **domain-specific** but **task-independent** neural network that is useful for a range of applications
  - Representation network provides transformation of network input to an effective feature space
  - Self-supervised training on very large amounts of data (O(PB)) with very large networks (O(10^9) parameters)
  - Useful for applications using tail network or fine-tuning

Can we perform representation learning in the Earth sciences?

Representation learning for the Earth sciences?

- Very large amounts of observational data
  - ERA5 reanalysis: 6+ PB
  - E.g. MetOp-SG: 8 x 864 GB/day

Representation learning for the Earth sciences?

- Very large amounts of **observational** data
  - ERA5 reanalysis: 6+ PB
  - E.g. MetOp-SG: 8 x 864 GB/day

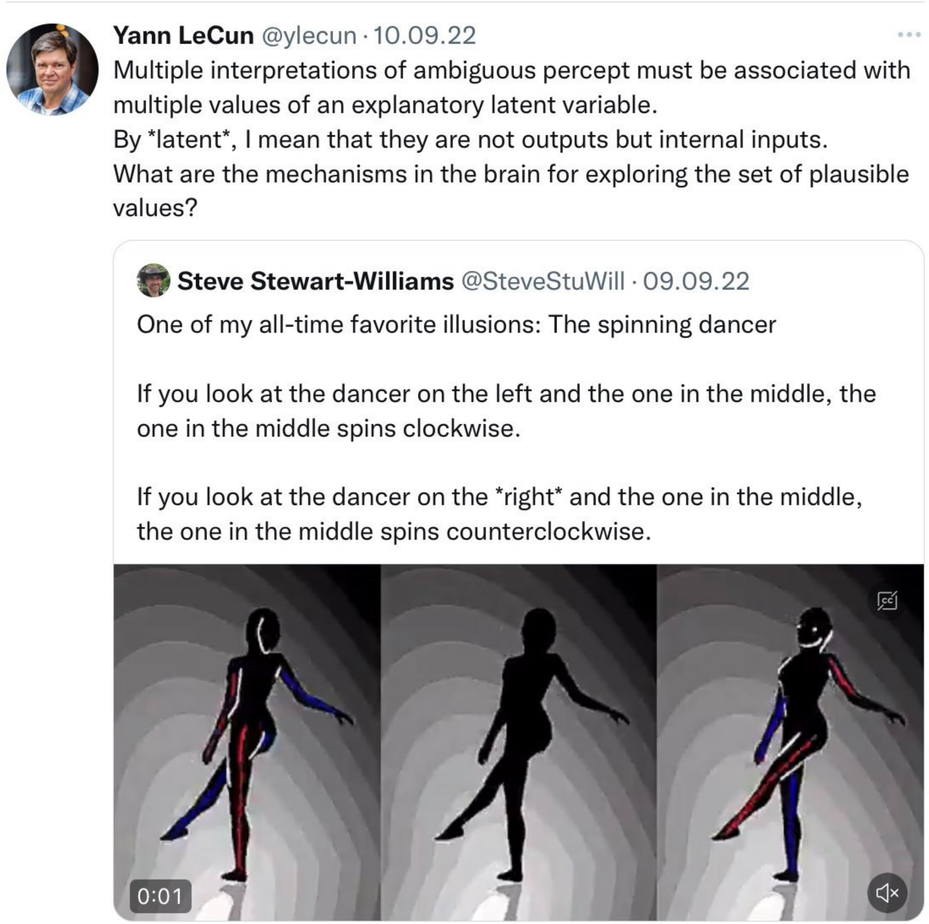
Representation learning for the Earth sciences?

- Very large amounts of **observational** data
- No complete classical model for system and dynamics
  - Central issue for forecasting and climate projections

Representation learning for the Earth sciences?

- Very large amounts of **observational** data
- No complete classical model for system and dynamics
- Large networks learn statistical representations

Representation learning for Earth science?



Representation learning for the Earth sciences?

- Very large amounts of **observational** data
- No complete classical model for system and dynamics
- Large networks learn statistical representations
  - Chaosticisty in atmospheric dynamics leads to ambiguity, i.e. there is often not one “correct answer“

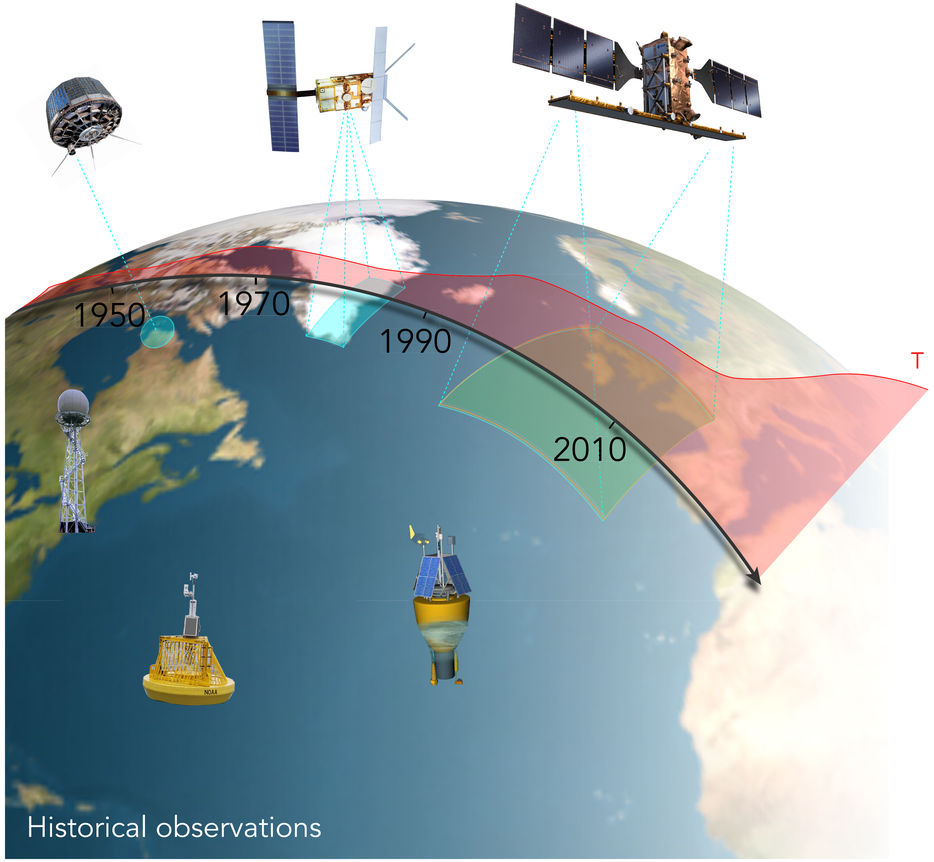
Representation learning for the Earth sciences?

- Very large amounts of **observational** data
- No complete classical model for system and dynamics
- Large networks learn statistical representations
- Data transformations like PCA have proven very useful

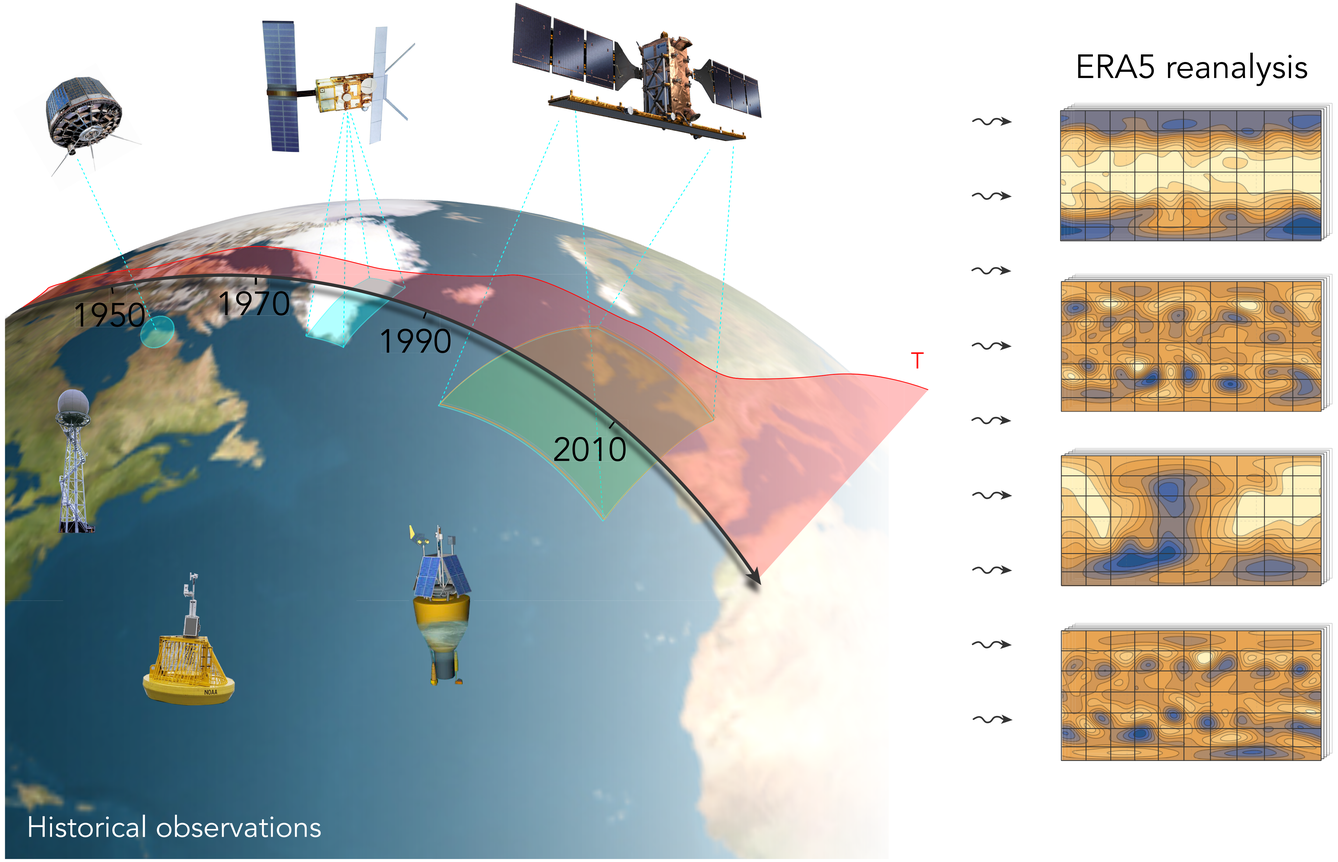
AtmoRep

Large scale representation learning of atmospheric dynamics

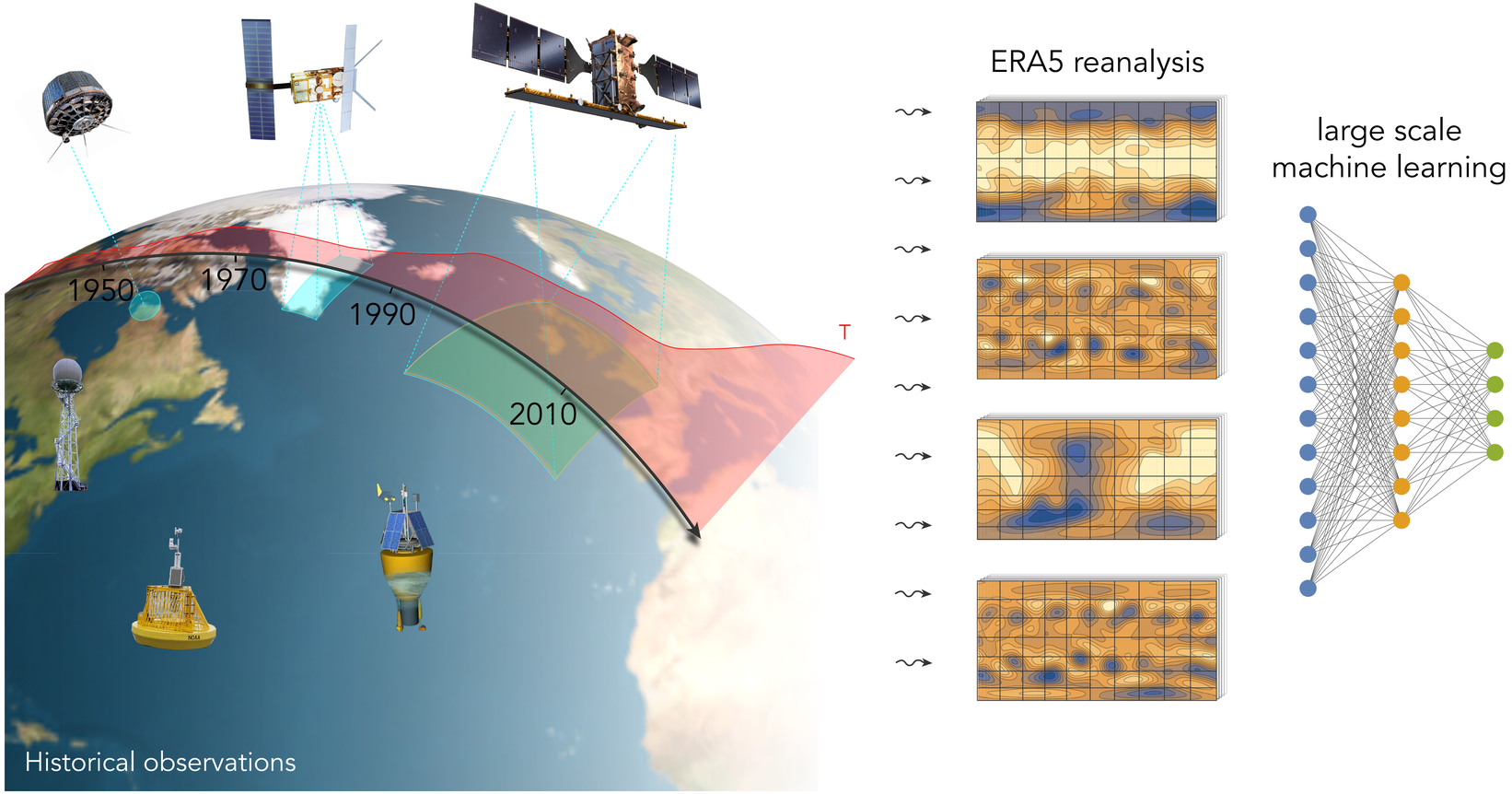
AtmoRep



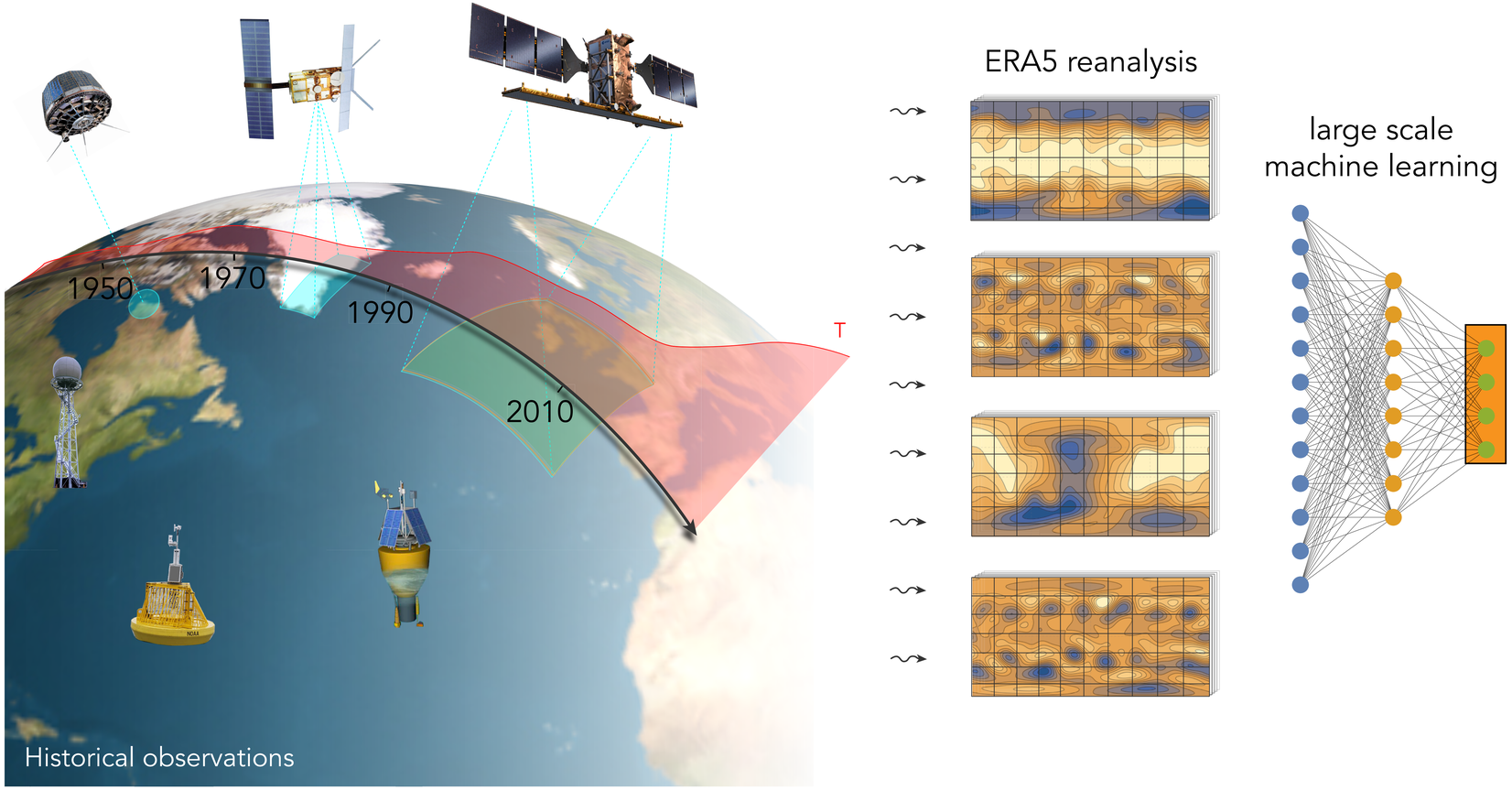
AtmoRep



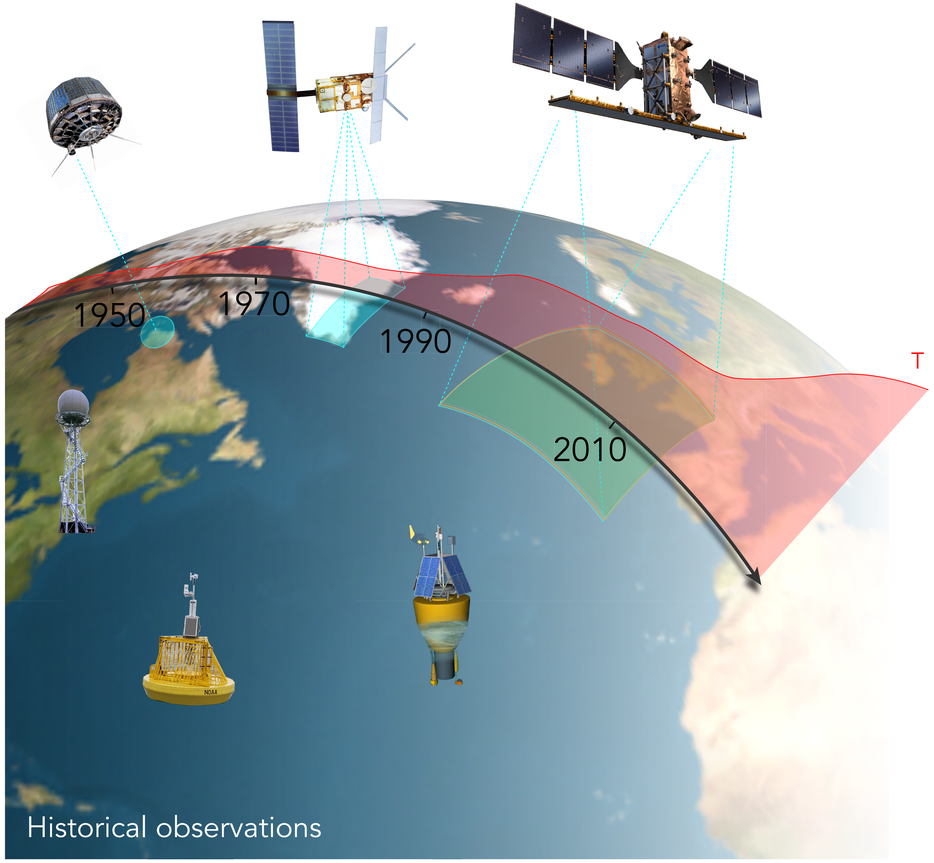
AtmoRep

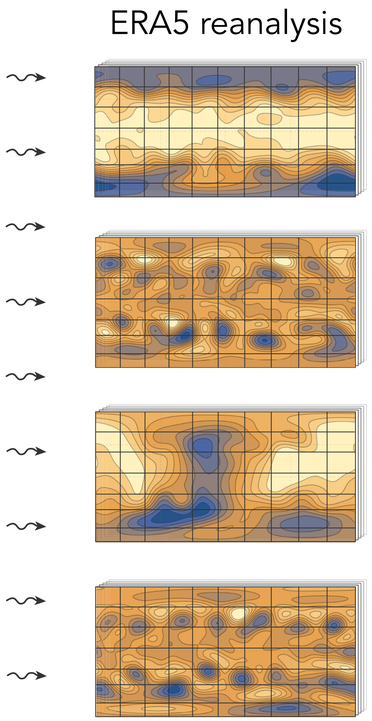


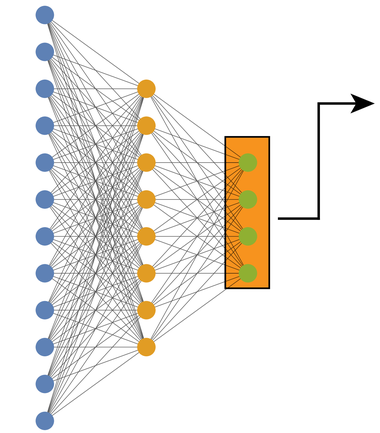
AtmoRep

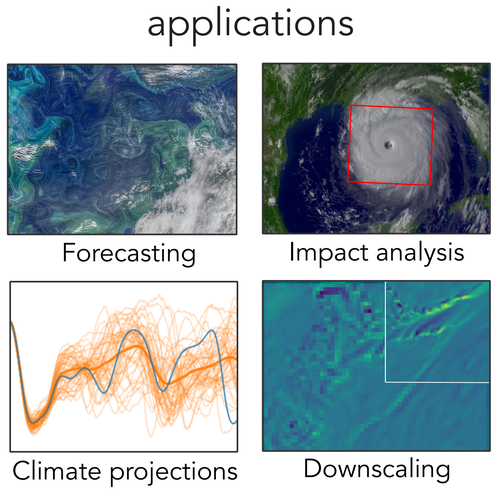


AtmoRep

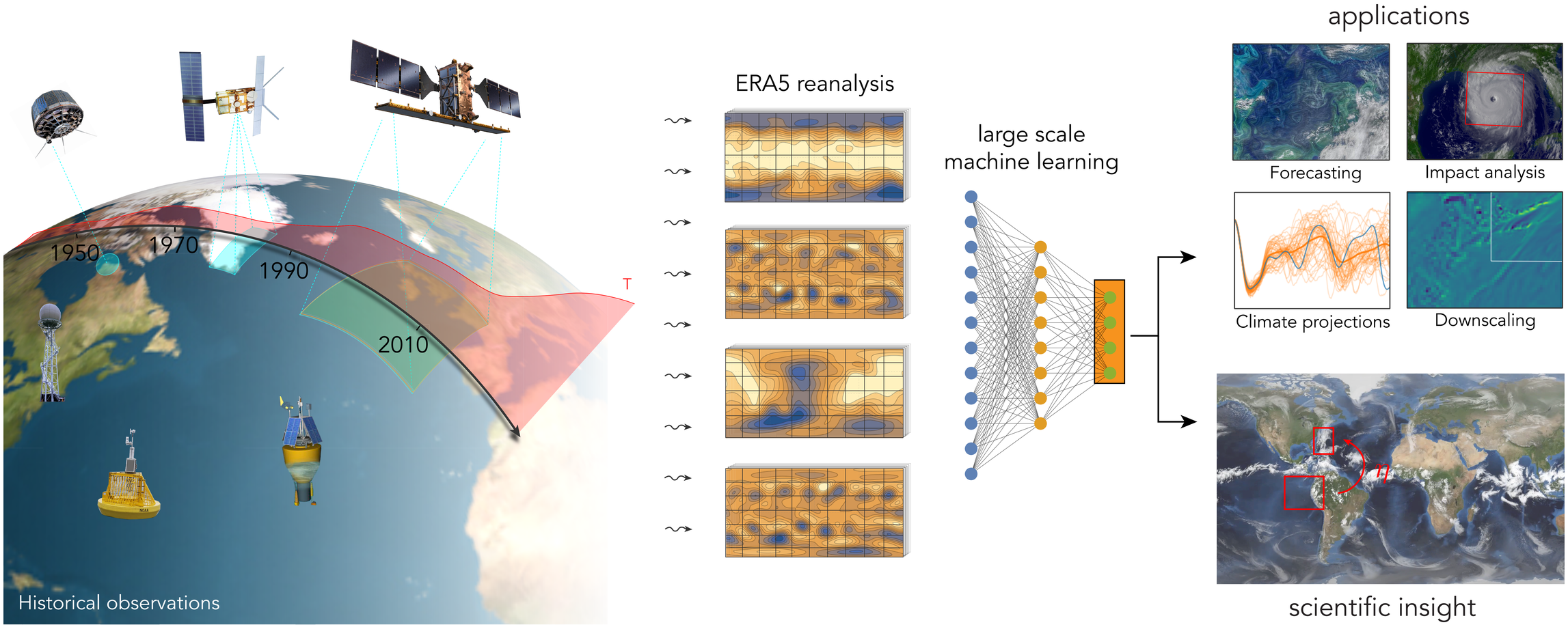




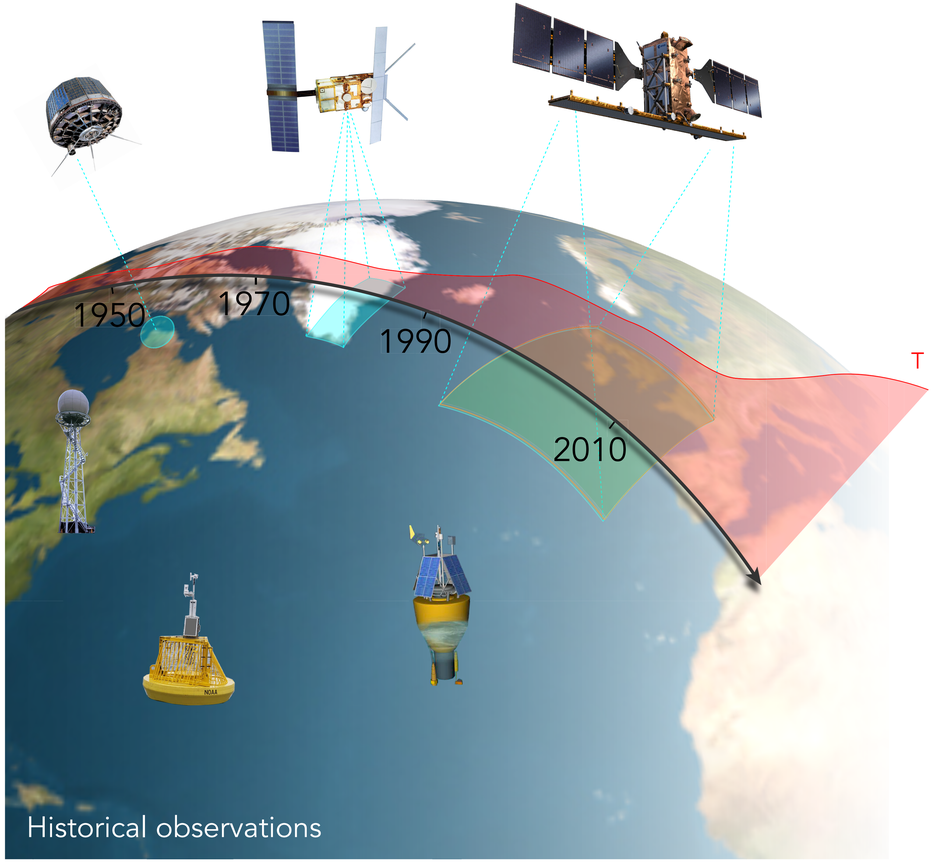


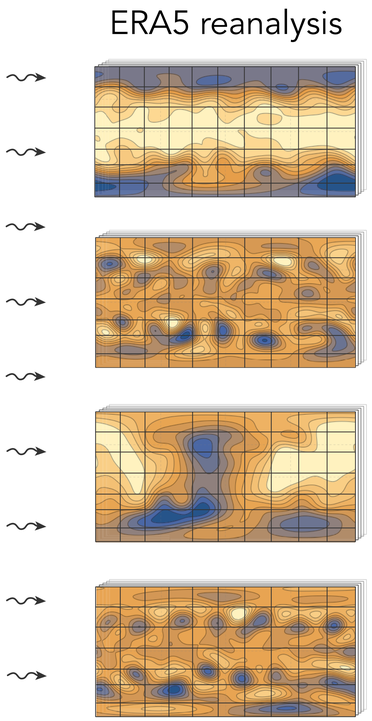


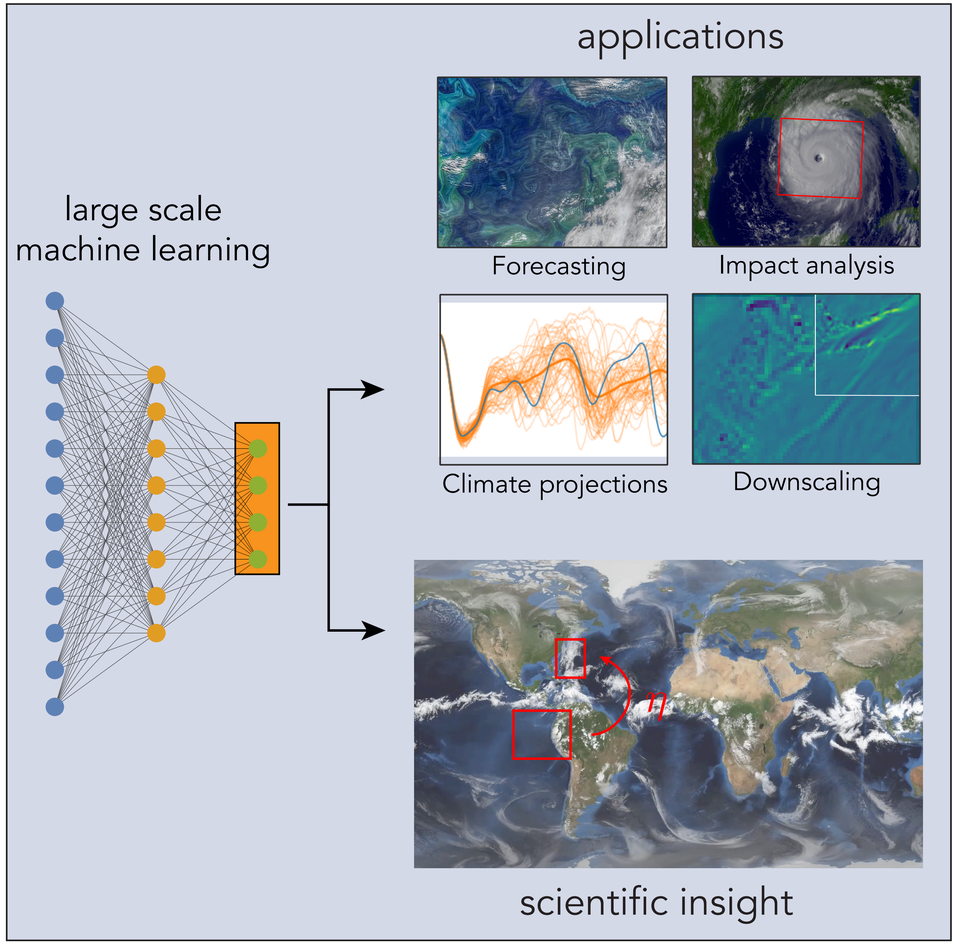
AtmoRep



AtmoRep

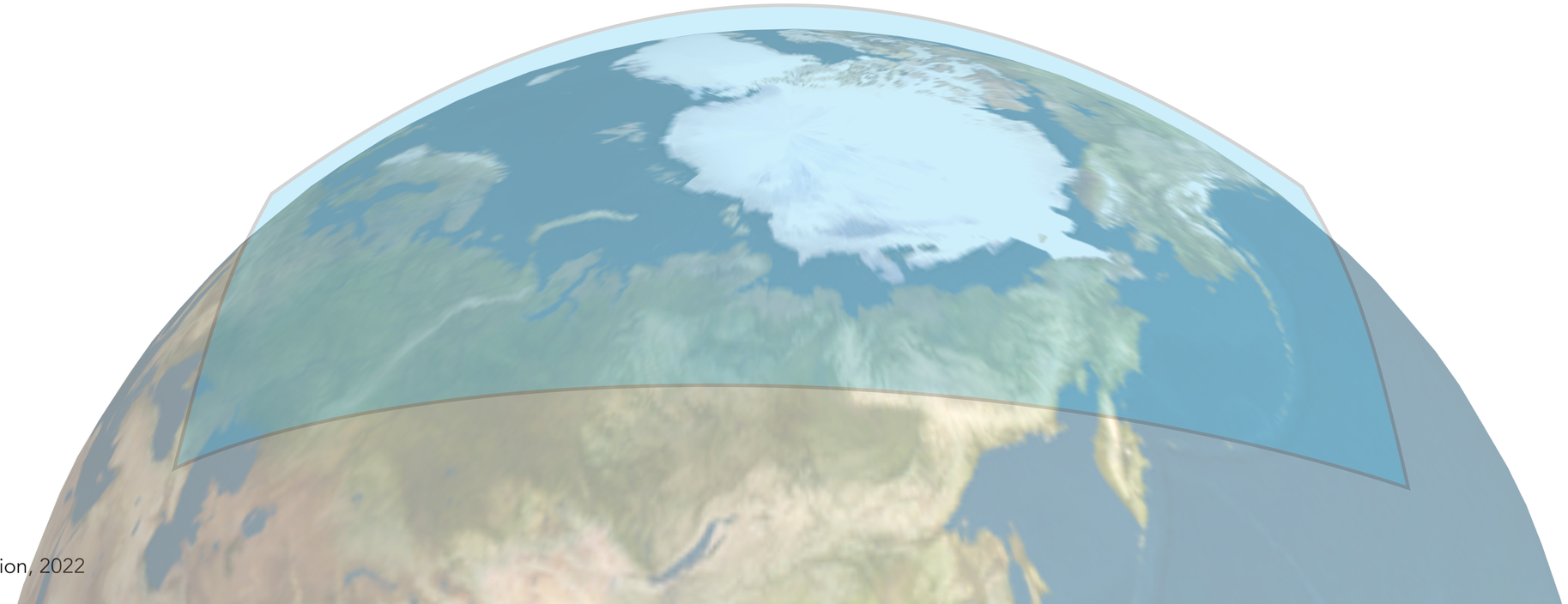




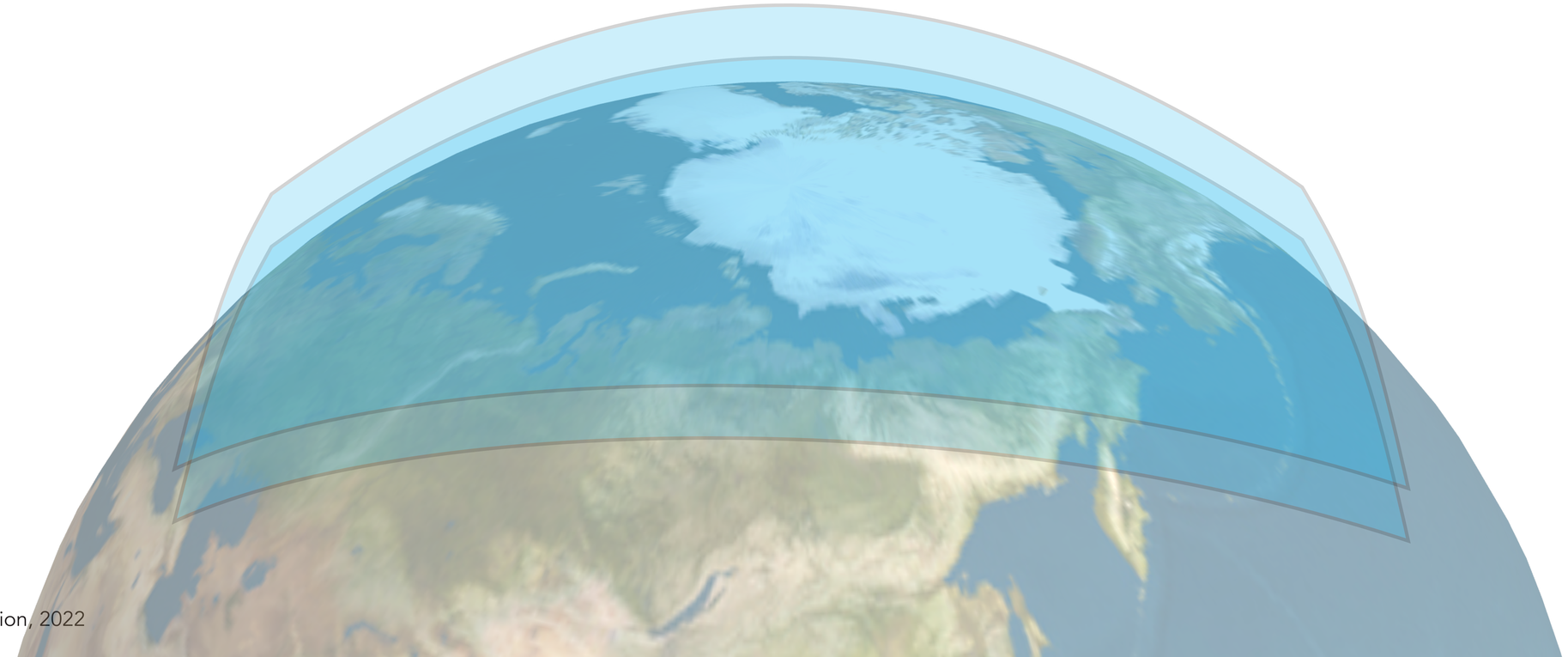


Data: ERA5 reanalysis

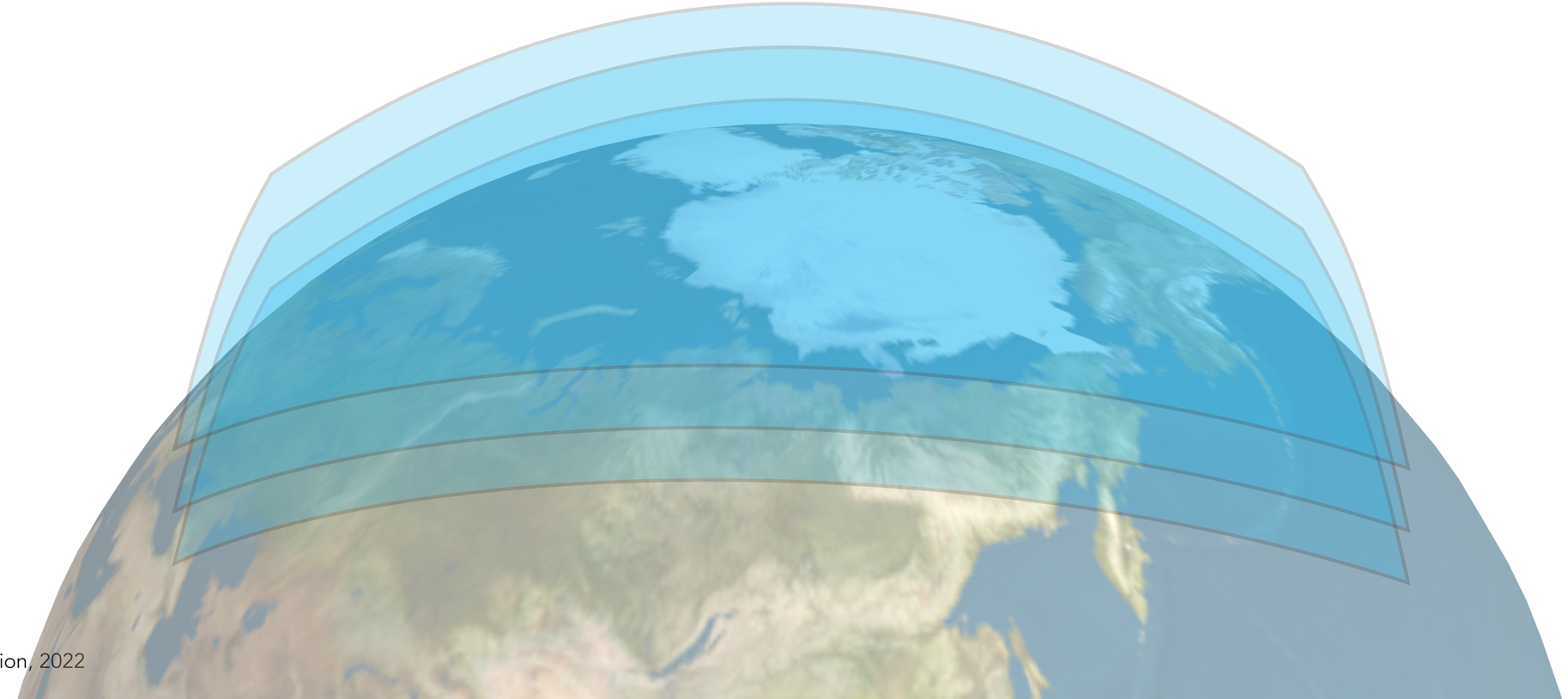
Data: ERA5 reanalysis



Data: ERA5 reanalysis

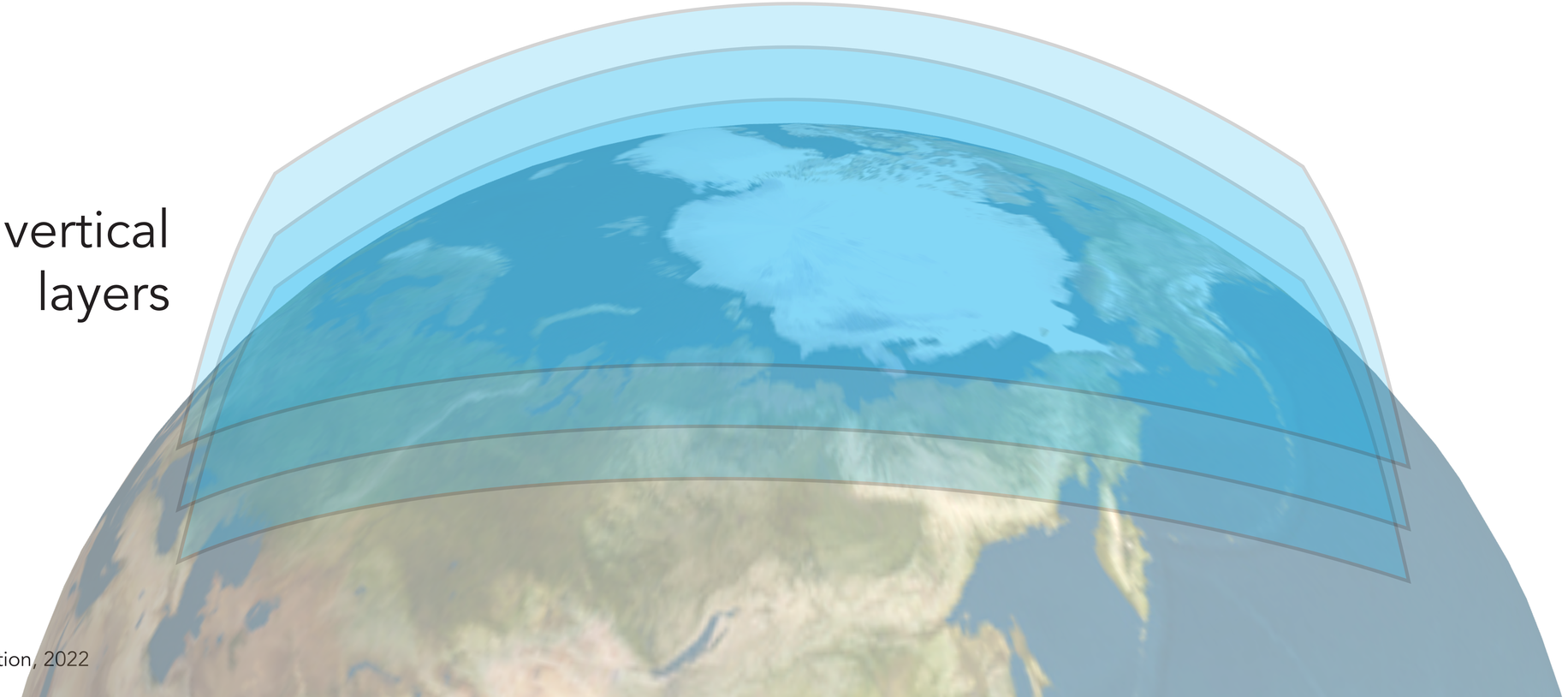


Data: ERA5 reanalysis



Data: ERA5 reanalysis

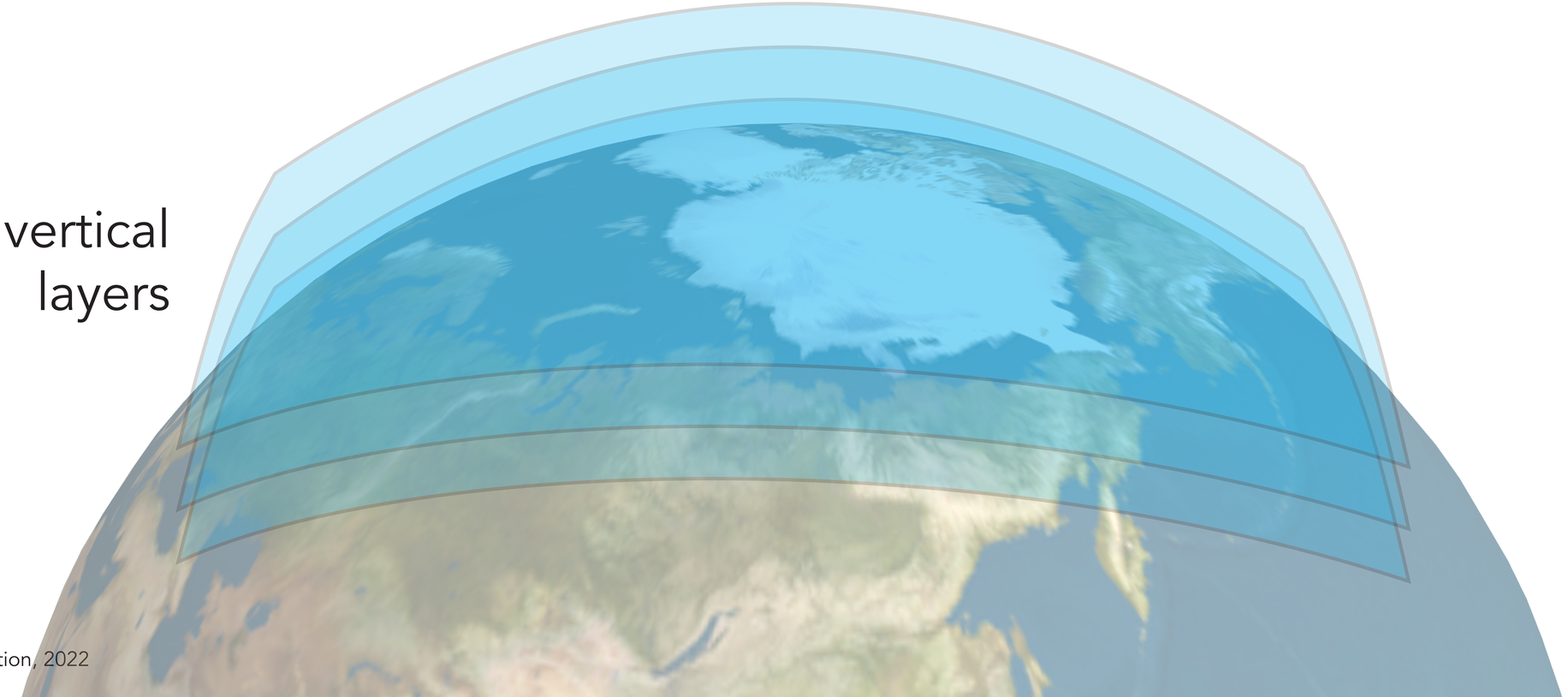
137 vertical layers



Data: ERA5 reanalysis

721x1440 horizontal grid (0.25 degree)

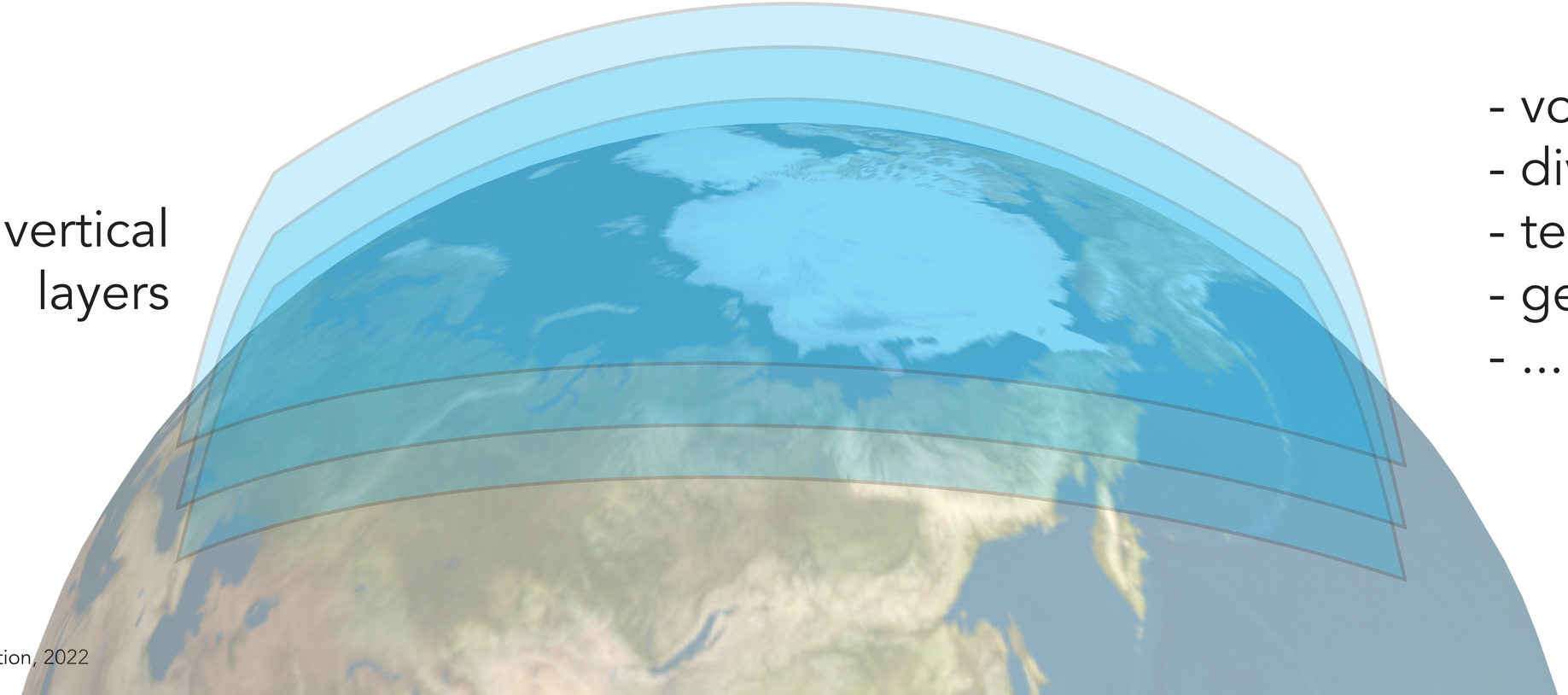
137 vertical layers



Data: ERA5 reanalysis

721x1440 horizontal grid (0.25 degree)

137 vertical layers

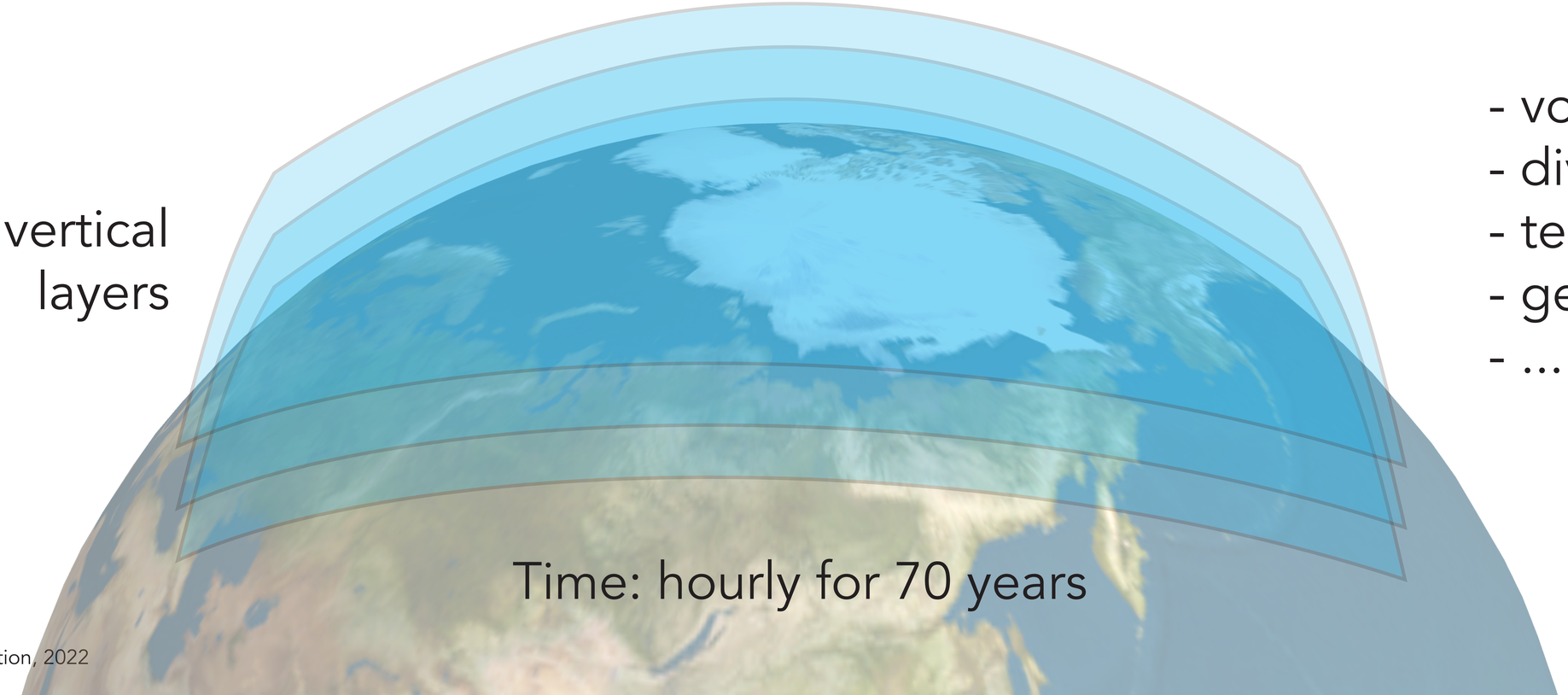


- vorticity
- divergence
- temperature
- geopotential
- ...

Data: ERA5 reanalysis

721x1440 horizontal grid (0.25 degree)

137 vertical layers



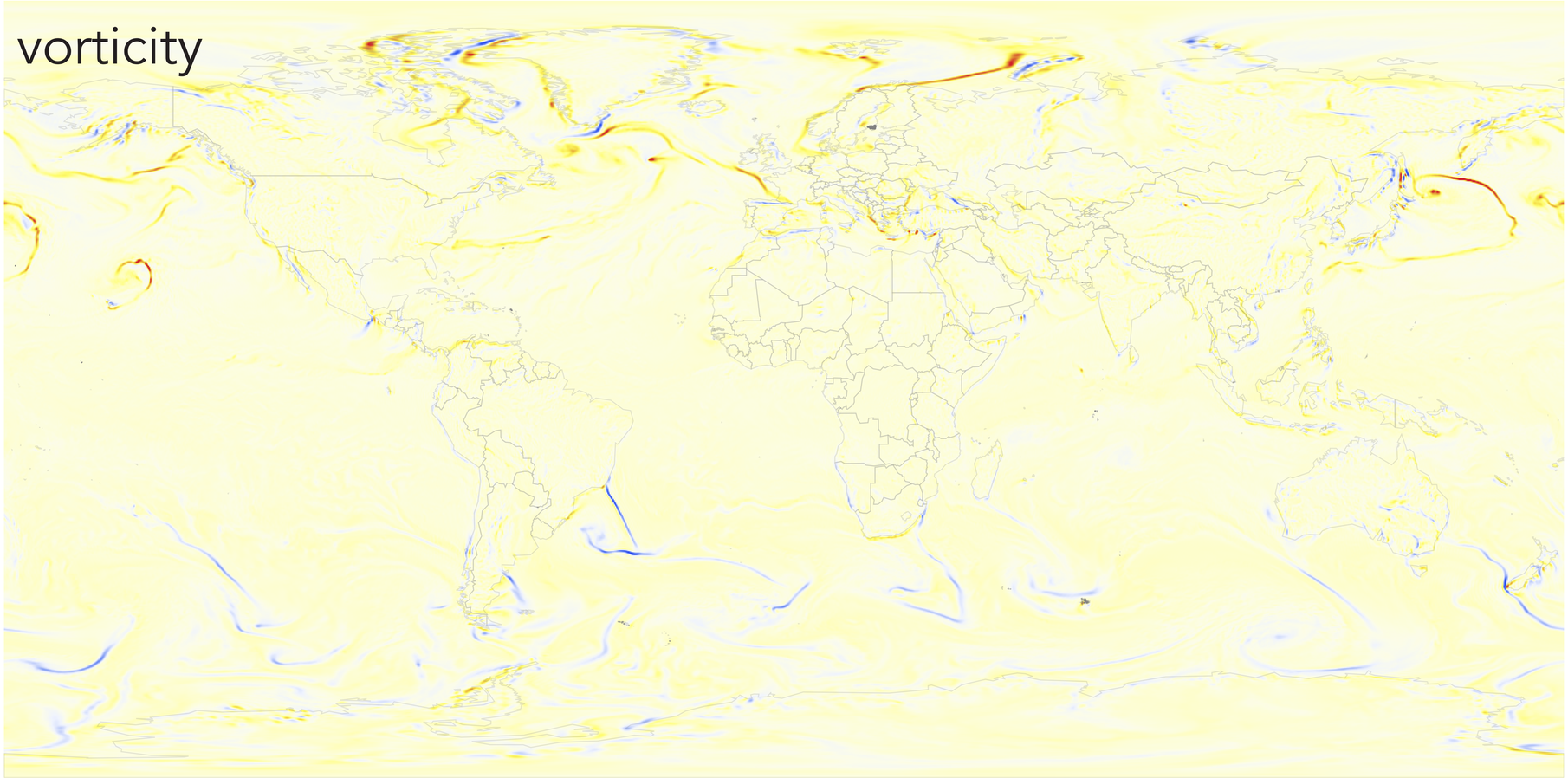
- vorticity
- divergence
- temperature
- geopotential
- ...

Time: hourly for 70 years

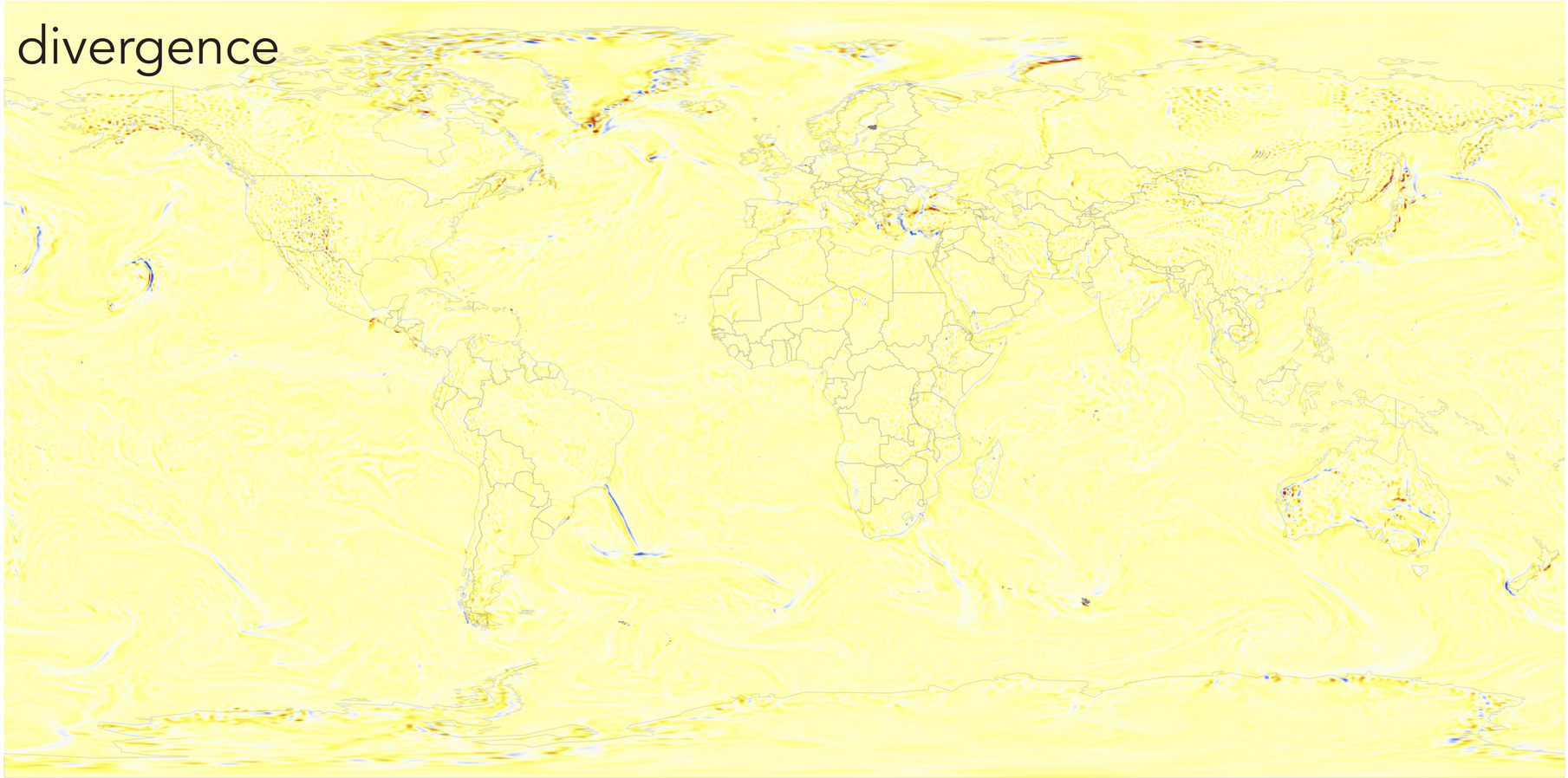
AtmoRep data

- Physical fields: vorticity, divergence, temperature, geo-potential height, humidity
- Space: 721 x 1440 x 10 vertical layers
- Time: 24 time steps per day for 365 days for 70 years

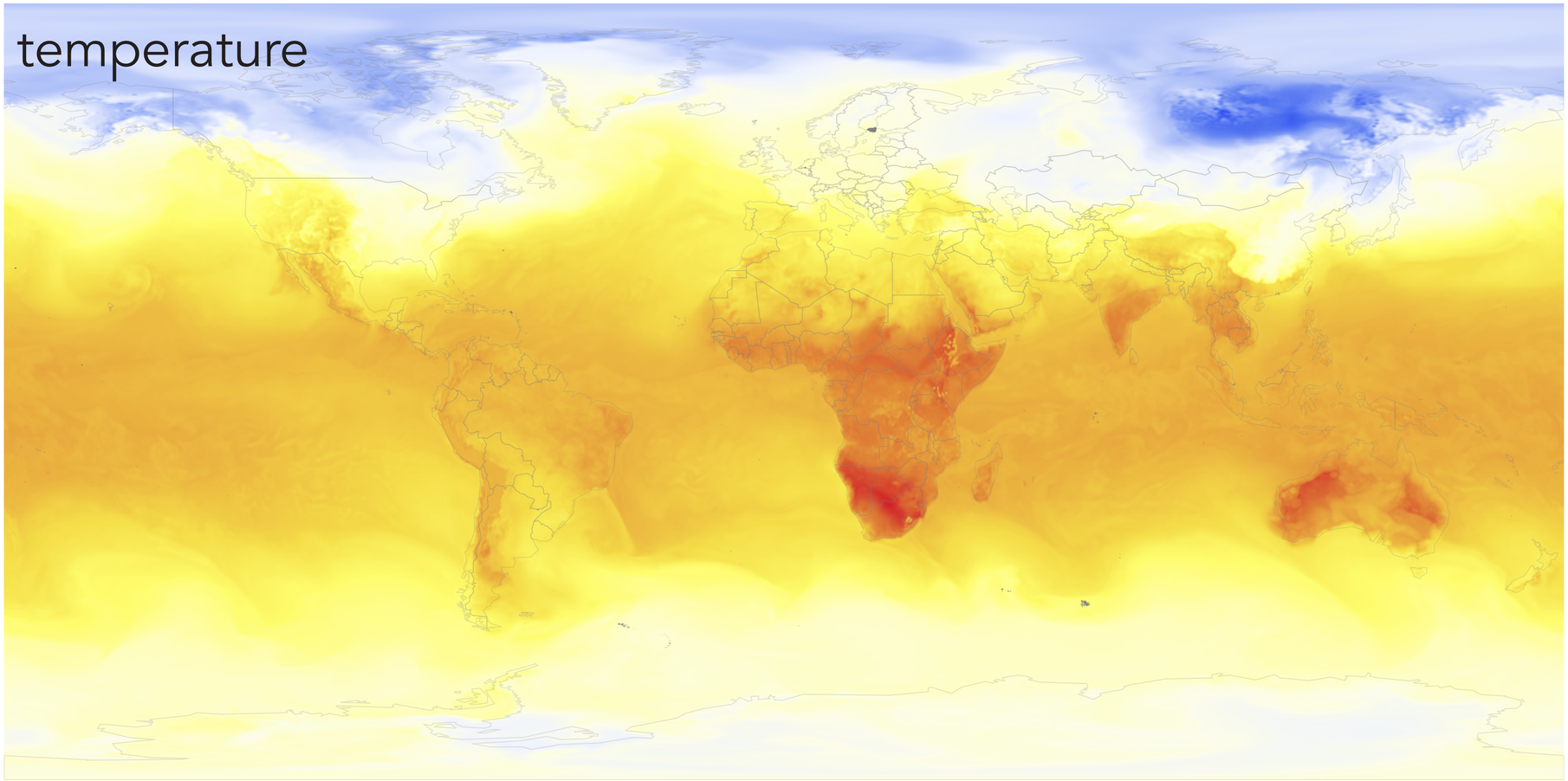
AtmoRep data



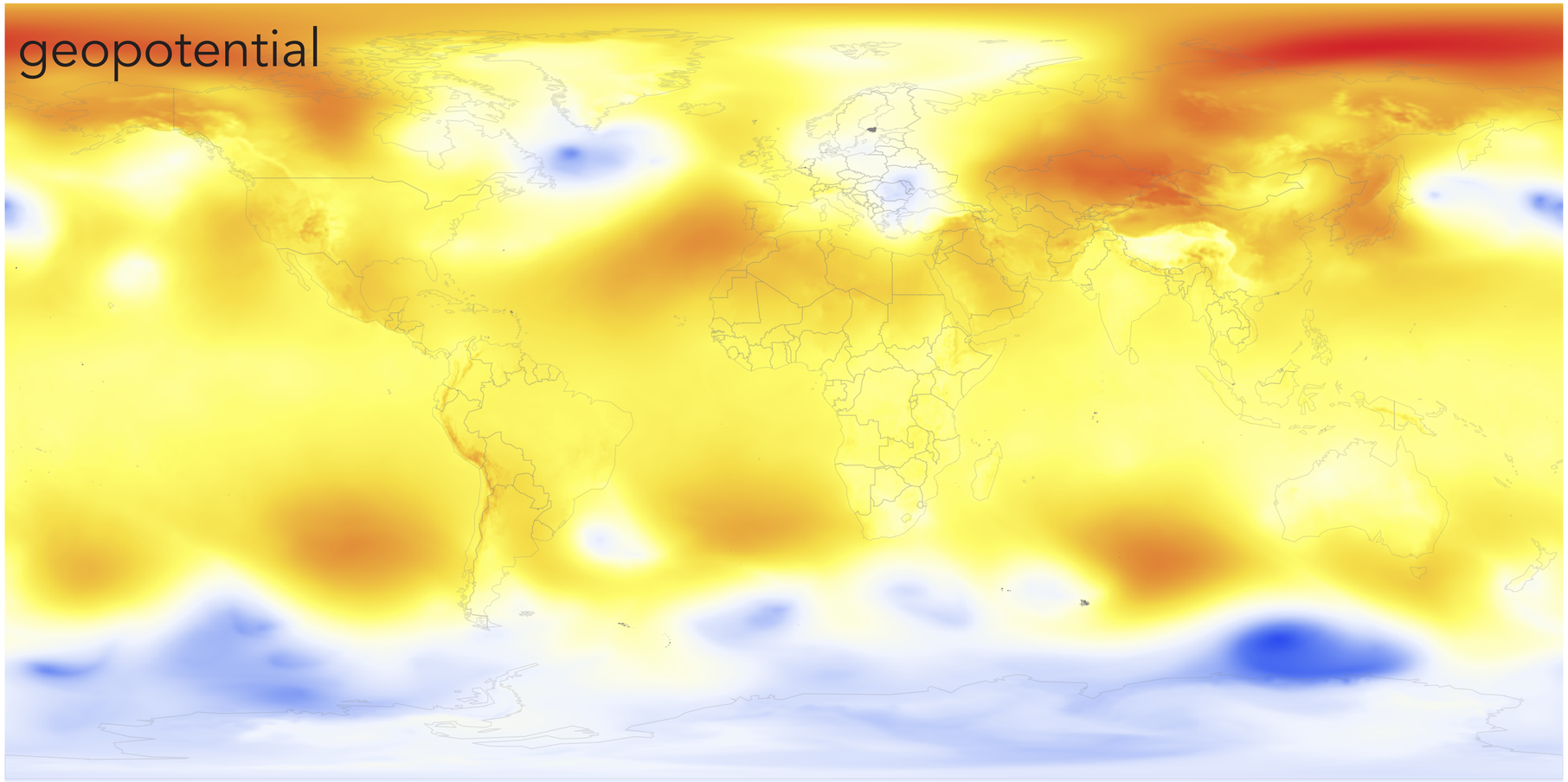
AtmoRep data



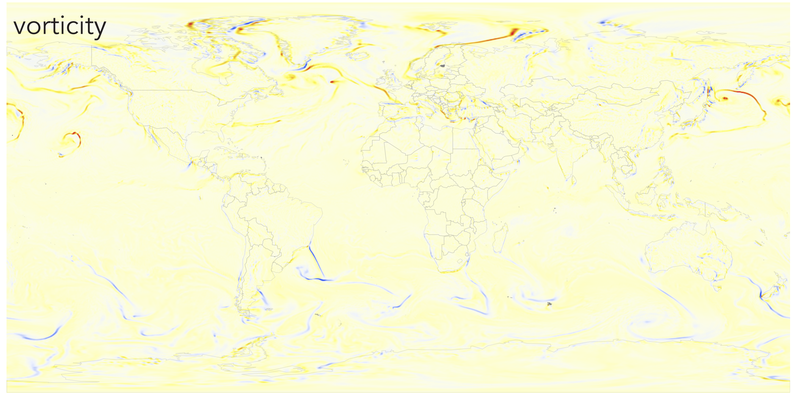
AtmoRep data

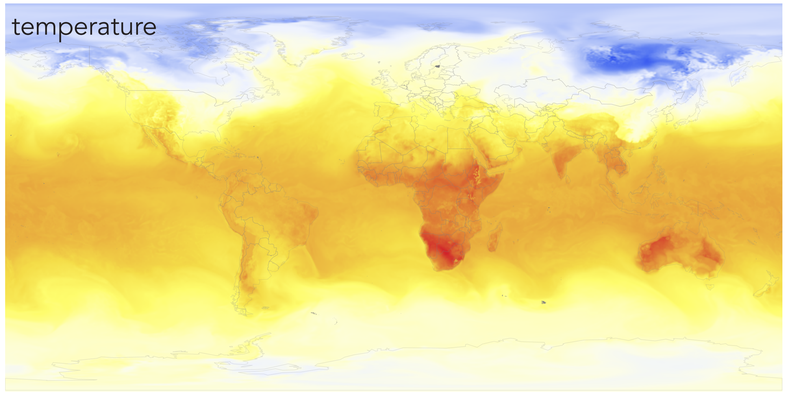


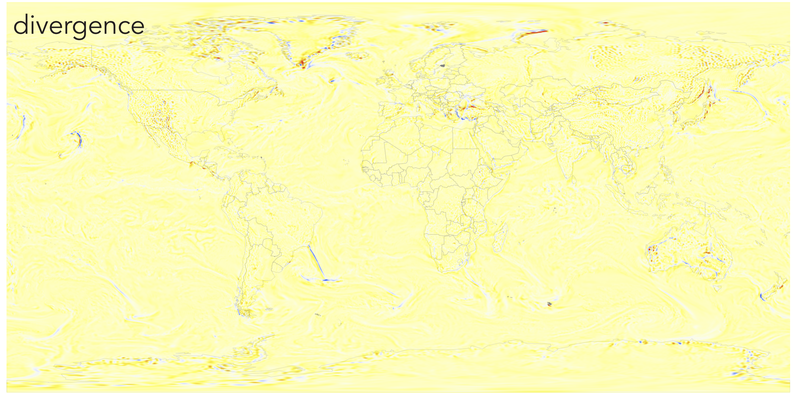
AtmoRep data

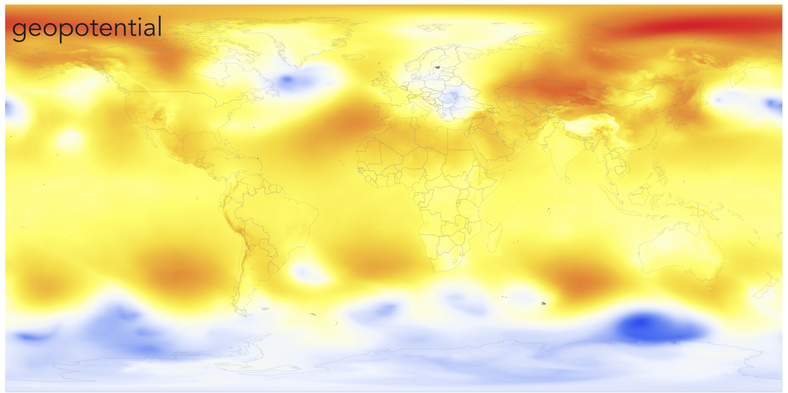


AtmoRep data

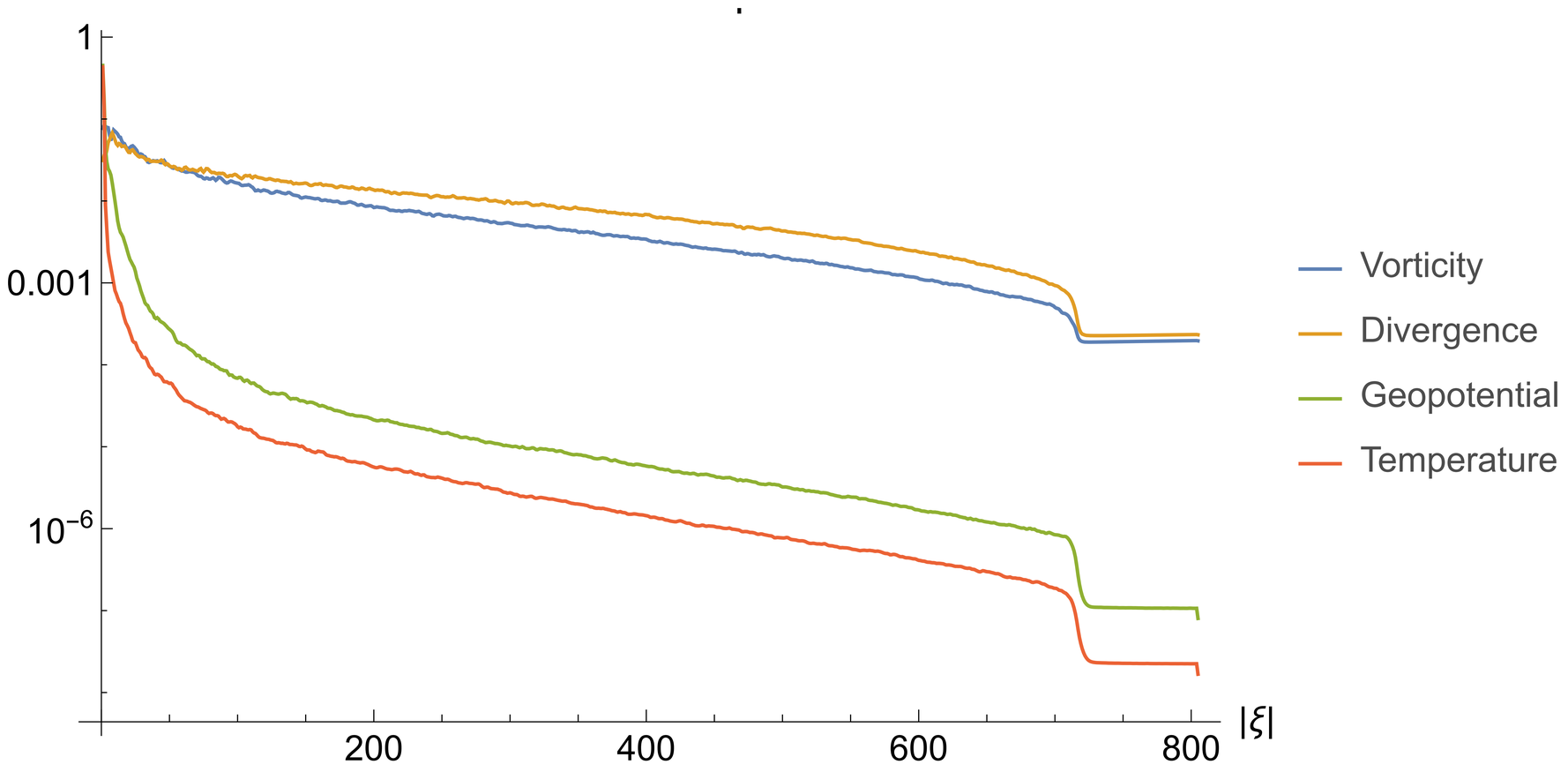




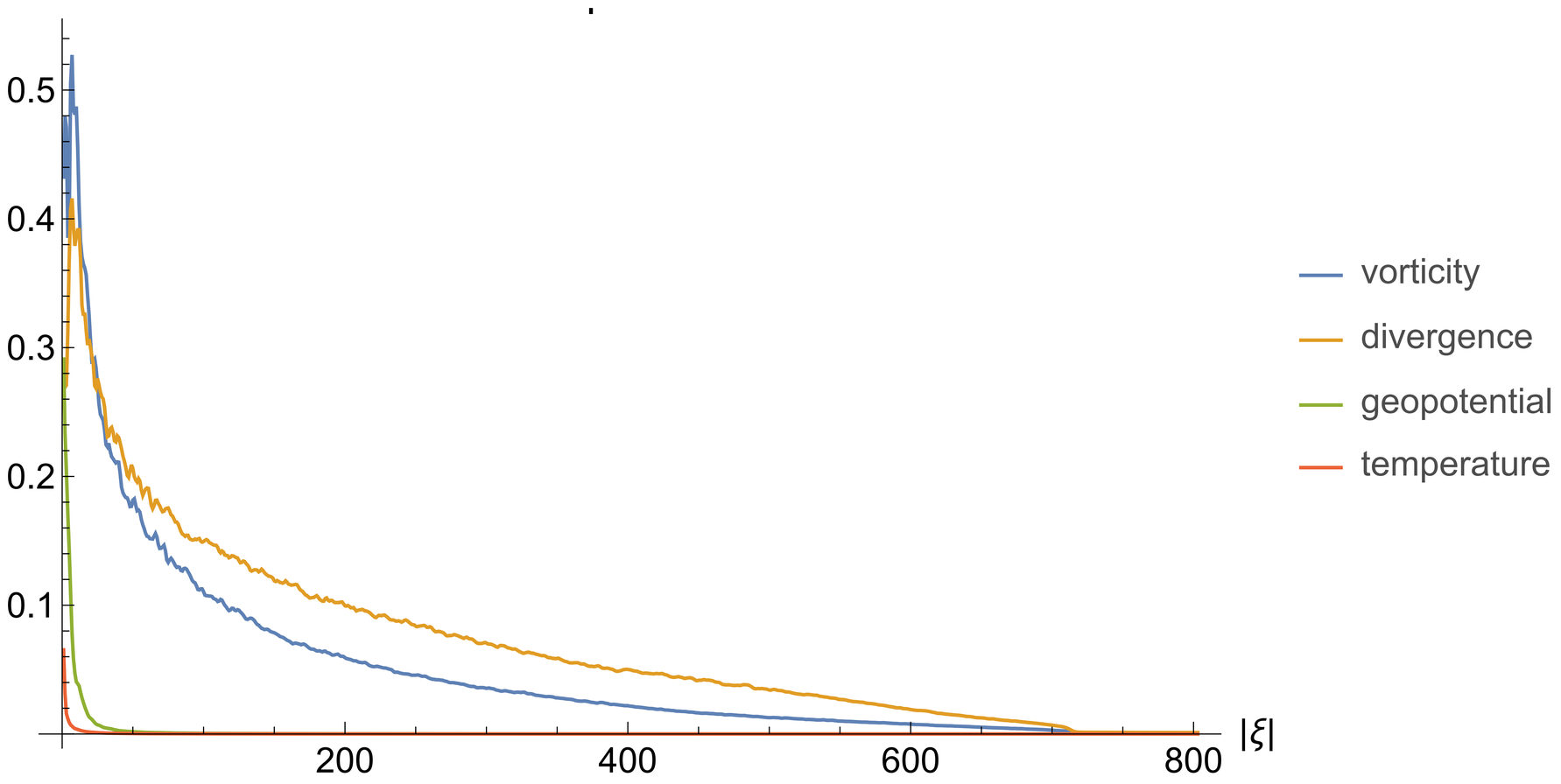




AtmoRep data



AtmoRep data



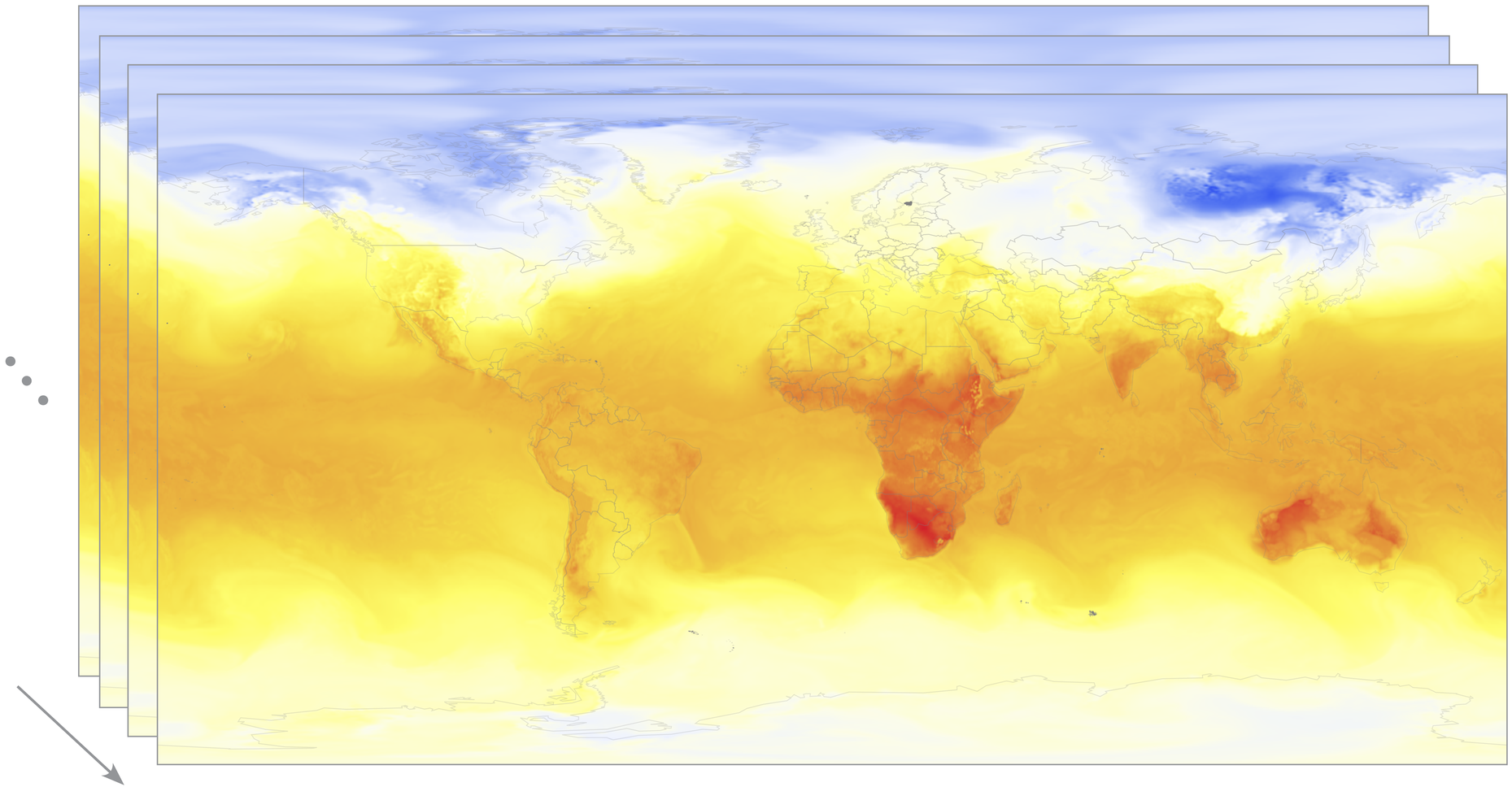
AtmoRep network architecture

- Transformer-based network architecture
  - Scales well to very large data-sets
  - Generative model (with decoder)
  - Attention maps provide (physical) interpretability

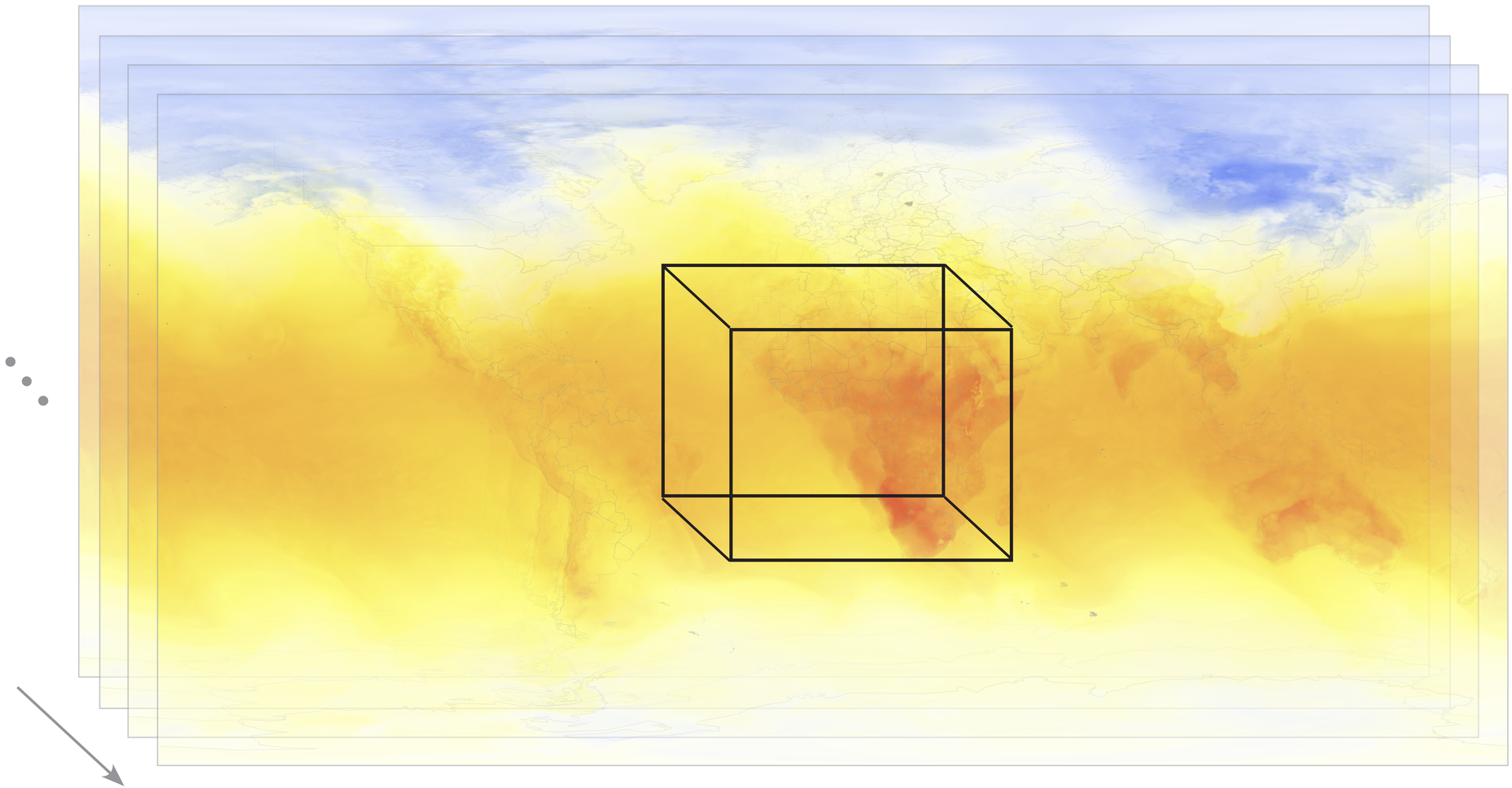
AtmoRep network architecture

- Transformer encoder-based network architecture
  - Scales well to very large data-sets
  - Generative model (with decoder)
  - Attention maps provide (physical) interpretability
- Network is local in space-time

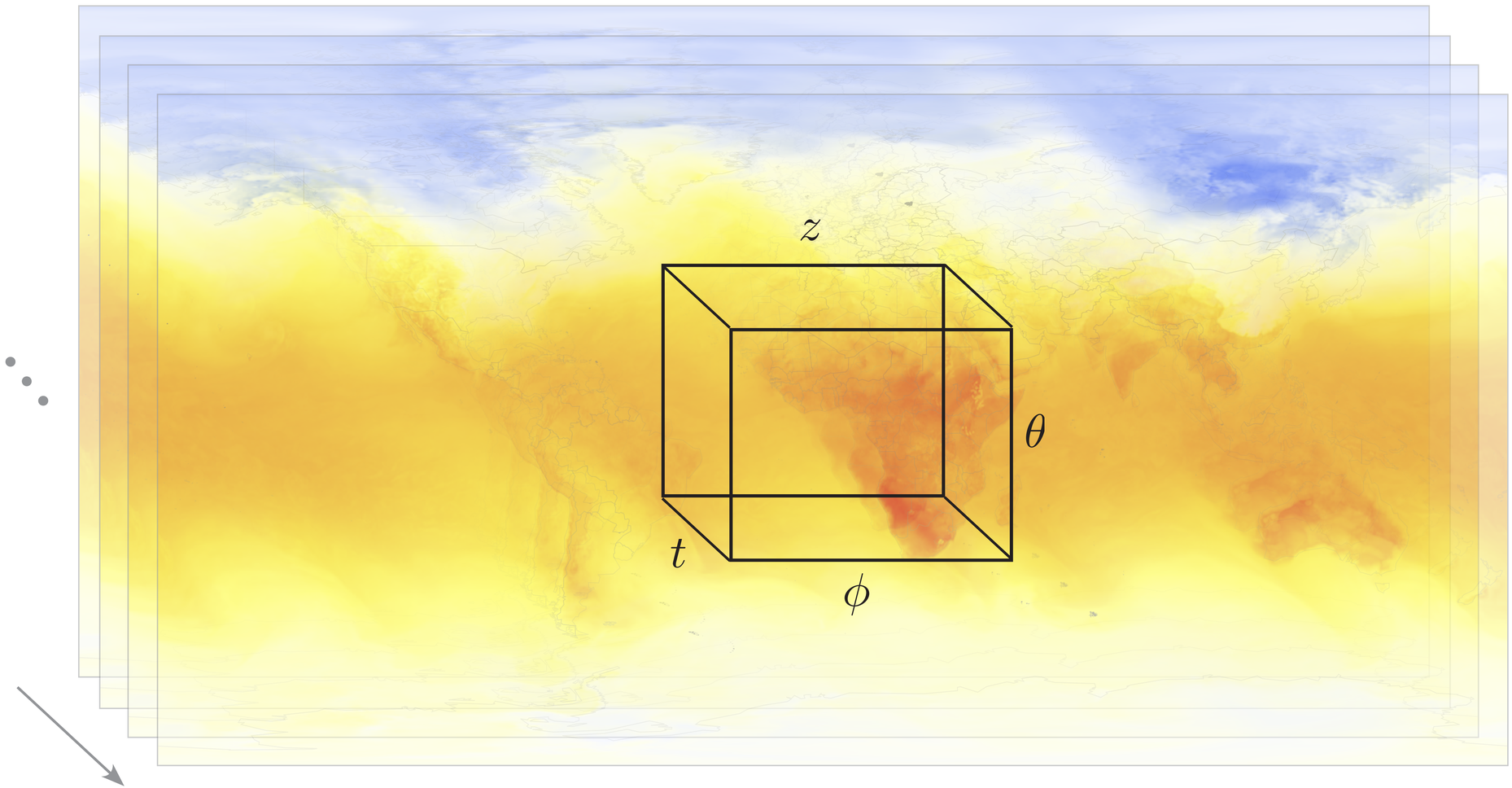
AtmoRep network architecture



AtmoRep network architecture



AtmoRep network architecture

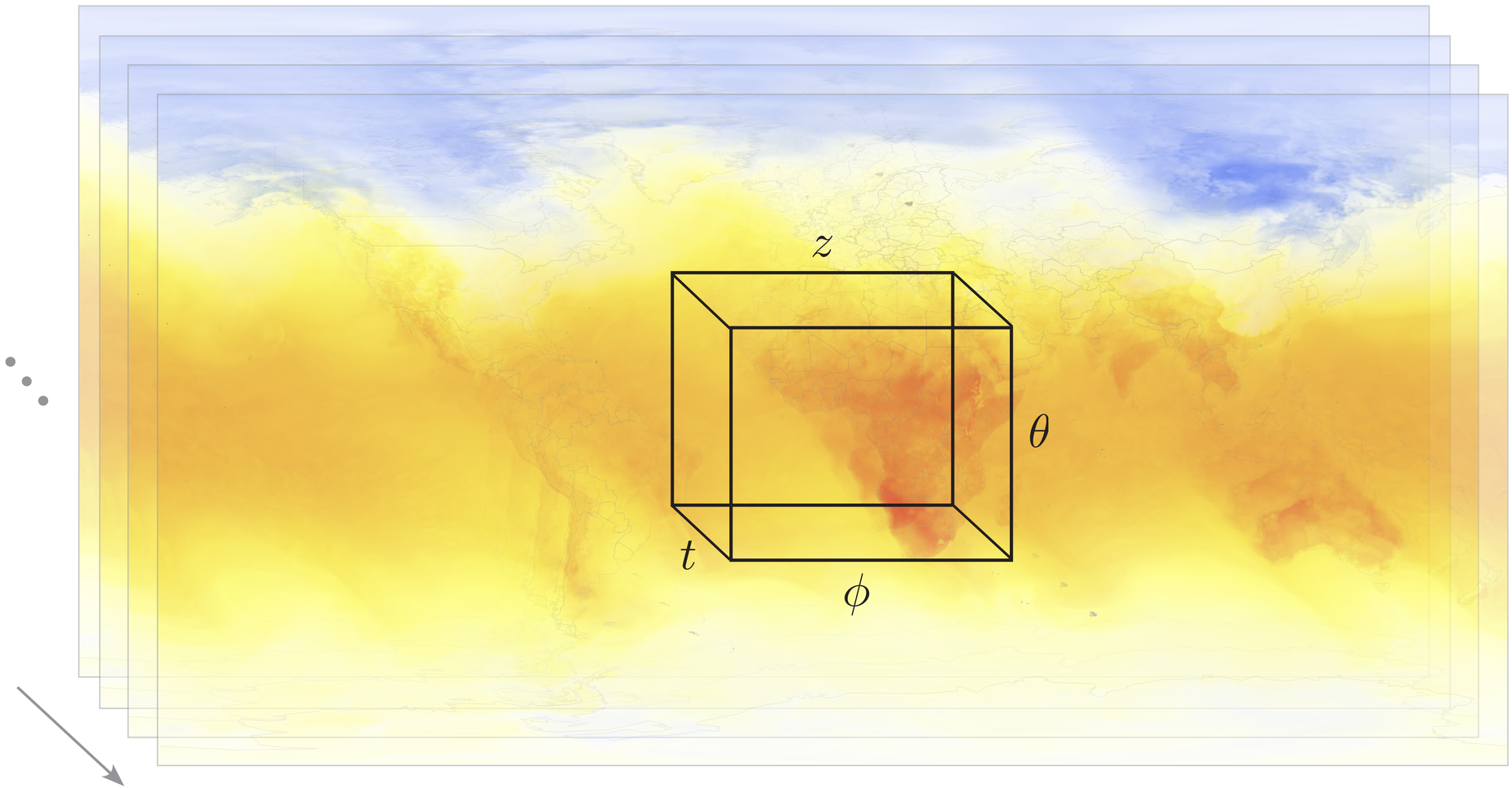


AtmoRep network architecture

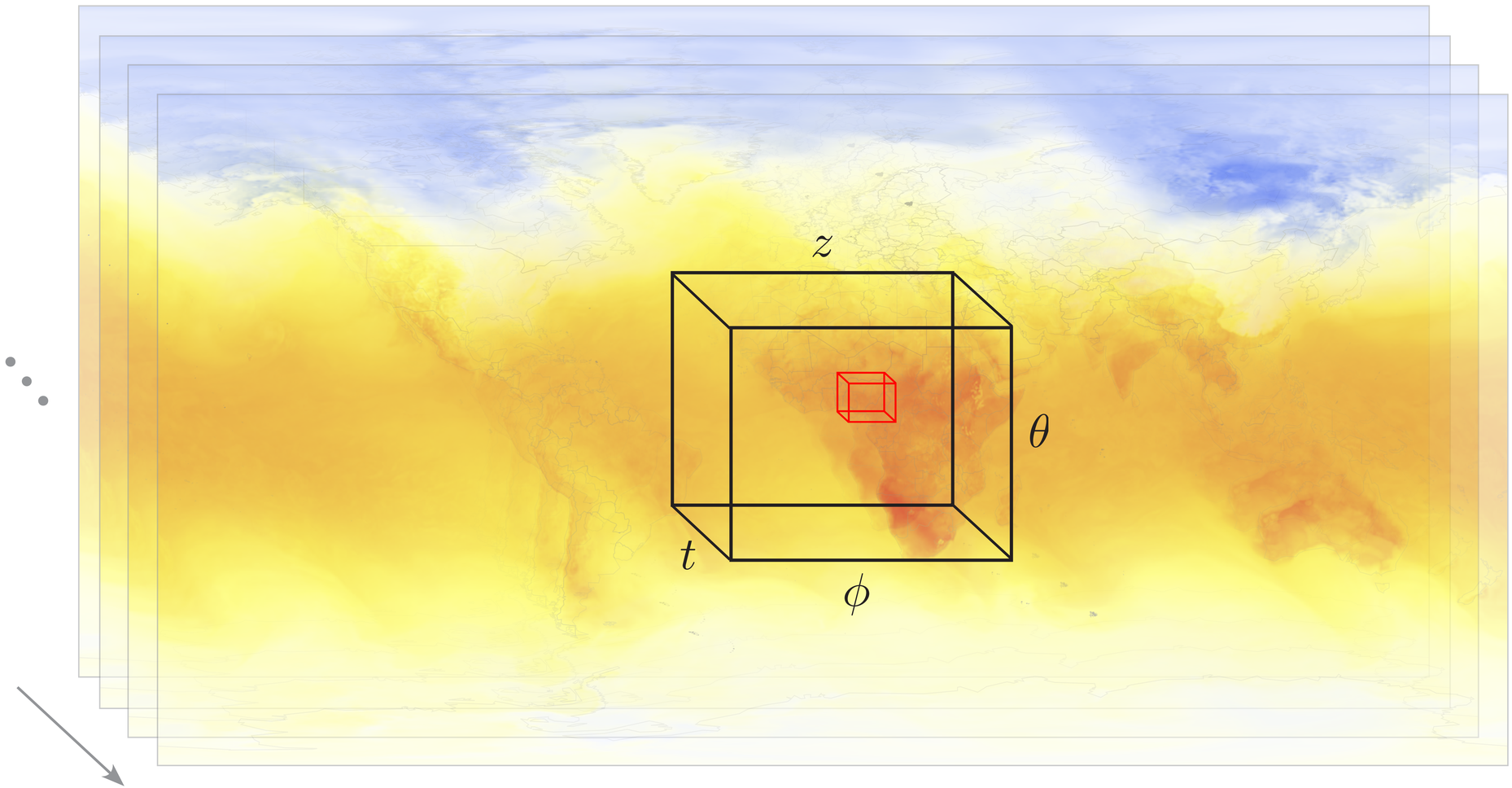
- Transformer encoder-based network architecture
  - Scales well to very large data-sets
  - Generative model (with decoder)
  - Attention maps provide (physical) interpretability
- Network is local in space-time
  - Principal of dynamics are universally valid
  - Local particularities can be learned by providing time + space position as auxiliary information

What is a token?

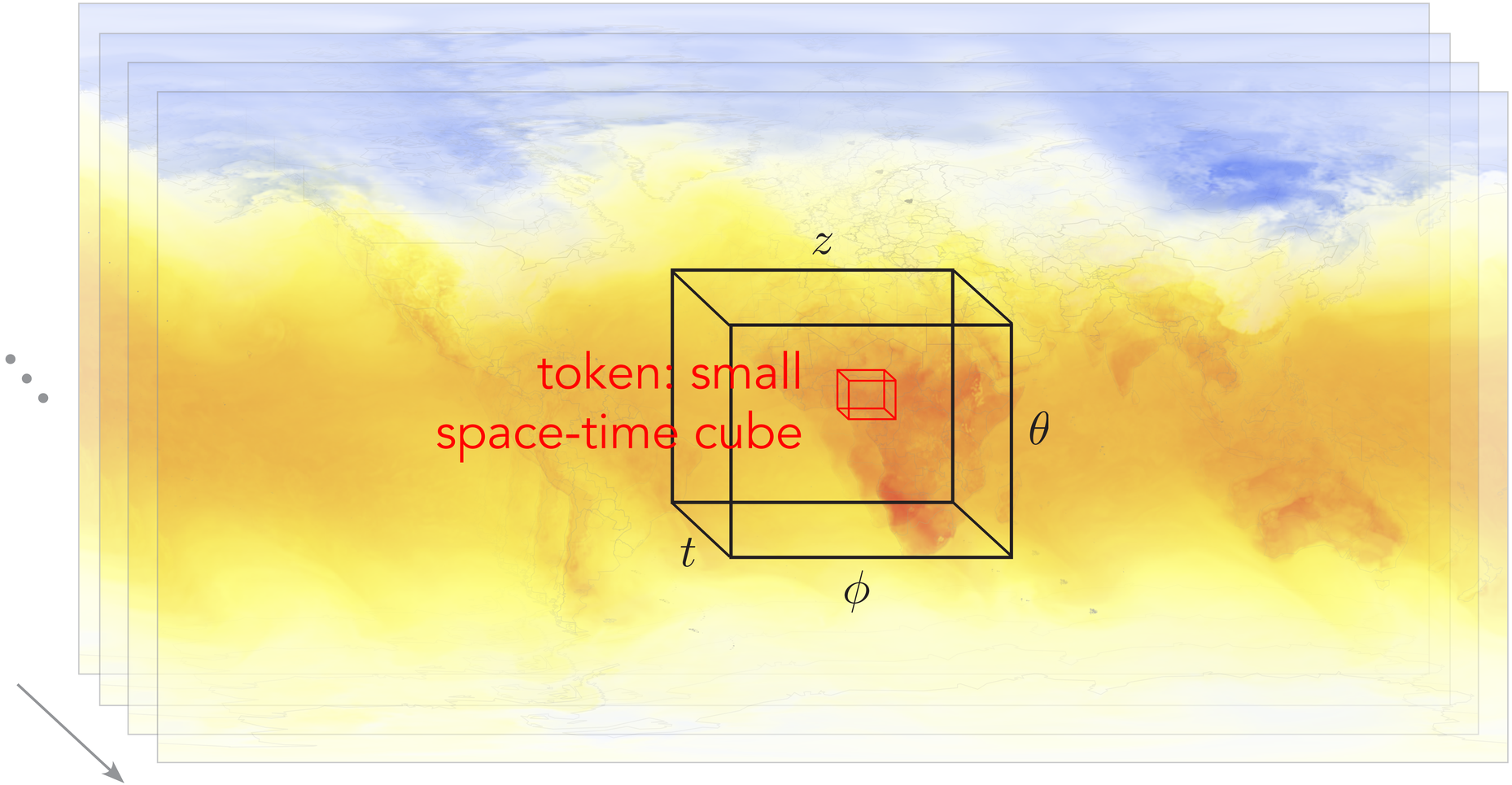
What is a token?



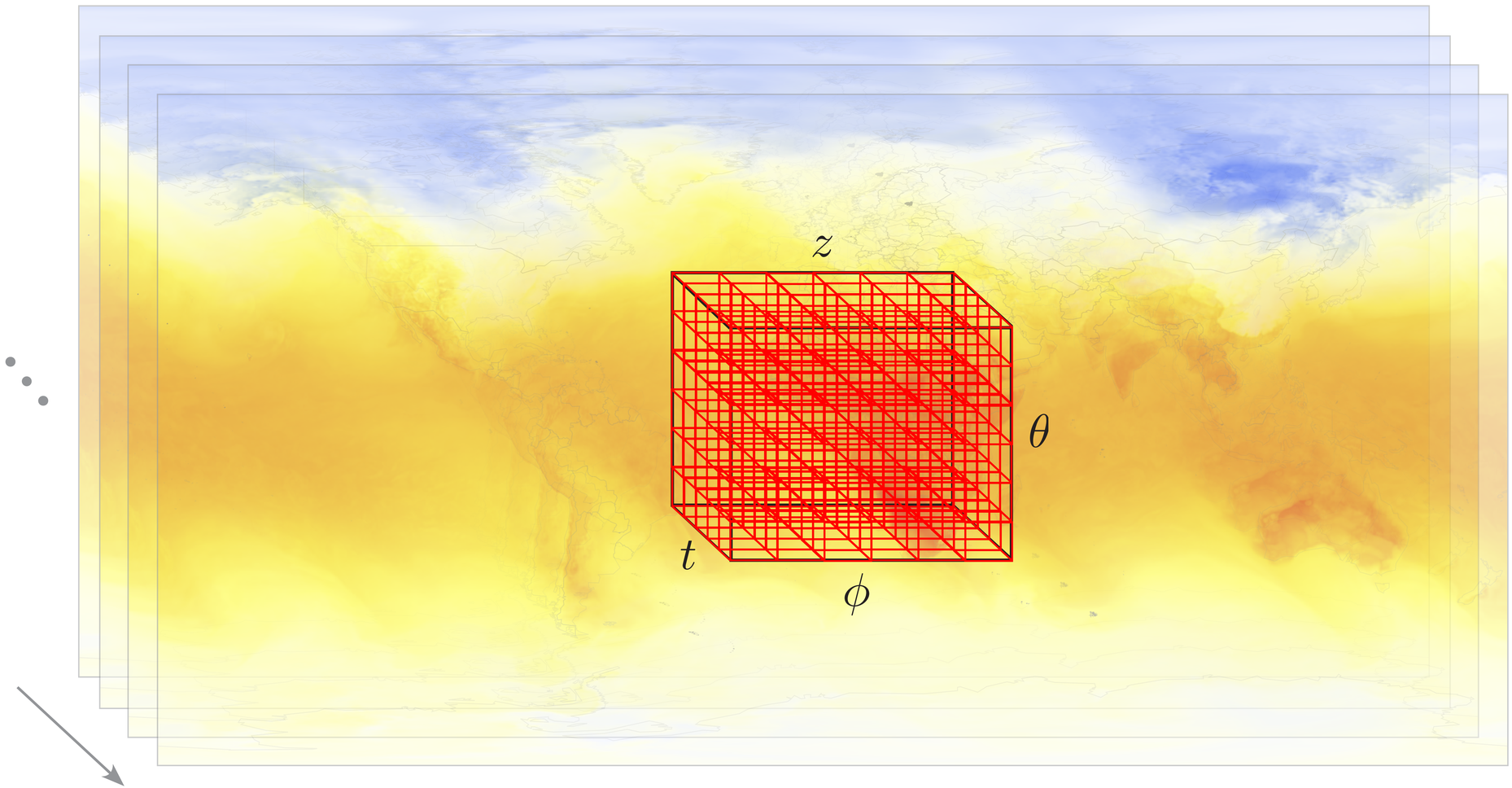
What is a token?



What is a token?

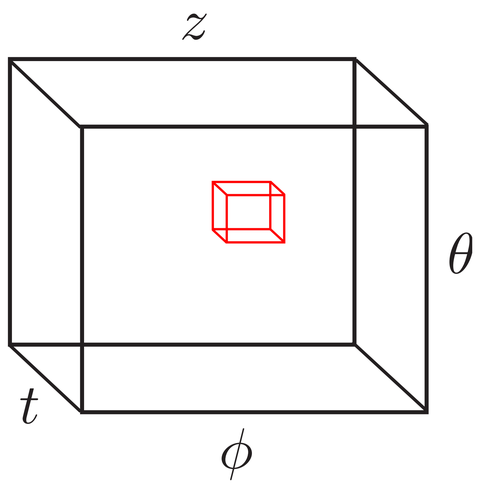


What is a token?



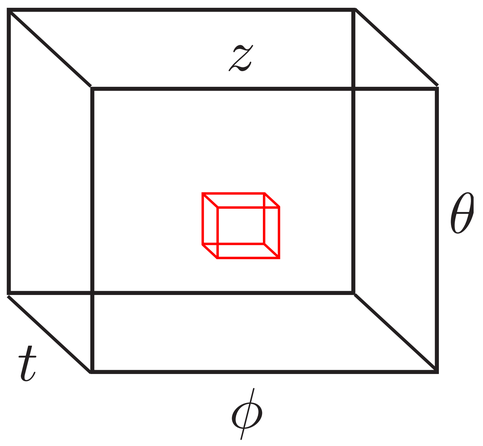
What is a token?

- Token is small neighborhood in space-time
  - Small for token attention / interaction to be informative
  - Big enough so token has rich internal strurcture



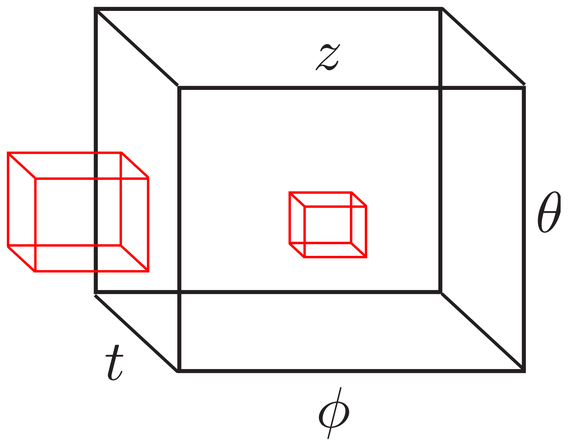
What is a token?

- Token is small neighborhood in space-time
  - Small for token attention / interaction to be informative
  - Big enough so token has rich internal strurcture
- Token size is field-dependent

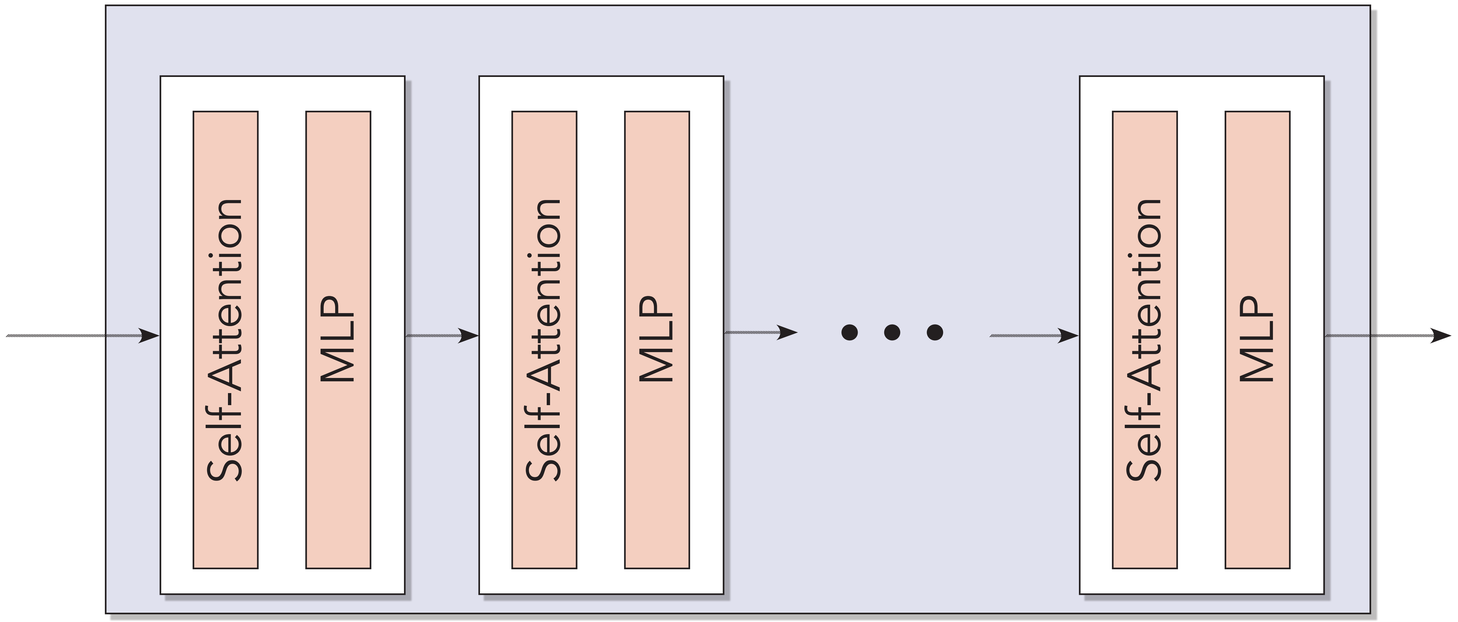


What is a token?

- Token is small neighborhood in space-time
  - Small for token attention / interaction to be informative
  - Big enough so token has rich internal strurcture
- Token size is field-dependent
- Multiple token sizes to provide multi-resolution structure and large contexts for neighborhod



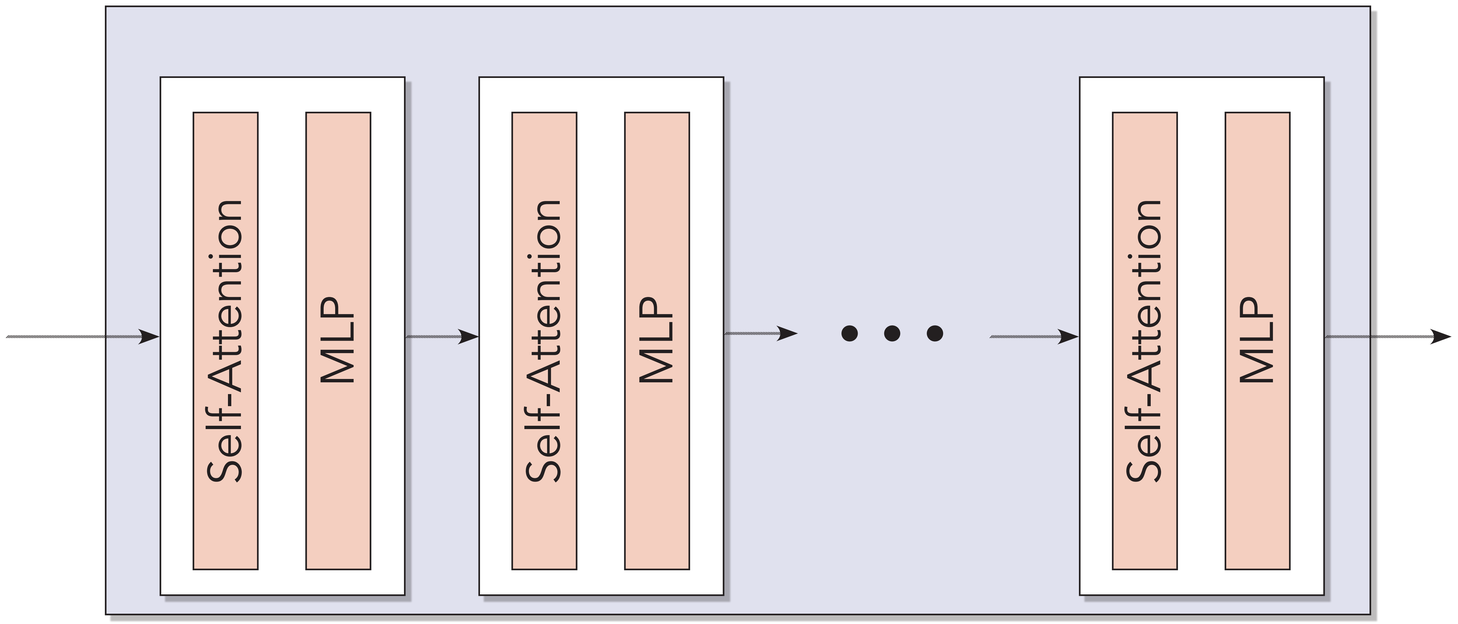
Multiformer: respect the physical fields



Multiformer: respect the physical fields

Self
attention

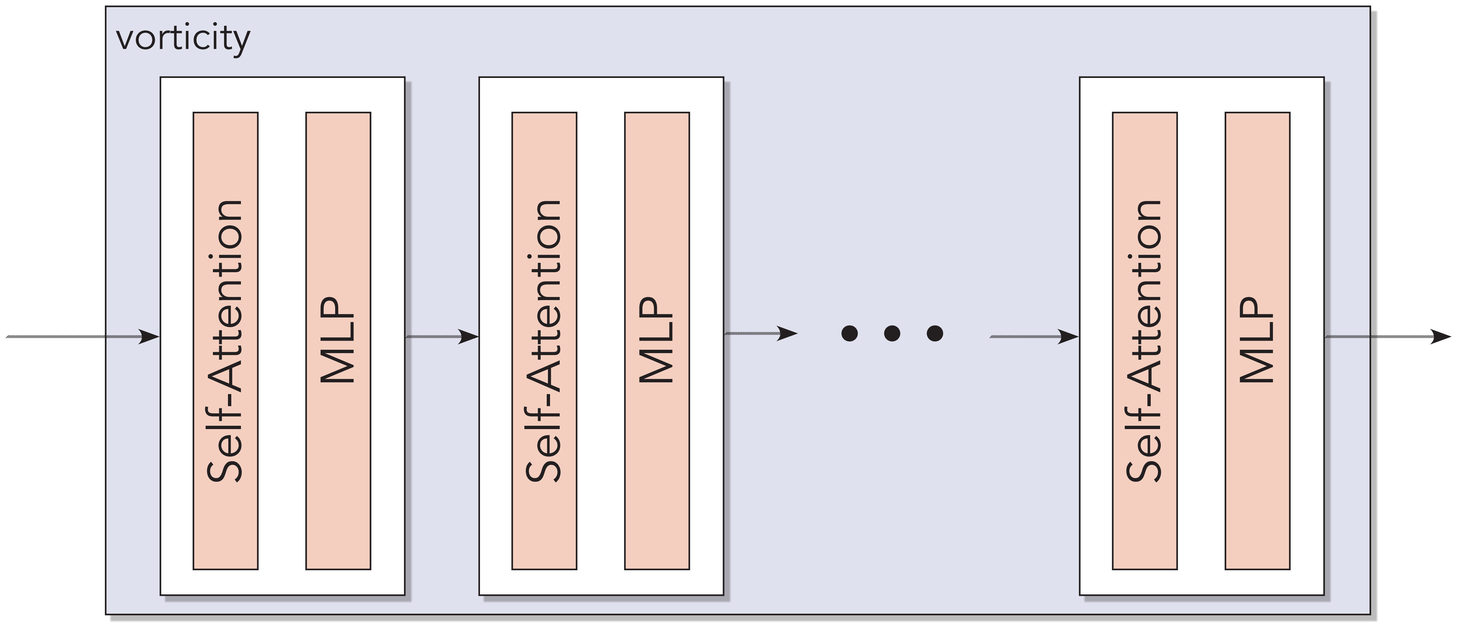
$$\sigma\left(Q\,K^T\right)V$$



Multiformer: respect the physical fields

Self
attention

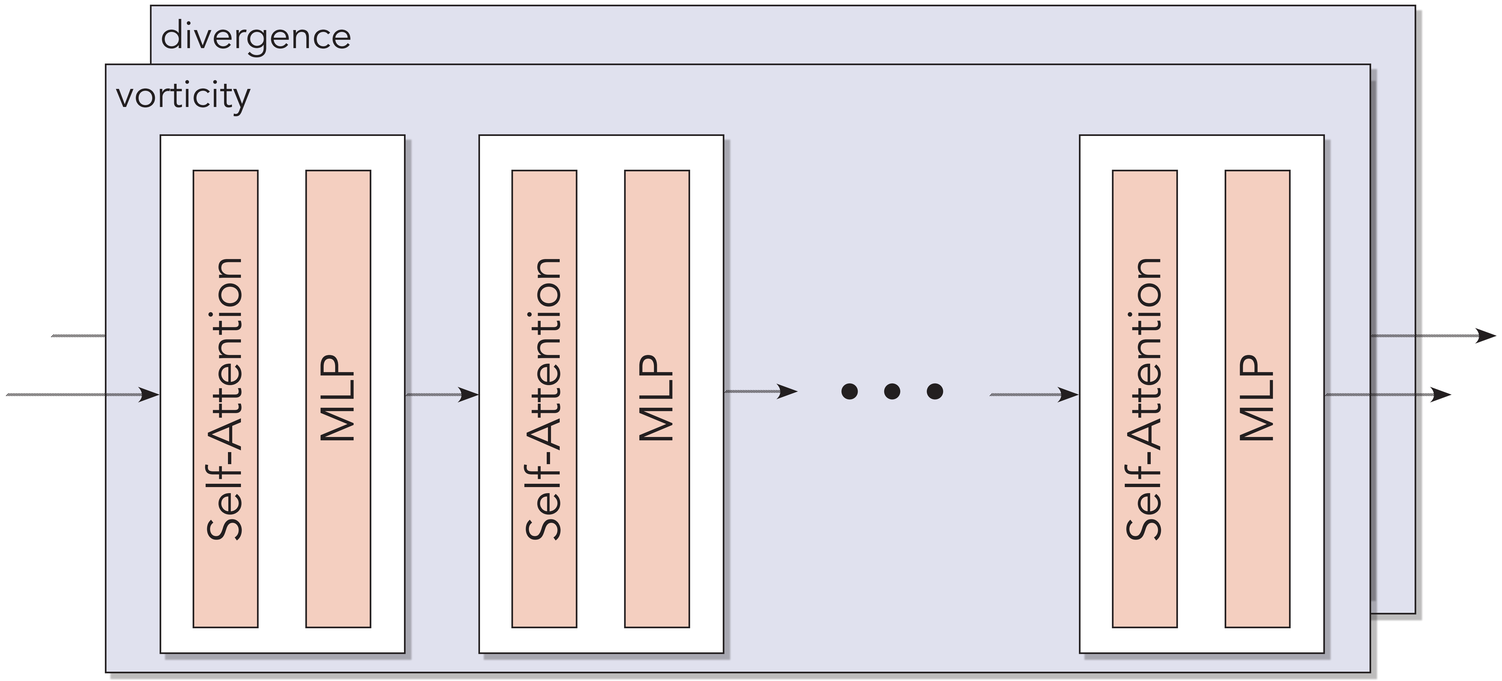
$$\sigma\left(Q\,K^T\right)V$$



Multiformer: respect the physical fields

Self
attention

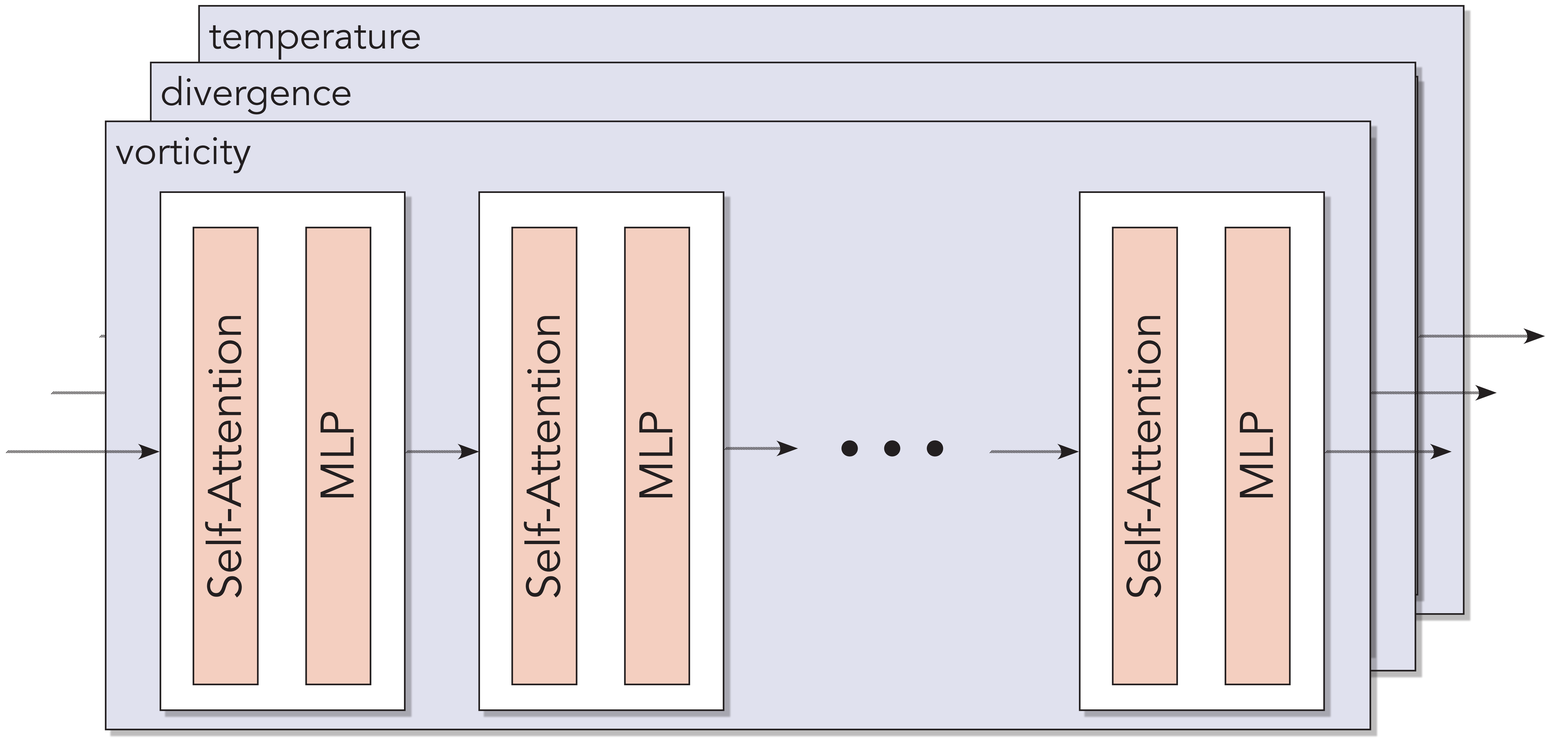
$$\sigma\left(Q\,K^T\right)V$$



Multiformer: respect the physical fields

Self
attention

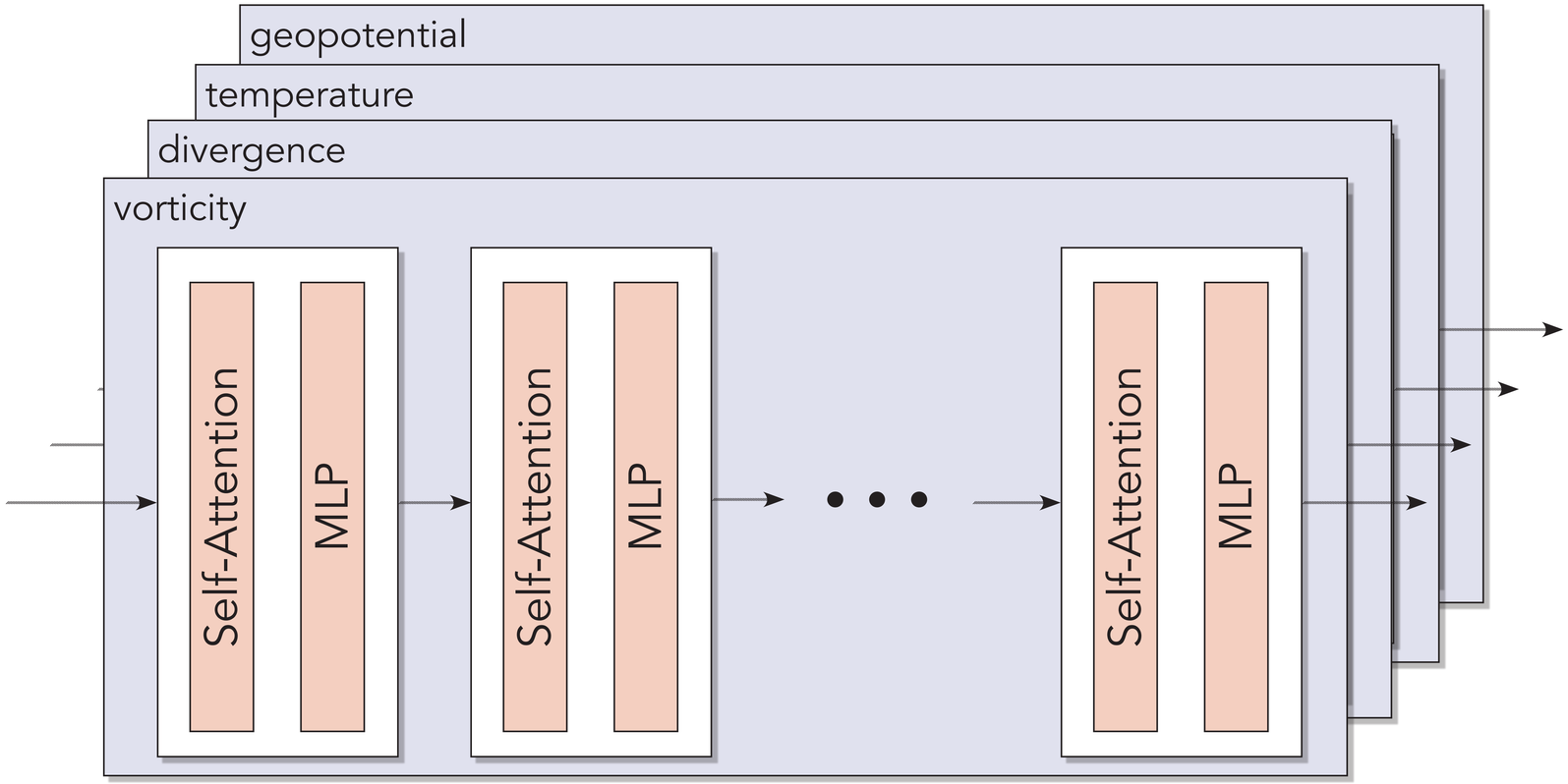
$$\sigma\left(Q\,K^T\right)V$$



Multiformer: respect the physical fields

Self
attention

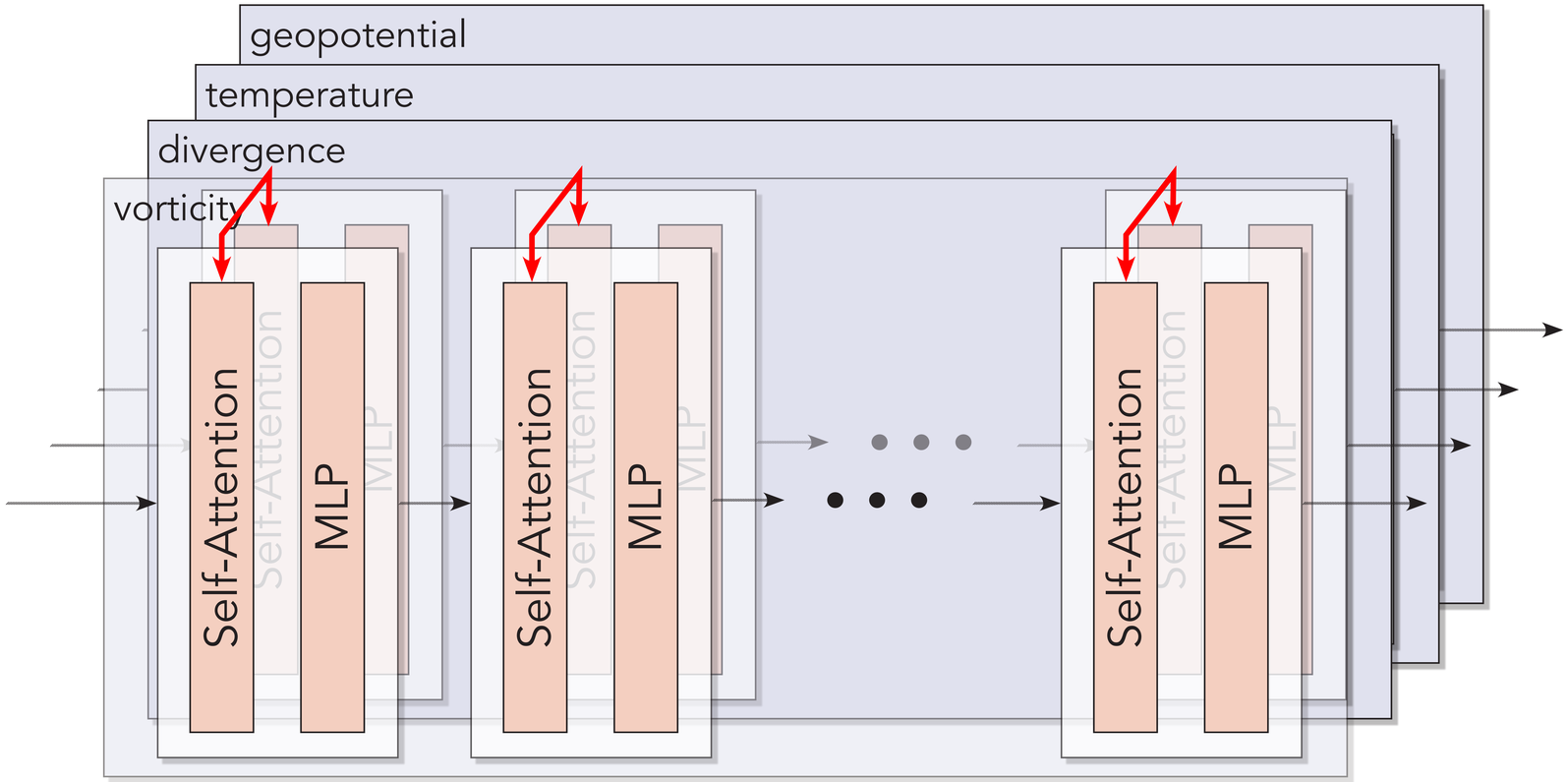
$$\sigma\left(Q\,K^T\right)V$$



Multiformer: respect the physical fields

Self
attention

$$\sigma\left(Q\,K^T\right)V$$



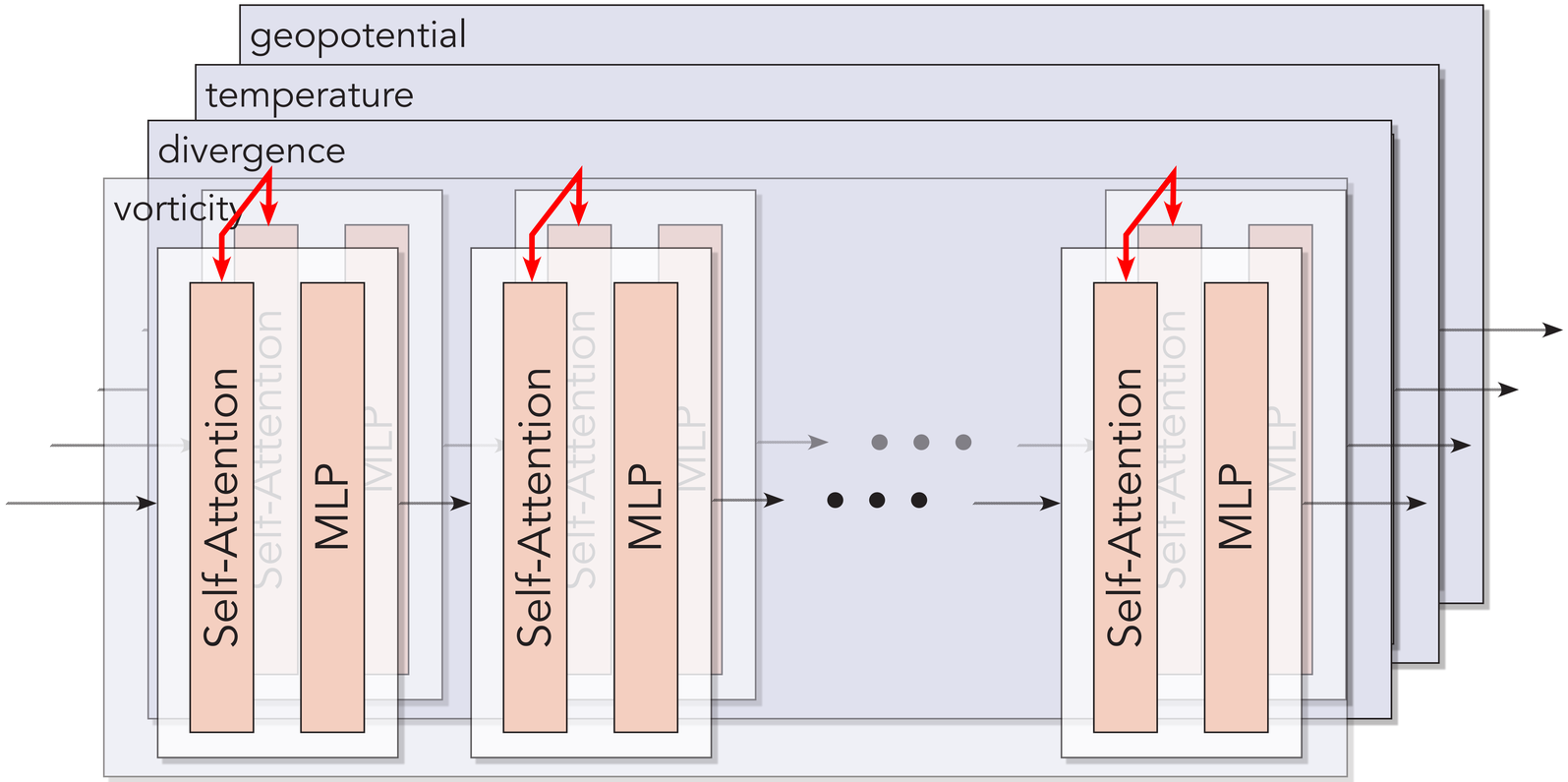
Multiformer: respect the physical fields

Self attention

$$\sigma\left(Q\,K^T\right)V$$

Cross attention

$$\sigma\left(Q_\zeta\,K_\mu^T\right)V_\mu$$



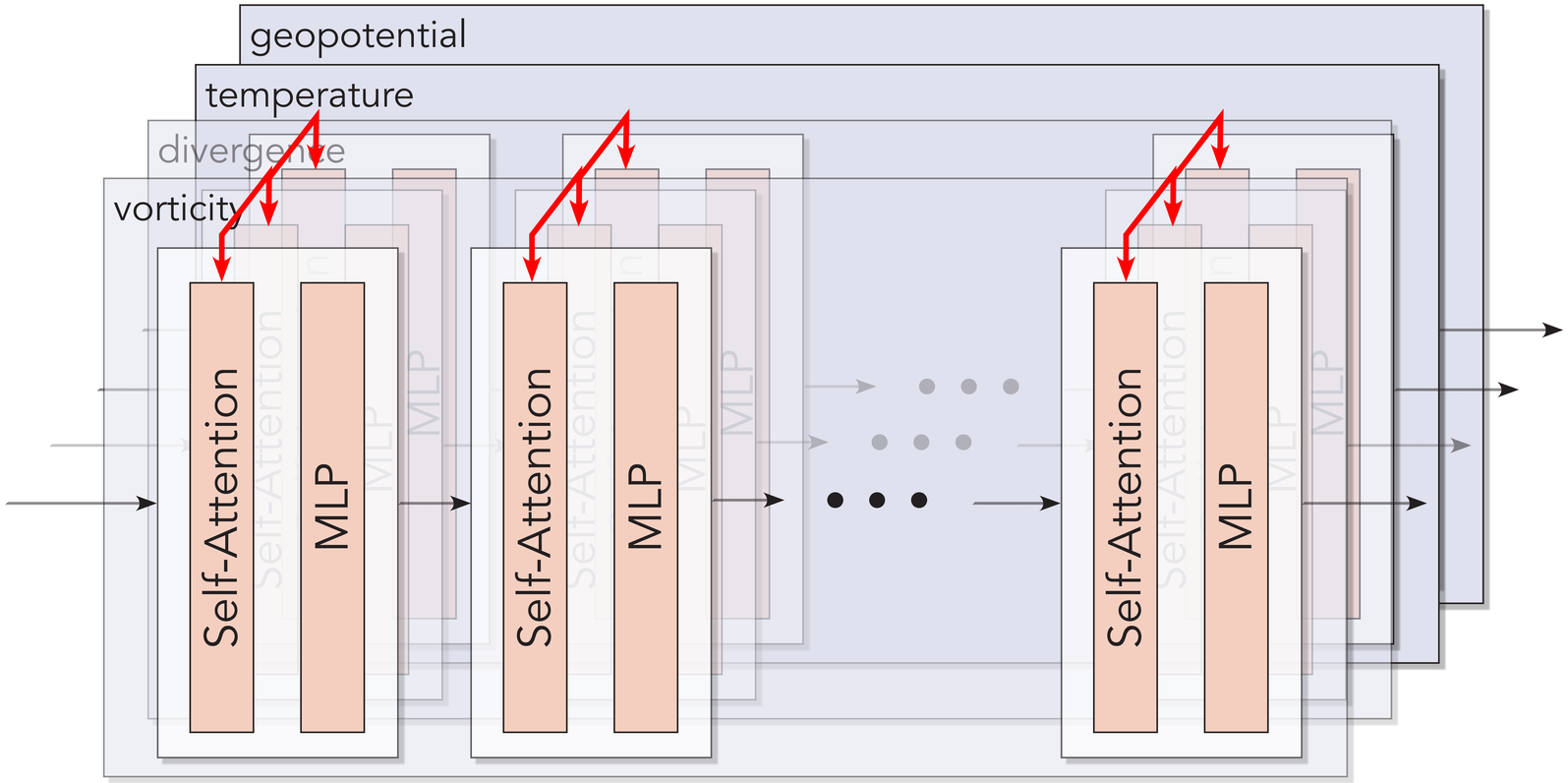
Multiformer: respect the physical fields

Self attention

$$\sigma\left(Q\,K^T\right)V$$

Cross attention

$$\sigma\left(Q_\zeta\,K_\mu^T\right)V_\mu$$



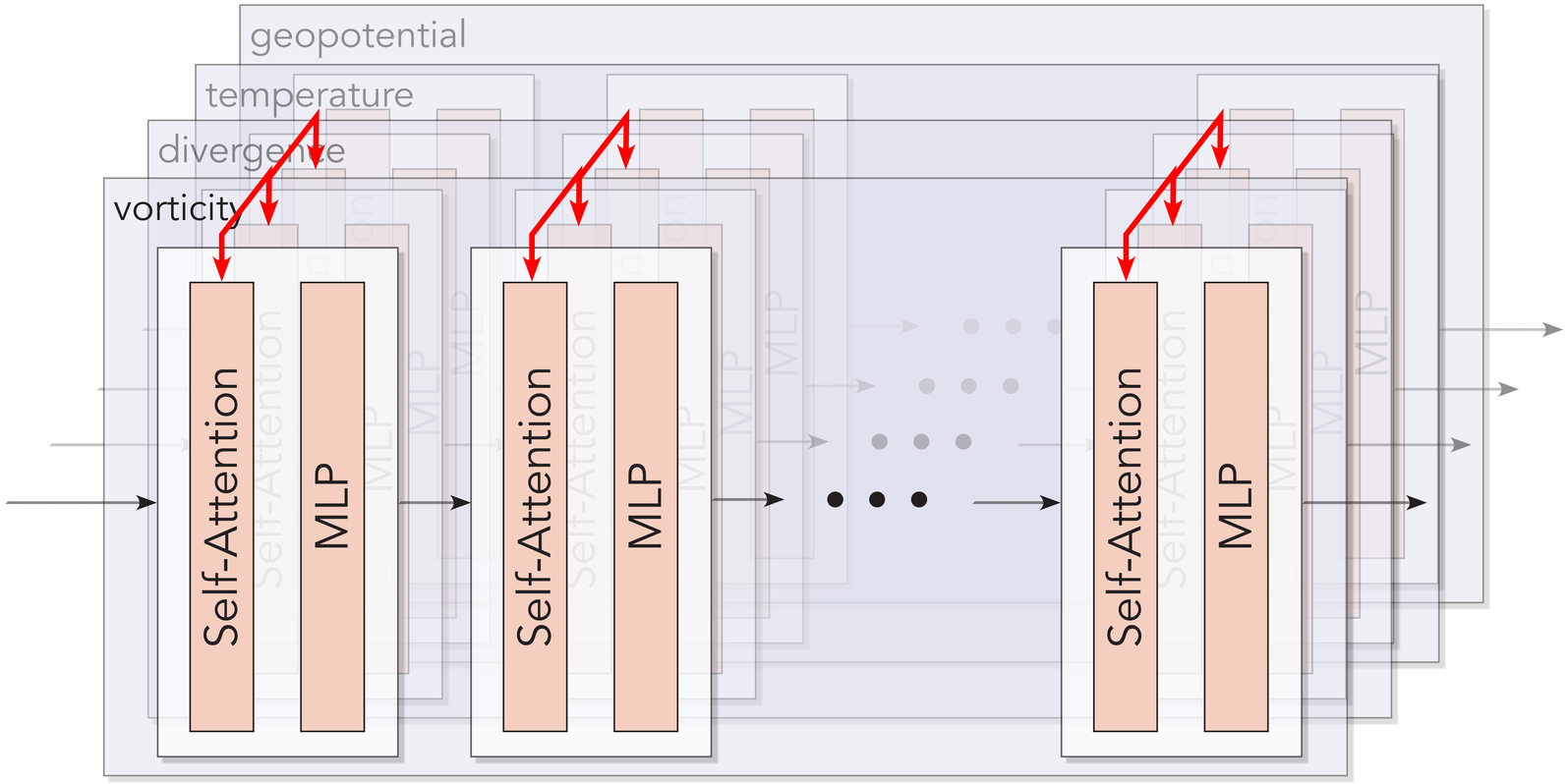
Multiformer: respect the physical fields

Self attention

$$\sigma\left(Q\,K^T\right)V$$

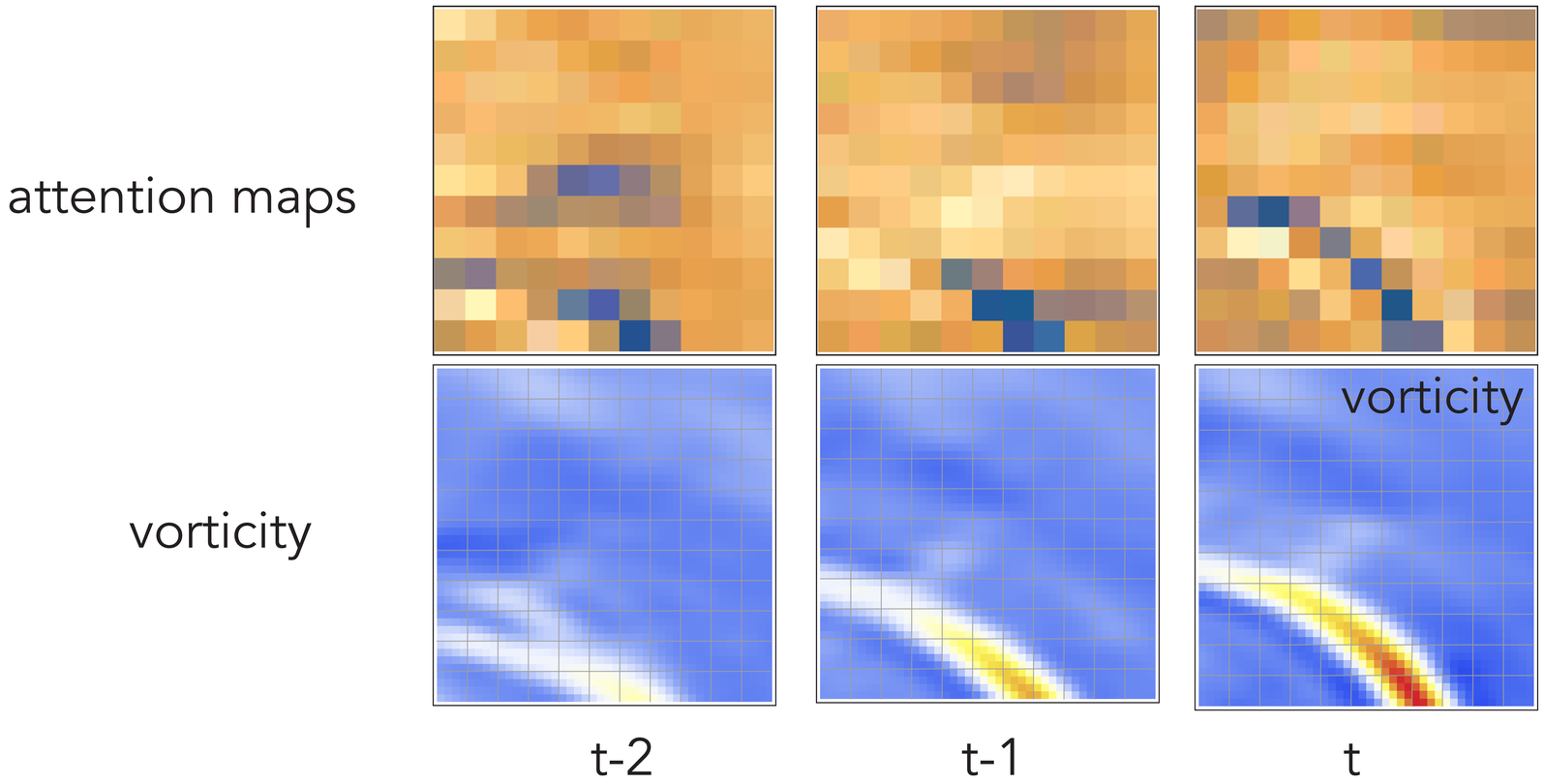
Cross attention

$$\sigma\left(Q_\zeta\,K_\mu^T\right)V_\mu$$



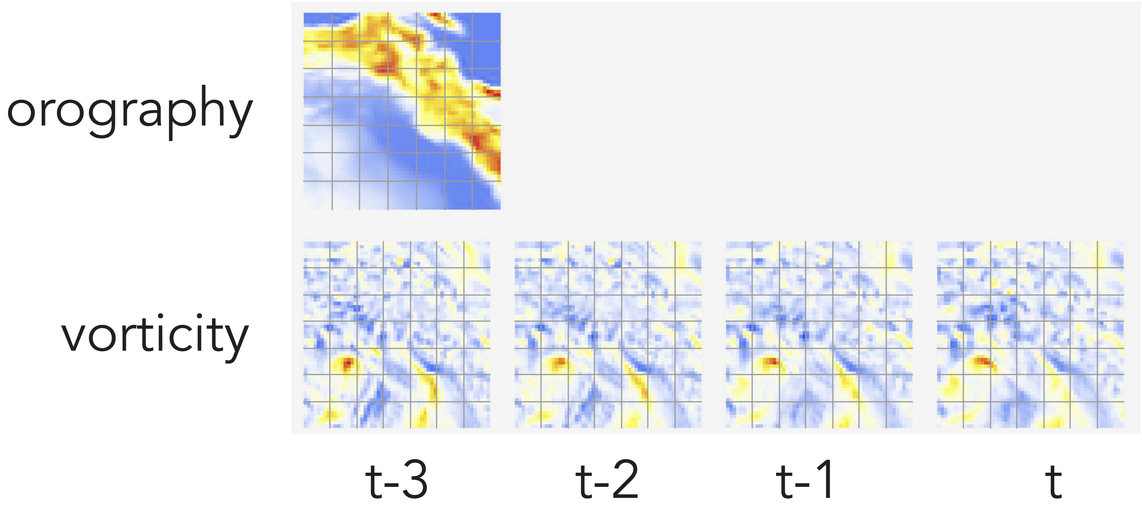
Multiformer: respect the physical fields

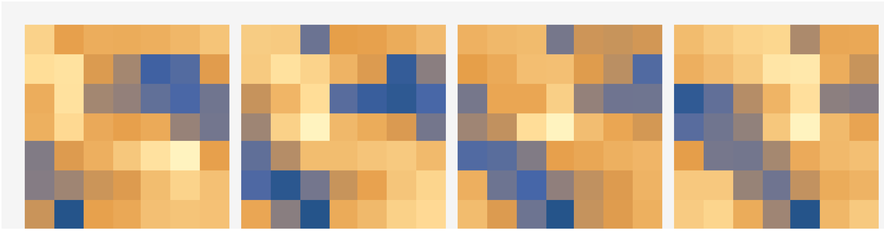
- Attention maps:



Multiformer: respect the physical fields

- Attention maps:

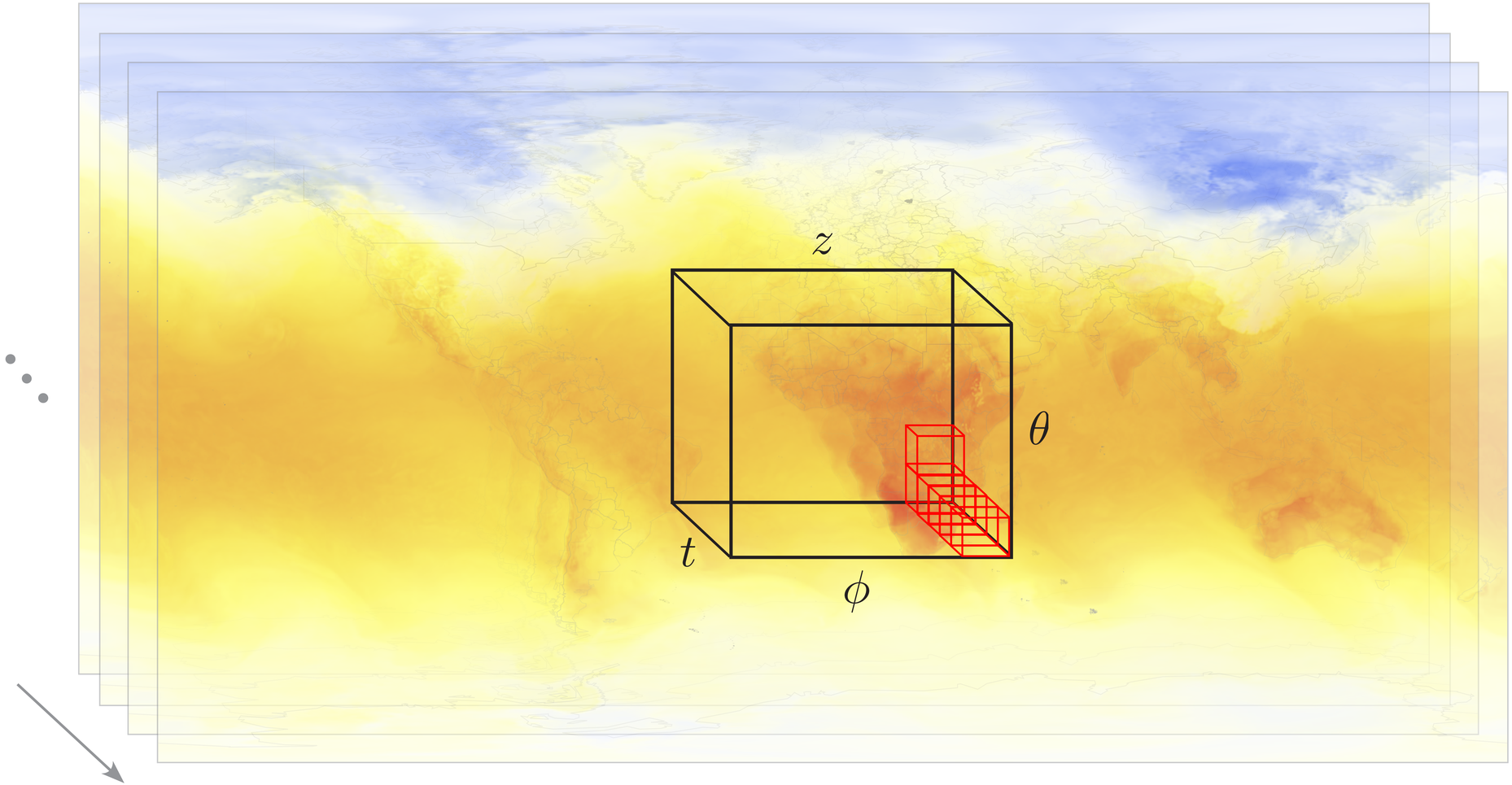




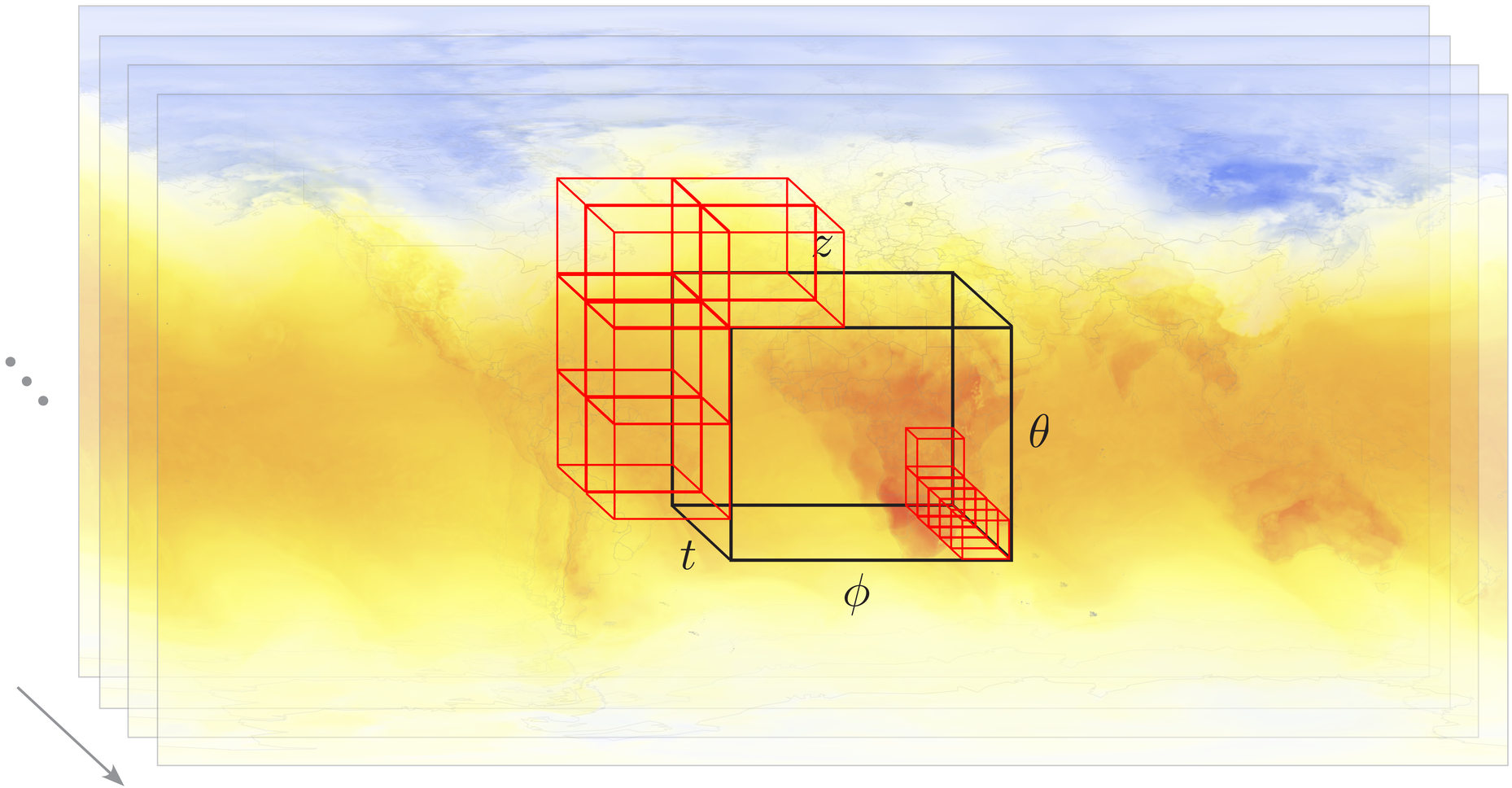
Embedding of tokens

- Use non-trivial embedding network so that it models longer range effects and field interactions in a rich latent space
- Allow for tokens of different size in space-time

Embedding of tokens



Embedding of tokens



Embedding of tokens

- Use non-trivial embedding network so that multiformer can model longer range effects and field interactions in a rich latent space
- Allow for tokens of different size in space-time

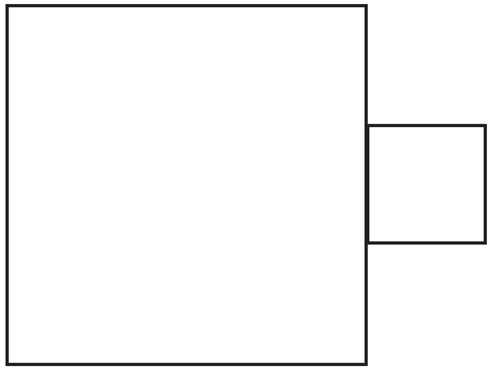
=> Use small/medium-size standard transformer as embedding network

Embedding of tokens

- Allow for tokens of different size in space-time

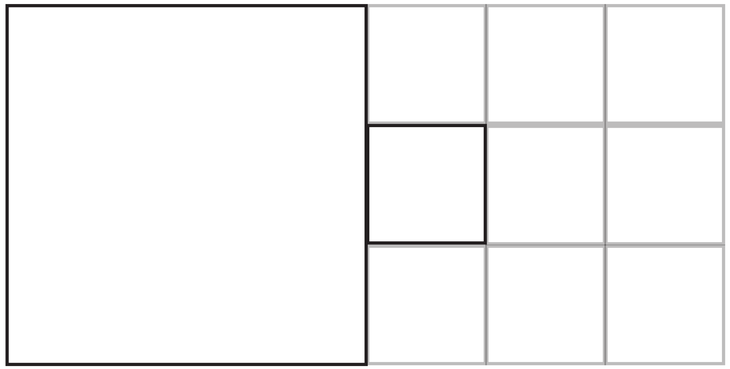
Embedding of tokens

- Allow for tokens of different size in space-time



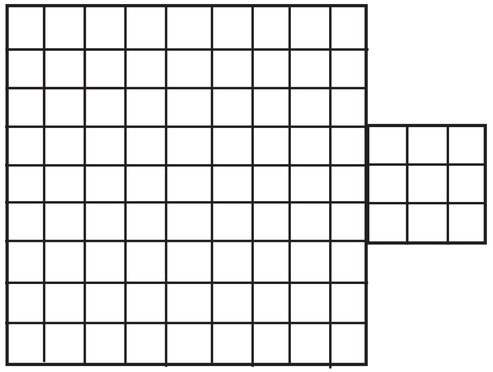
Embedding of tokens

- Allow for tokens of different size in space-time



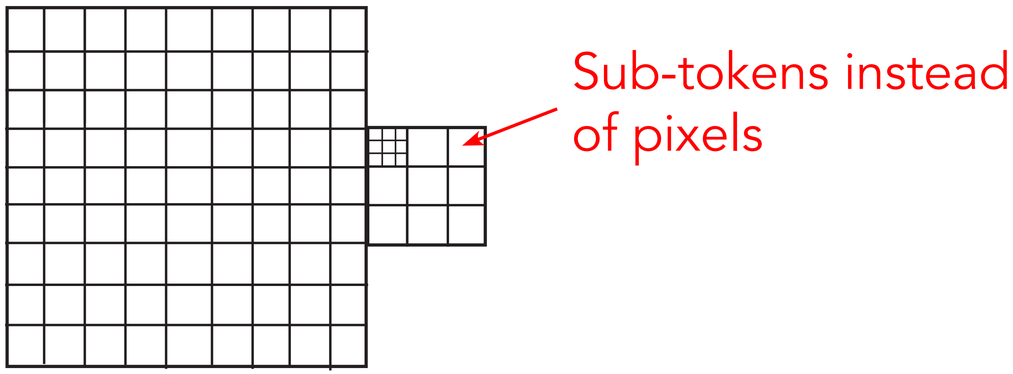
Embedding of tokens

- Allow for tokens of different size in space-time



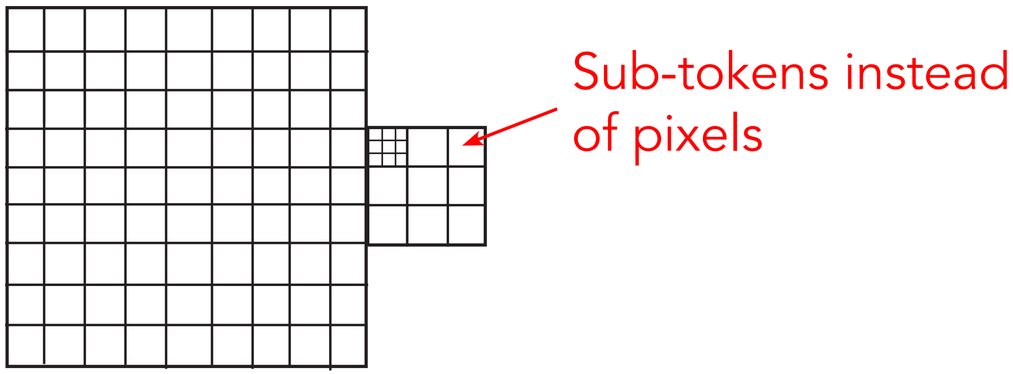
Embedding of tokens

- Allow for tokens of different size in space-time



Embedding of tokens

- Allow for tokens of different size in space-time

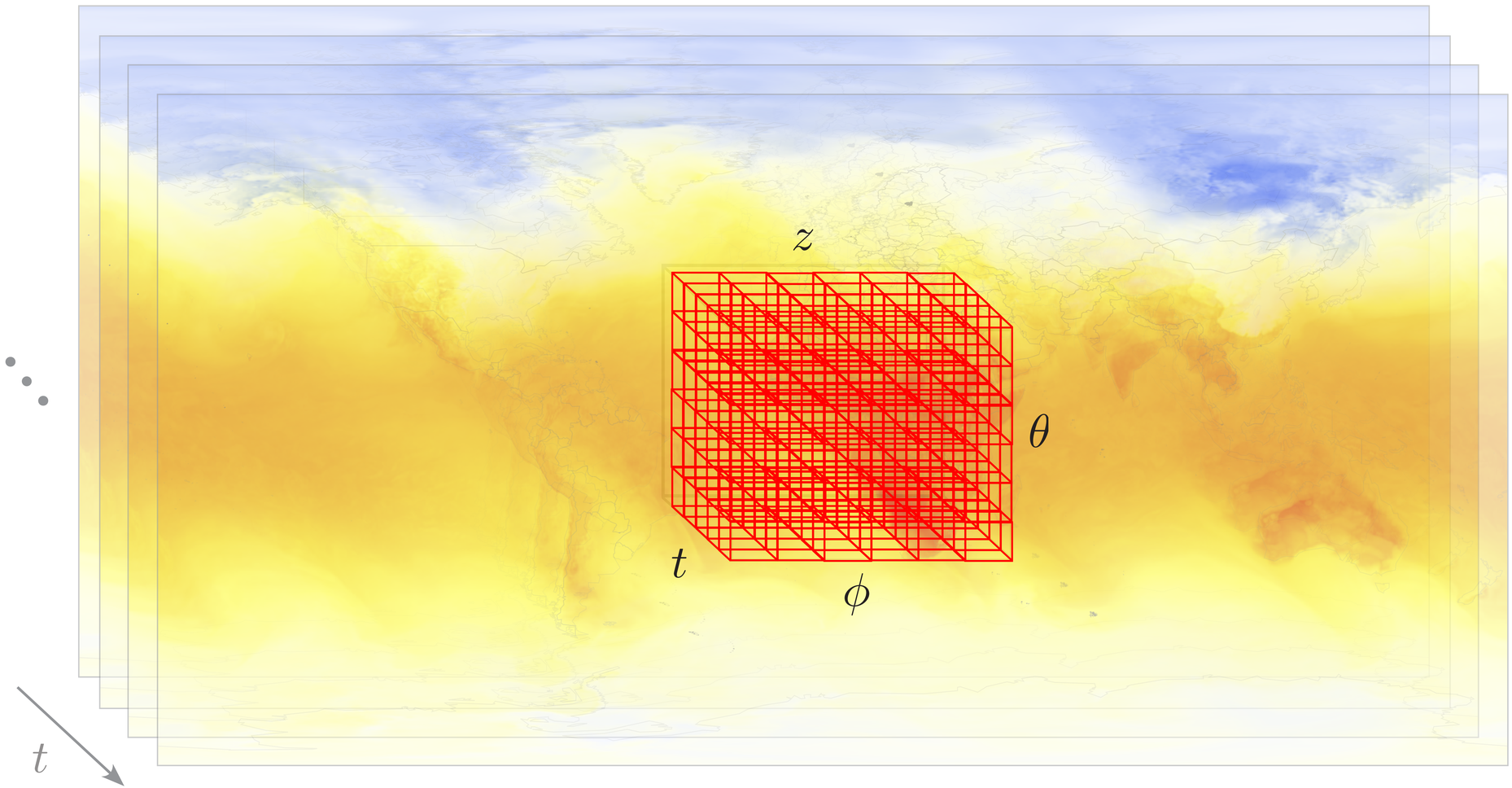


- Transformer takes an arbitrary number of tokens as input
  - Training yields consistent embedding

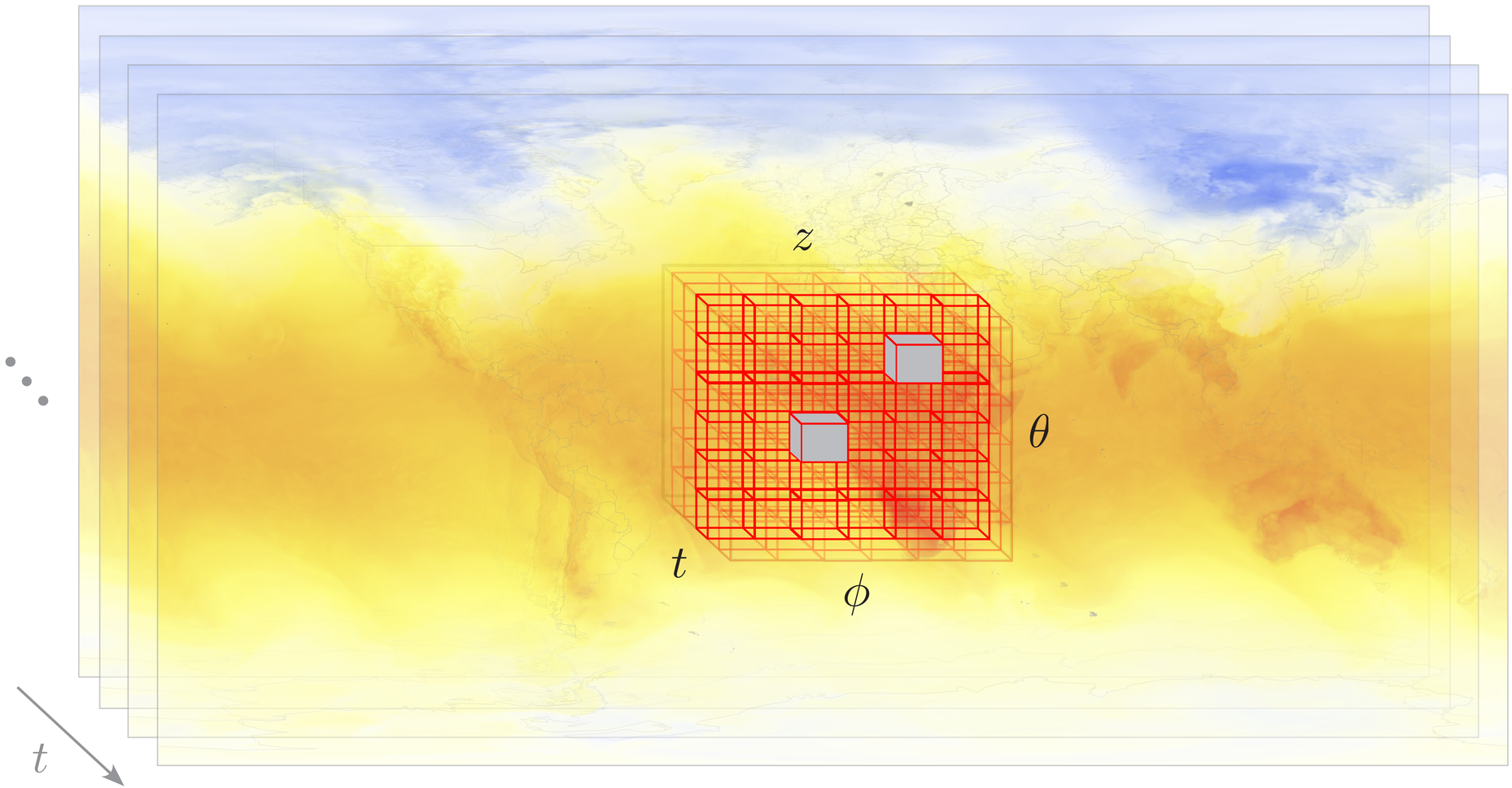
Training of embedding network

- Self-supervised training with variation of BERT masked language language model

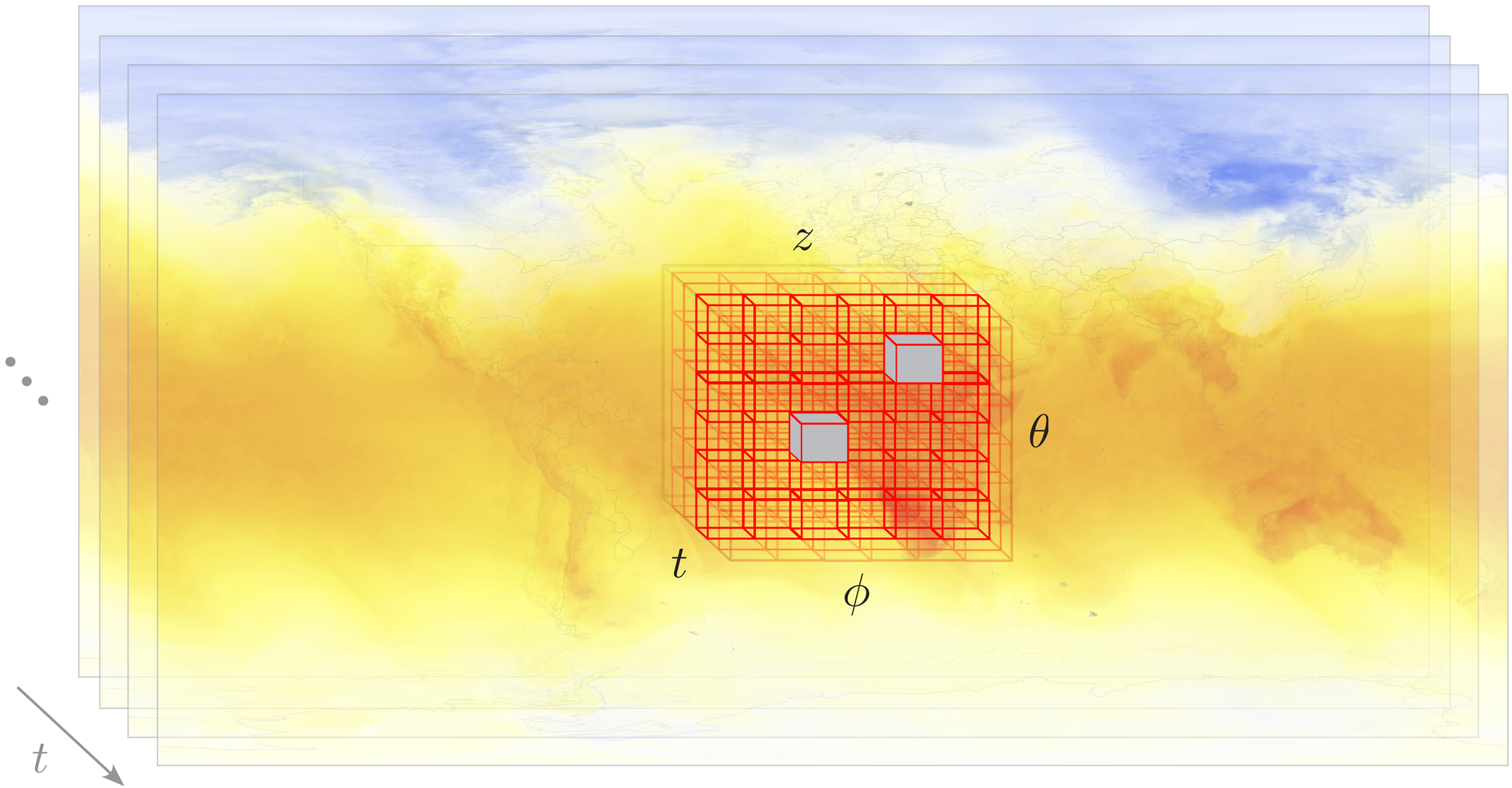
AtmoRep training



AtmoRep training



AtmoRep training



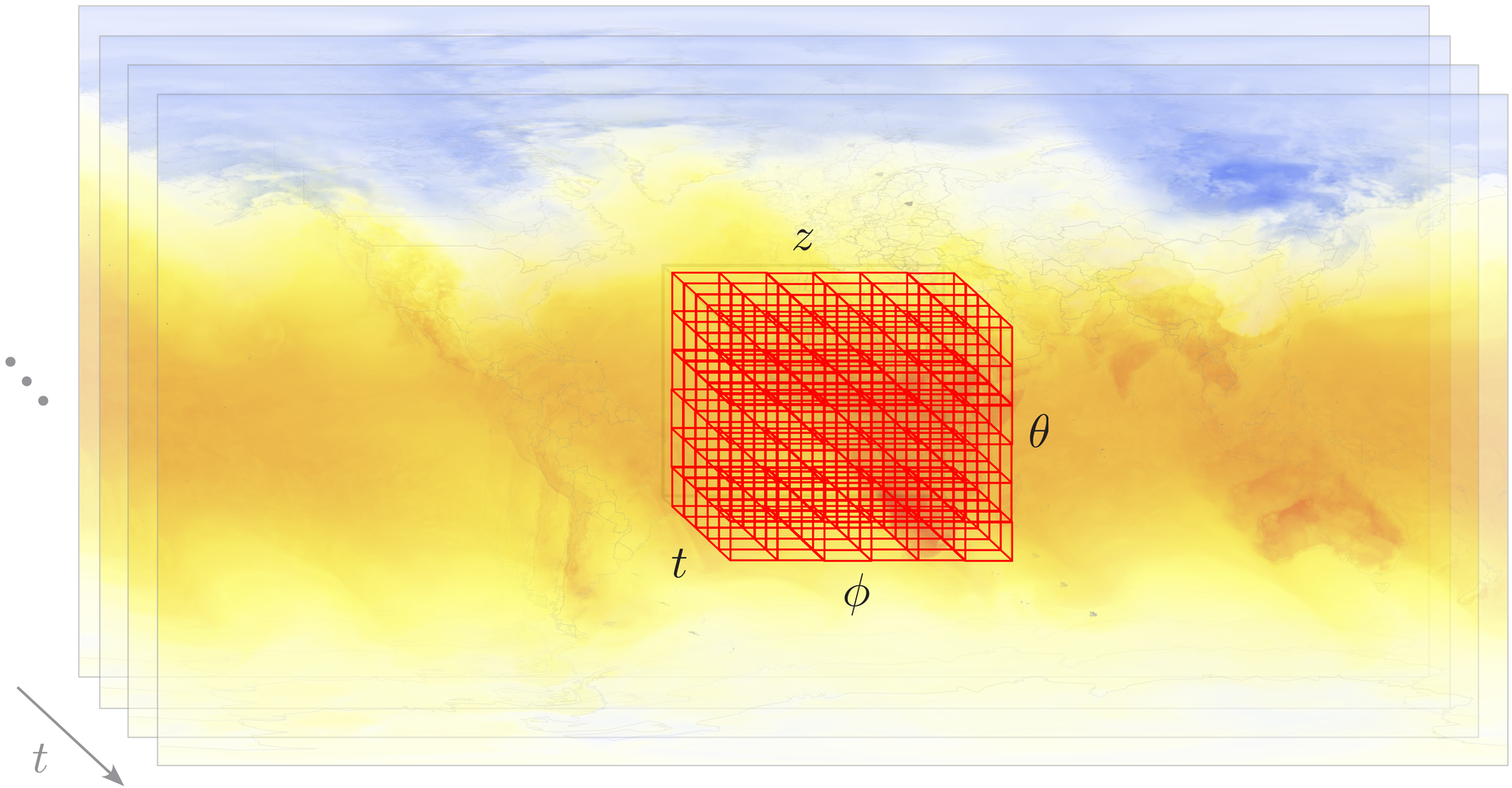
Training of embedding network

- Self-supervised training with variation of BERT masked language language model
  - Natural interpretation as forecasting / hindcasting / interpolation

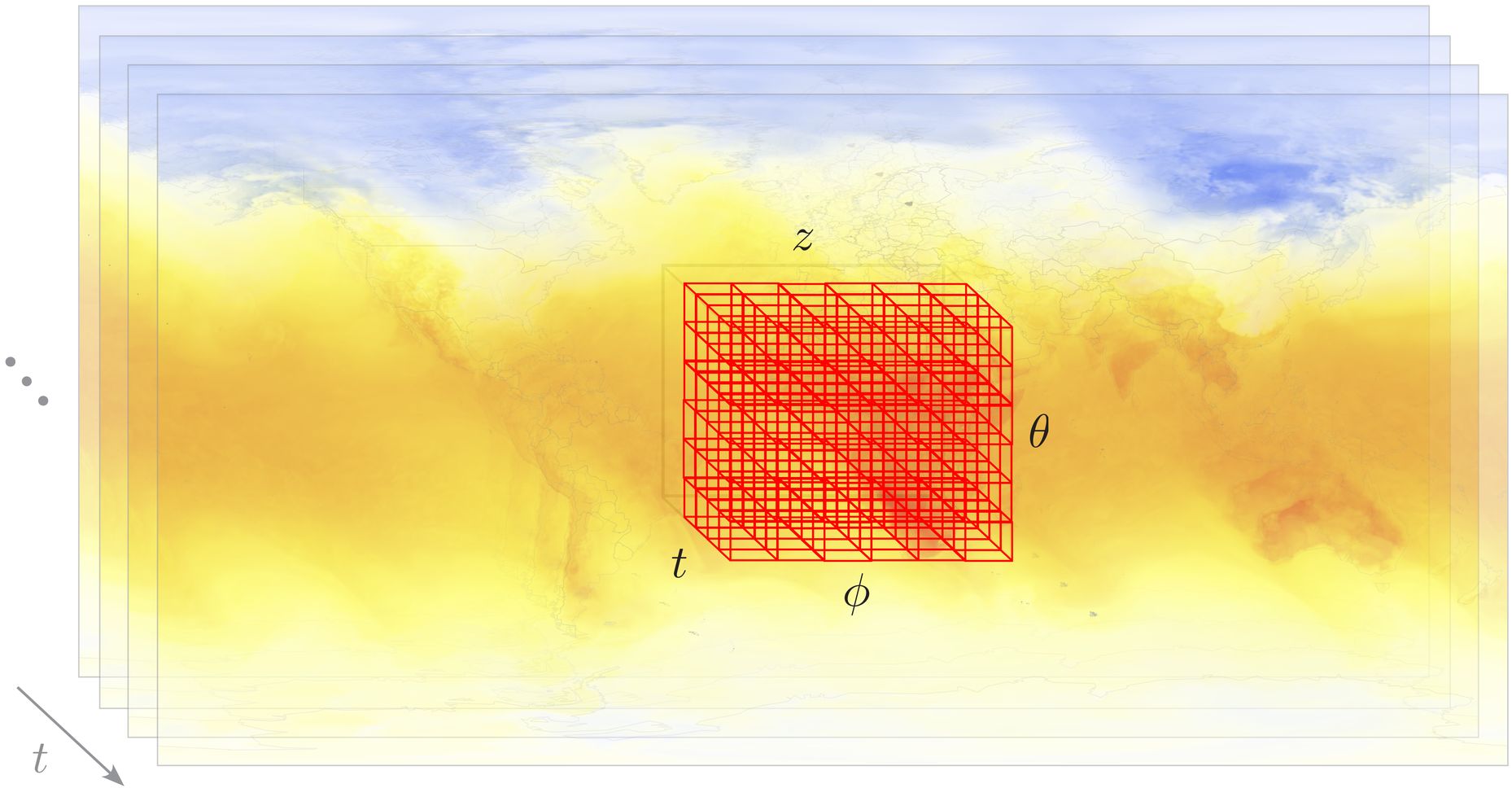
Training of embedding network

- Self-supervised training with variation of BERT masked language language model
  - Natural interpretation as forecasting / hindcasting / interpolation
  - Performed on randomly cropped subset to obtain consitent embedding network for different sized tokens

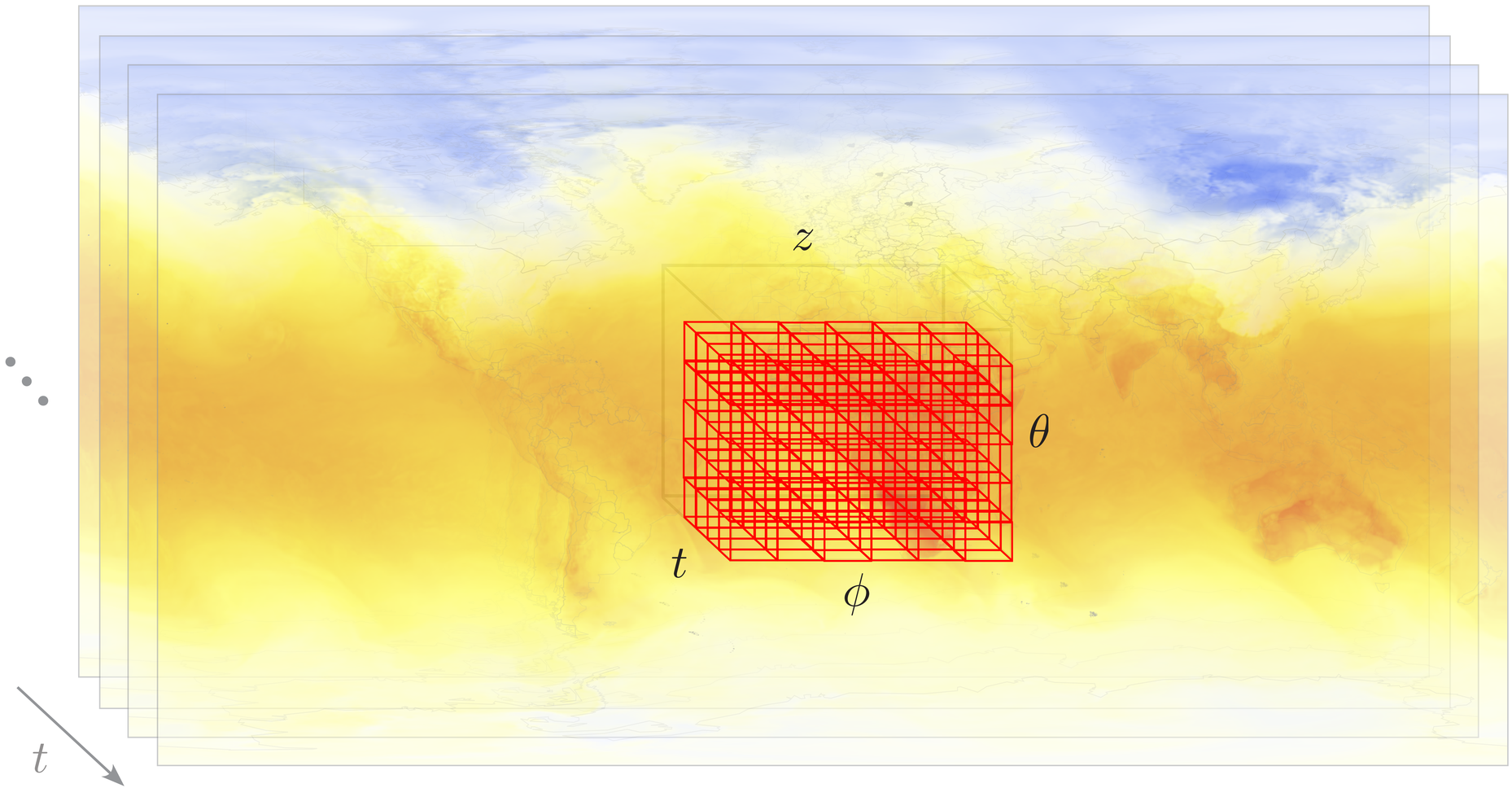
AtmoRep training



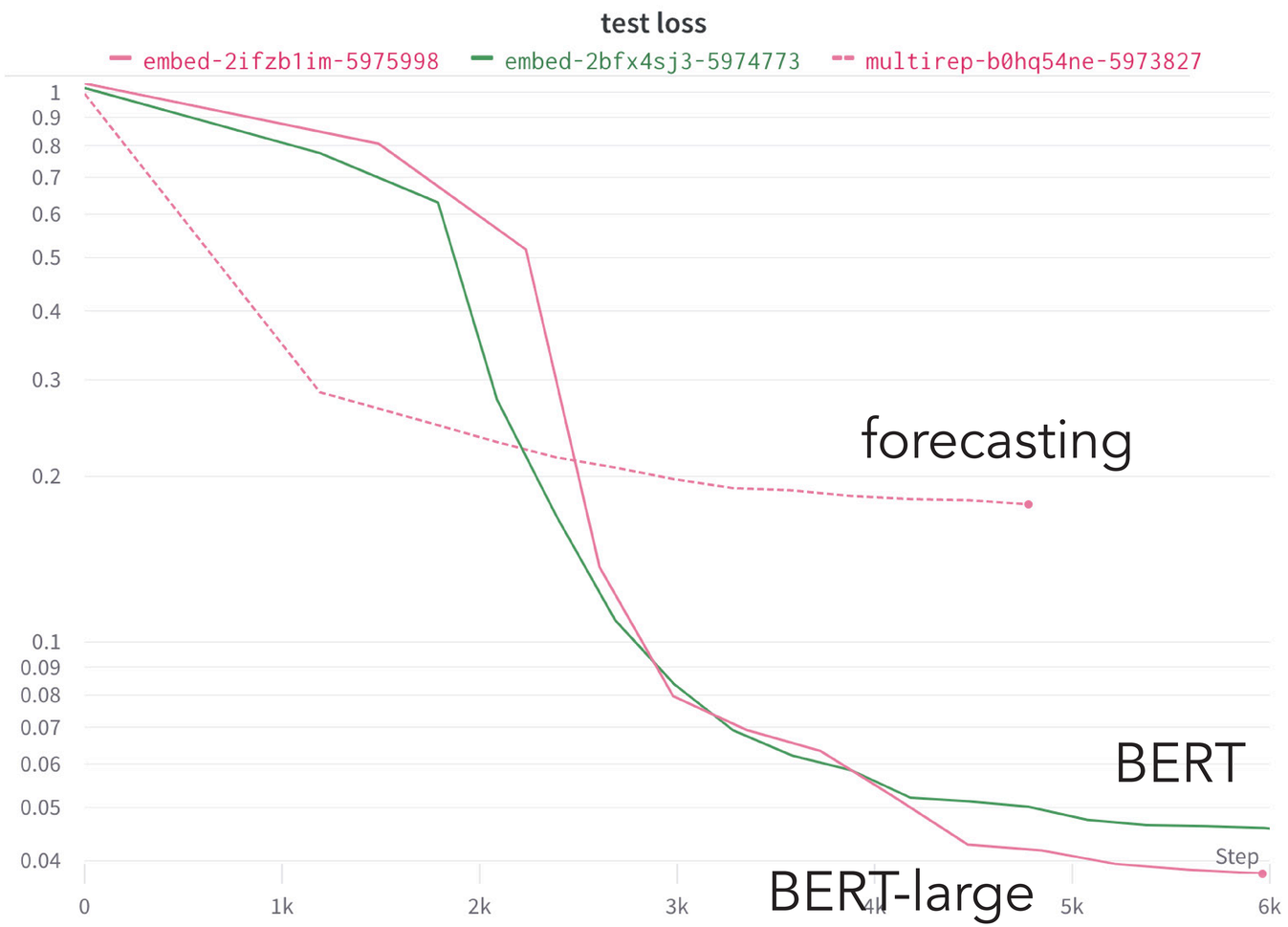
AtmoRep training



AtmoRep training



AtmoRep training

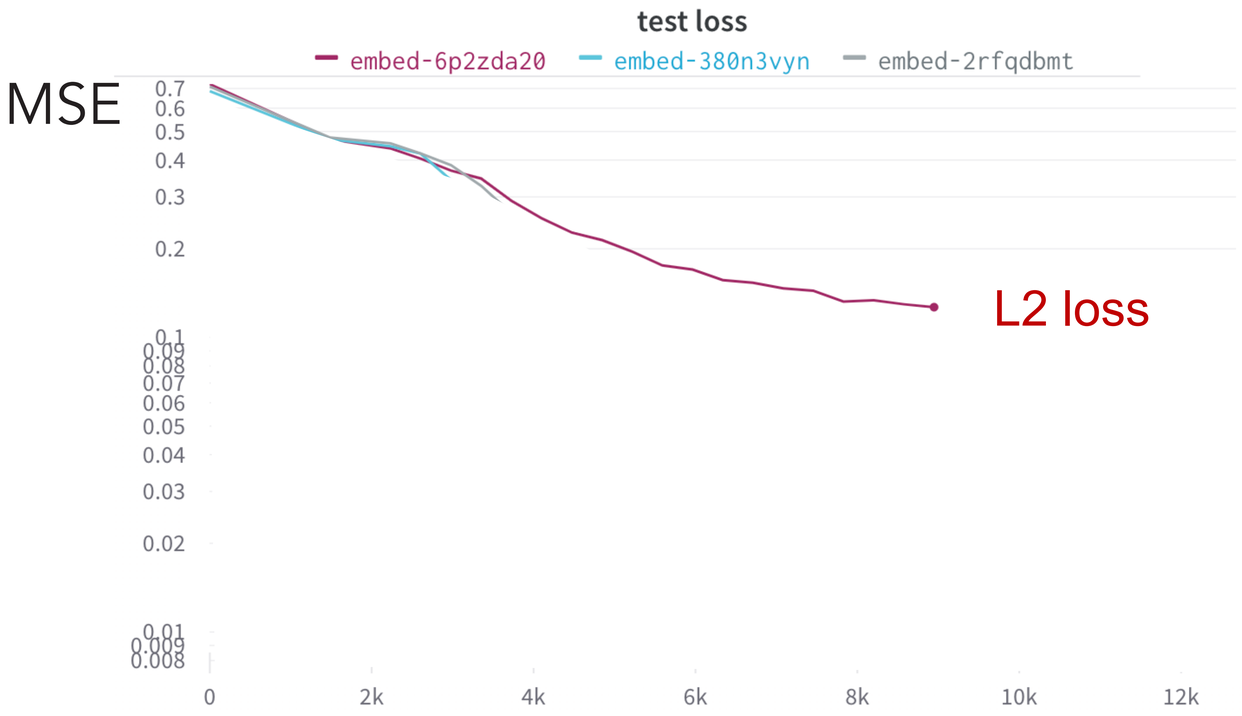


Statistical loss: respect the stochasticity

- Machine learning: Training on MSE/L_2 loss is problematic in terms of training dynamics

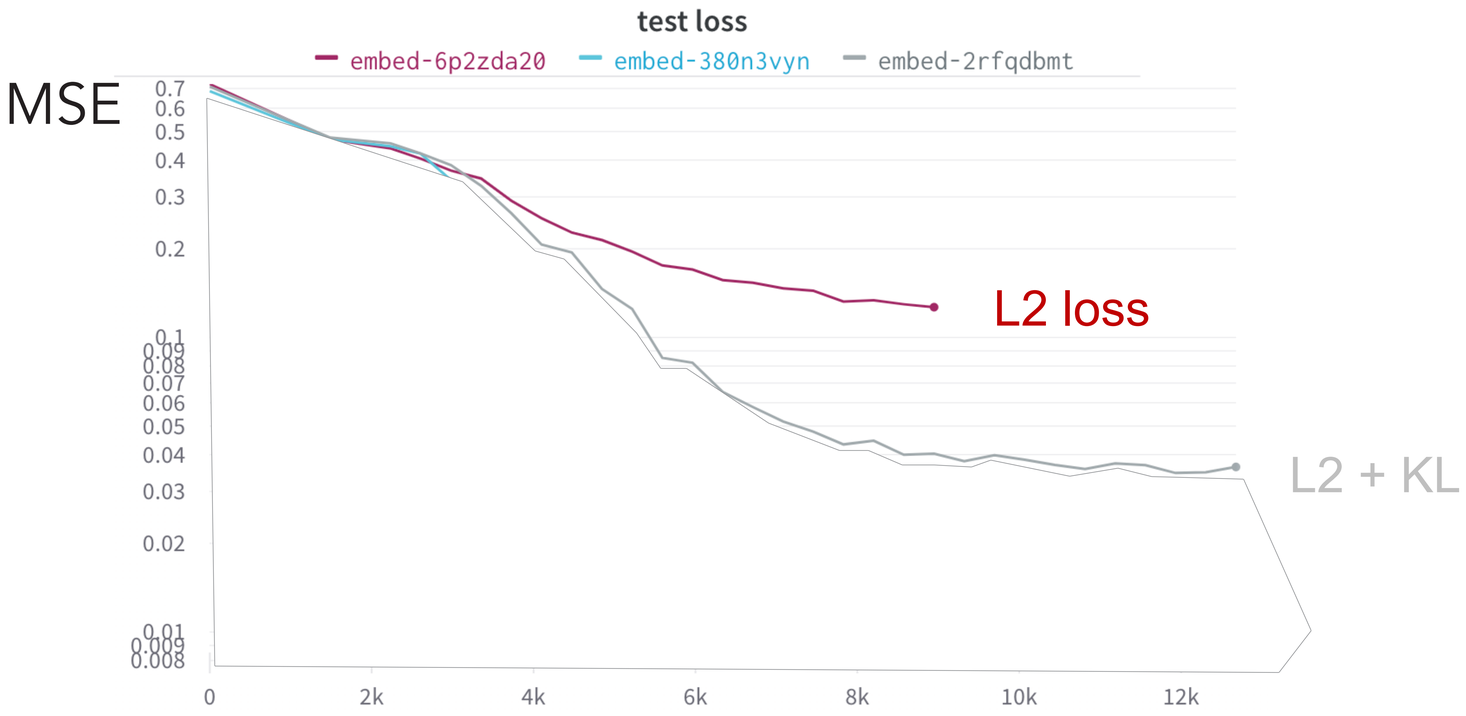
Statistical loss: respect the stochasticity

- Machine learning: Training on MSE/L_2 loss is problematic in terms of training dynamics



Statistical loss: respect the stochasticity

- Machine learning: Training on MSE/L_2 loss is problematic in terms of training dynamics



Statistical loss: respect the stochasticity

- Respect the stochasticity in the dynamics
  - ML: Training on MSE/L_2 loss is problematic in terms of training dynamics
  - Training on just the mean is sub-optimal to learn a probabilitic/statistical representation of the dynamics and the system

Statistical loss: respect the stochasticity

Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. 2014. How transferable are features in deep neural networks? In *Advances in neural information processing systems*, pages 3320–3328.

Adams Wei Yu, David Dohan, Minh-Thang Luong, Rui Zhao, Kai Chen, Mohammad Norouzi, and Quoc V Le. 2018. QANet: Combining local convolution with global self-attention for reading comprehension. In *ICLR*.

Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 2018. Swag: A large-scale adversarial dataset for grounded commonsense inference. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE international conference on computer vision*, pages 19–27.

Appendix for "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding"

- 80% of the time: Replace the word with the `[MASK]` token, e.g., `my dog is hairy` → `my dog is [MASK]`
- 10% of the time: Replace the word with a random word, e.g., `my dog is hairy` → `my dog is apple`
- 10% of the time: Keep the word unchanged, e.g., `my dog is hairy` → `my dog is hairy`. The purpose of this is to bias the representation towards the actual observed word.

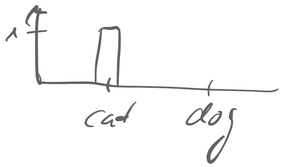
The advantage of this procedure is that the Transformer encoder does not know which words it will be asked to predict or which have been replaced by random words, so it is forced to keep a distributional contextual representation of *every* input token. Additionally, because random replacement only occurs for 1.5% of all tokens (i.e., 10% of 15%), this does not seem to harm the model's language understanding capability. In Section C.2, we evaluate the impact this procedure.

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

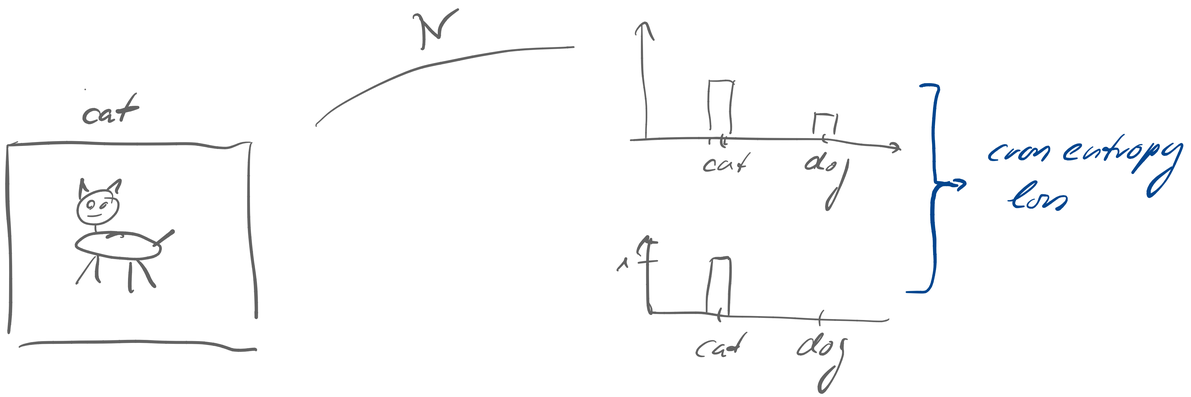
Statistical loss: respect the stochasticity

- How to obtain better training dynamics and ensure probabilistic/statistical representation in network?

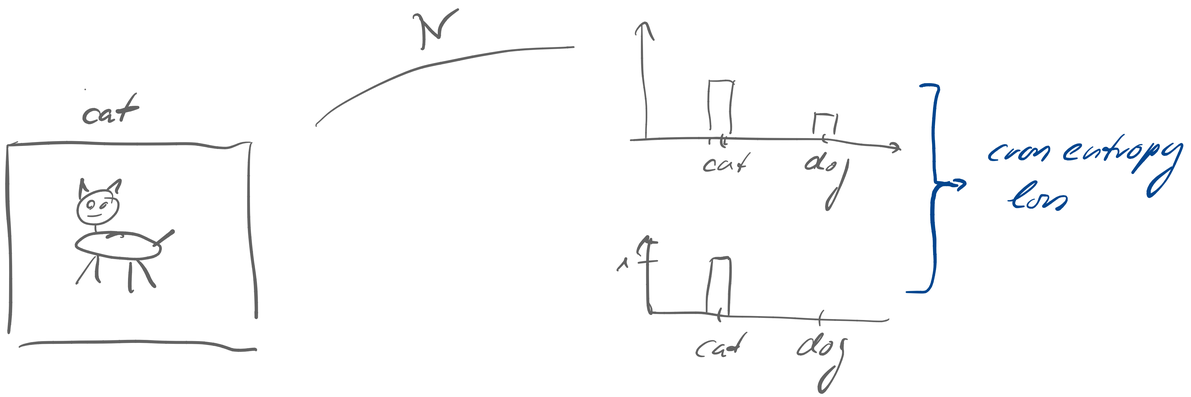
Statistical loss: respect the stochasticity

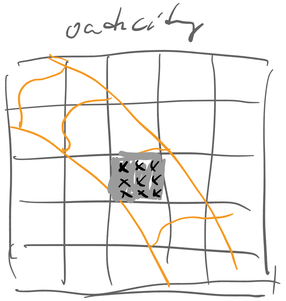


Statistical loss: respect the stochasticity

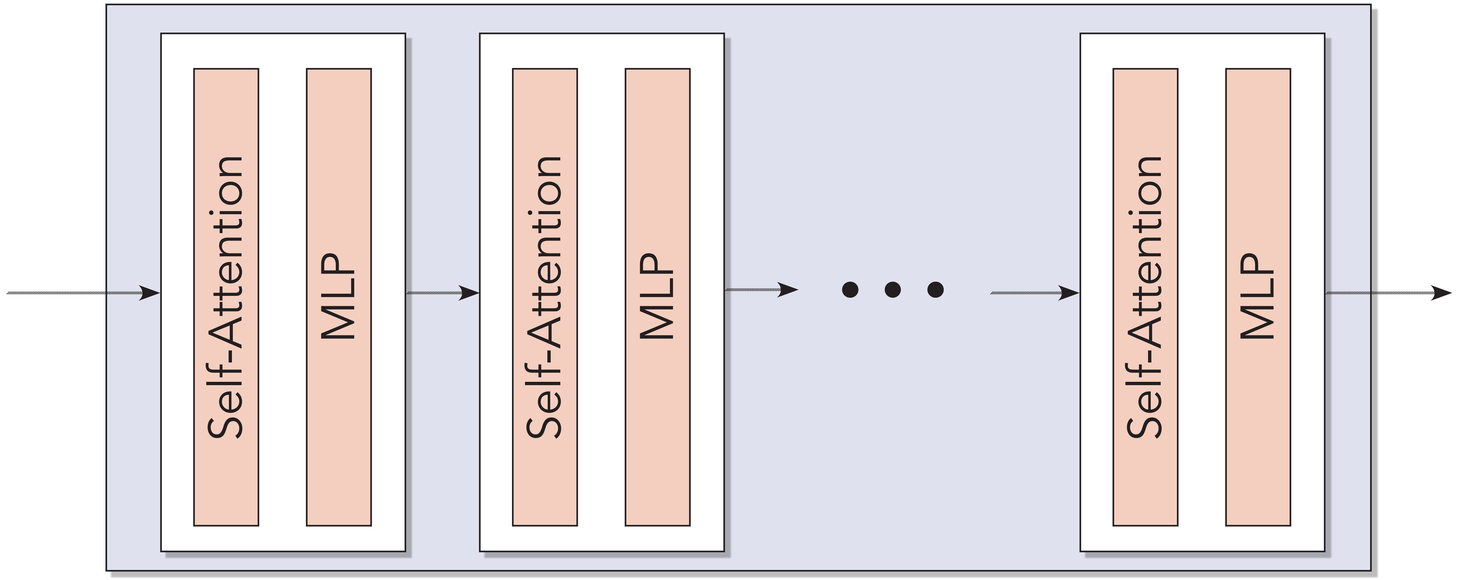


Statistical loss: respect the stochasticity

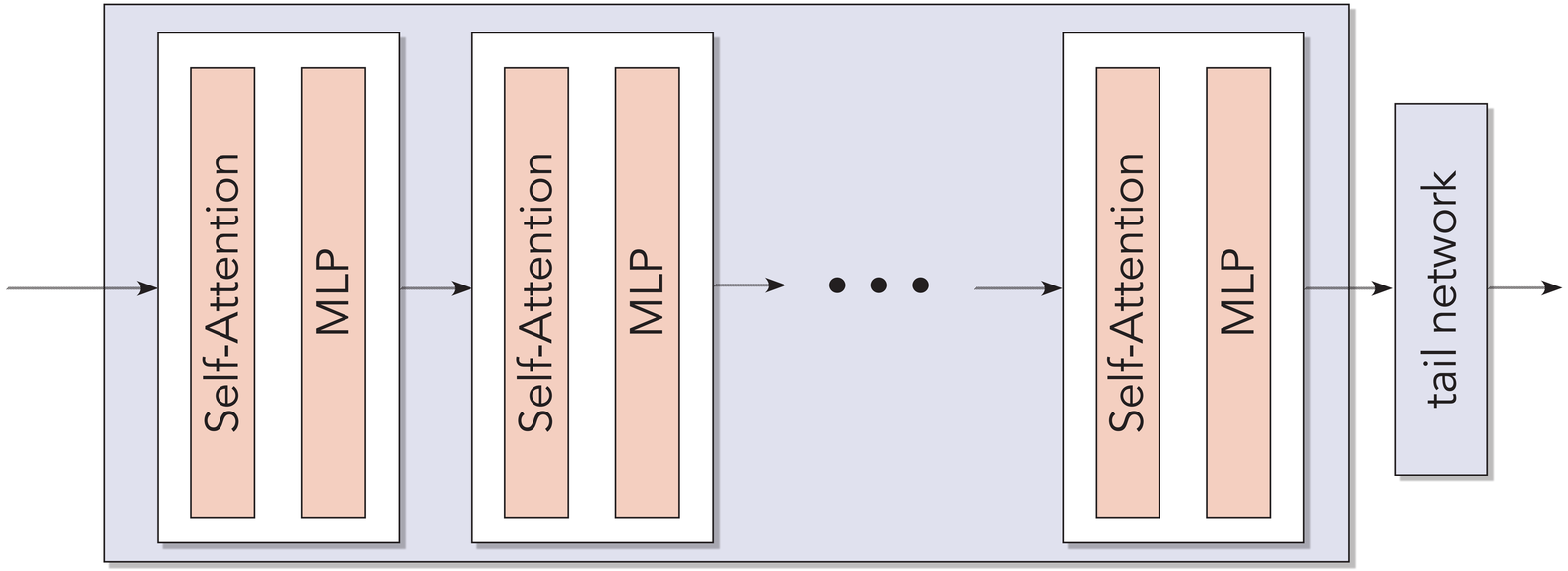




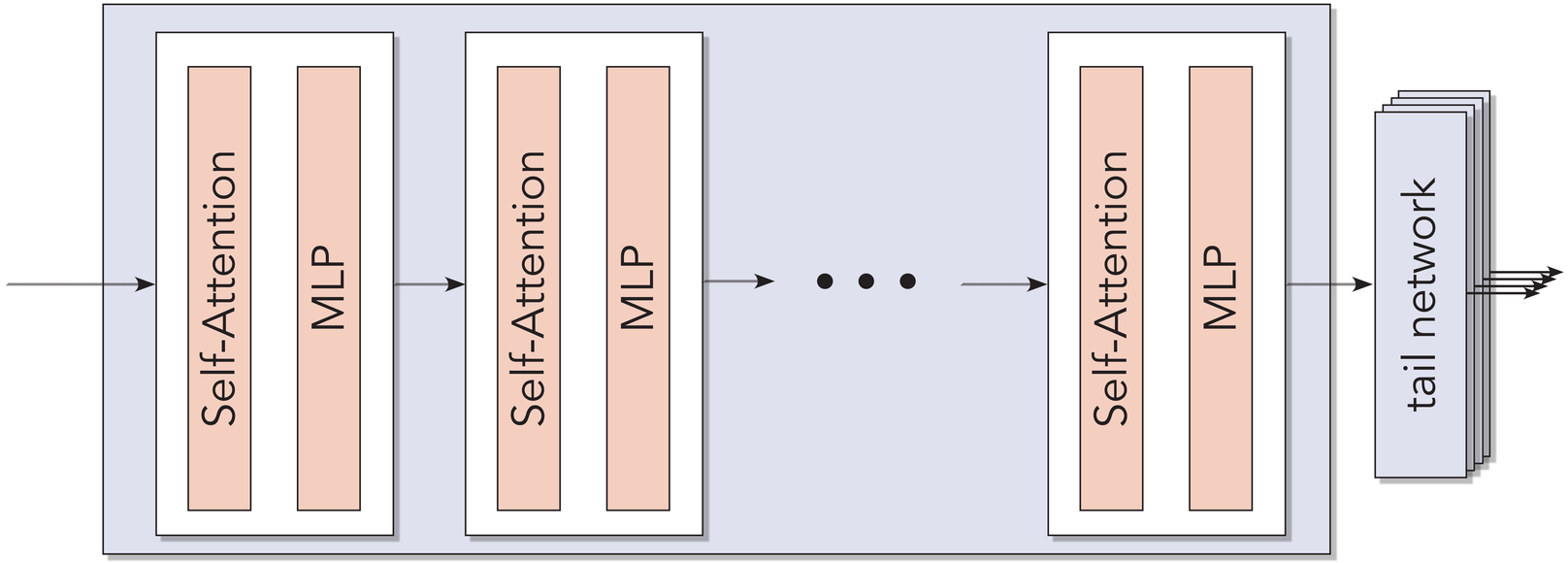
Statistical loss: respect the stochasticity



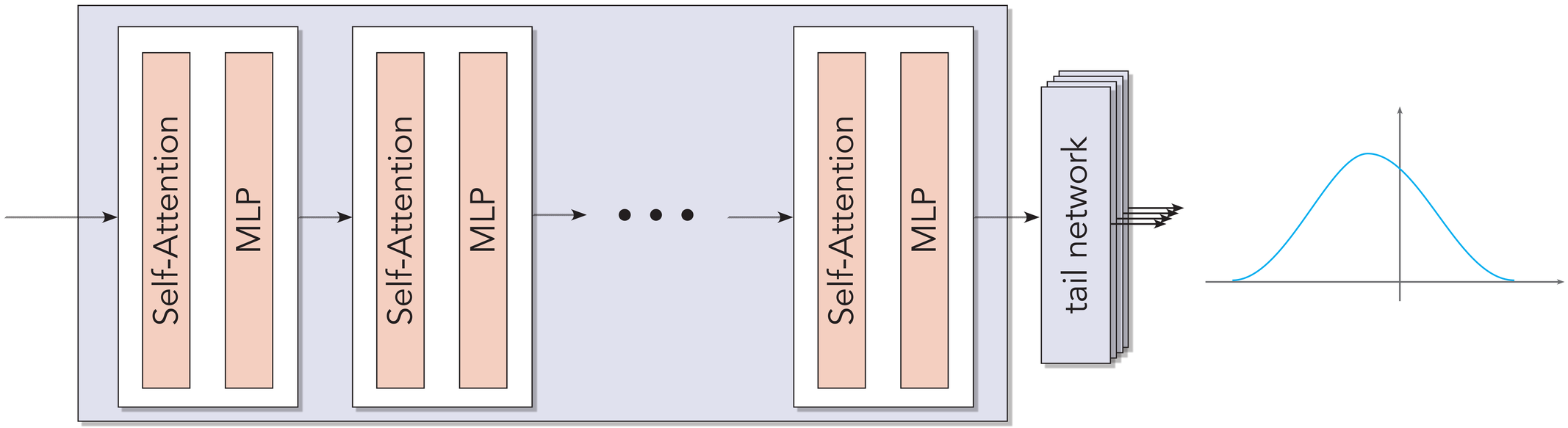
Statistical loss: respect the stochasticity



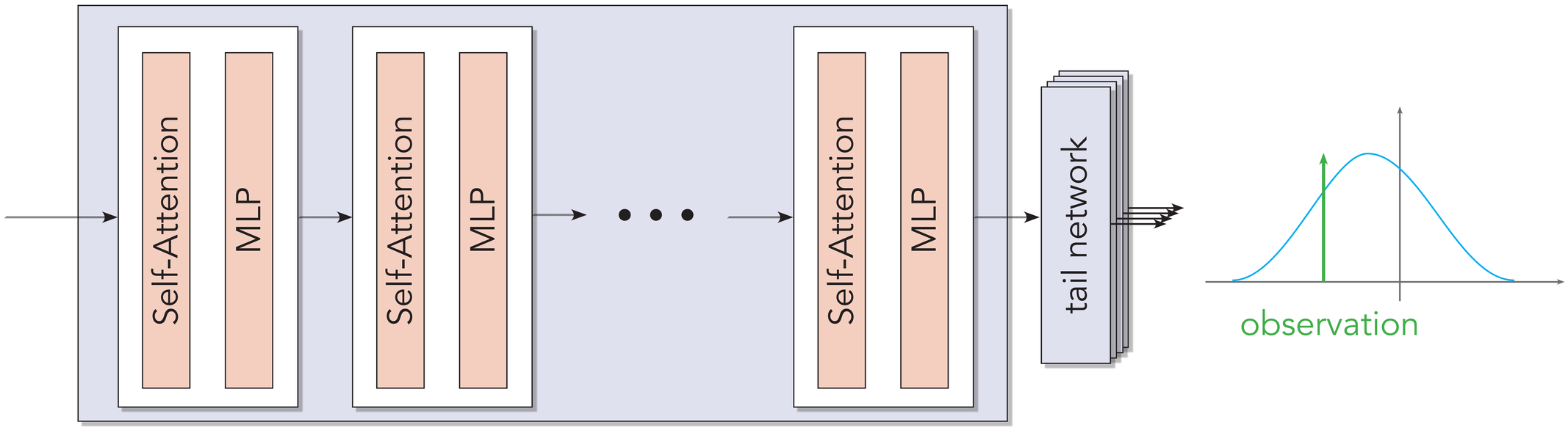
Statistical loss: respect the stochasticity



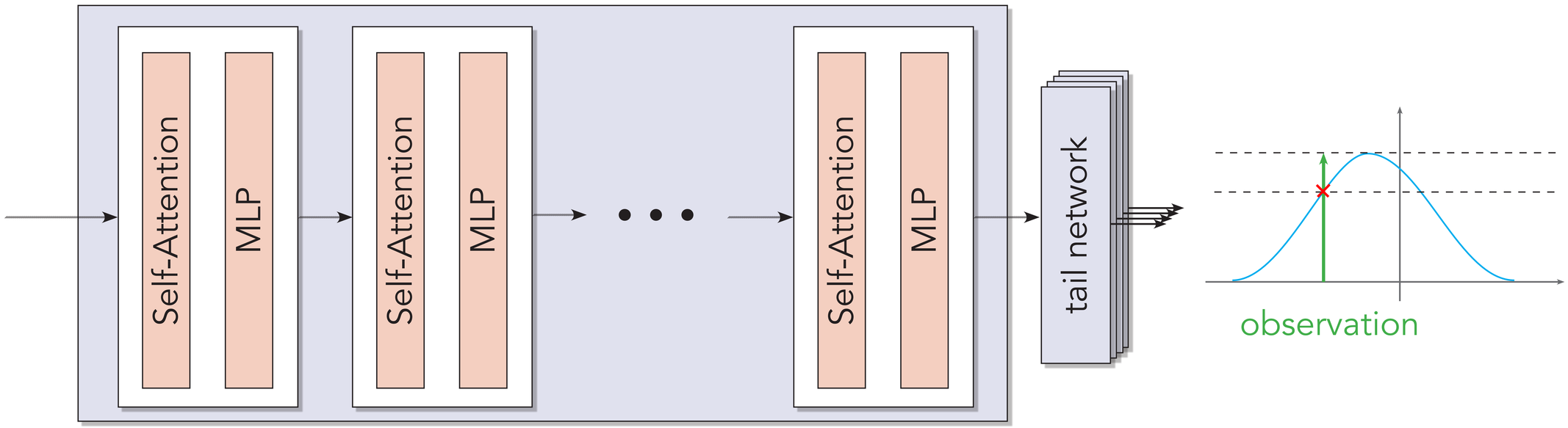
Statistical loss: respect the stochasticity



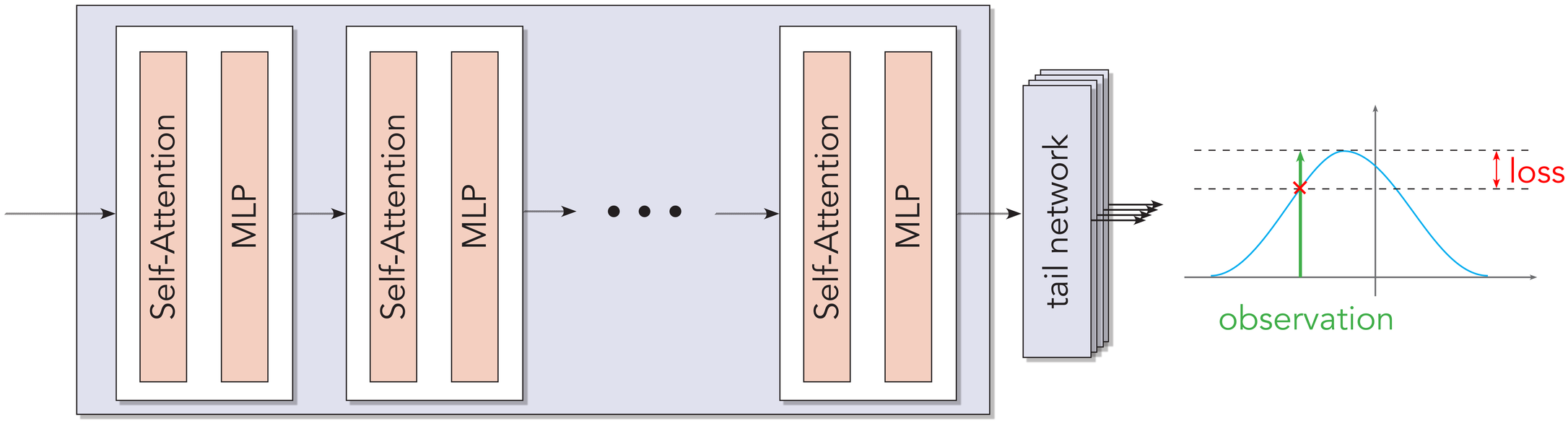
Statistical loss: respect the stochasticity



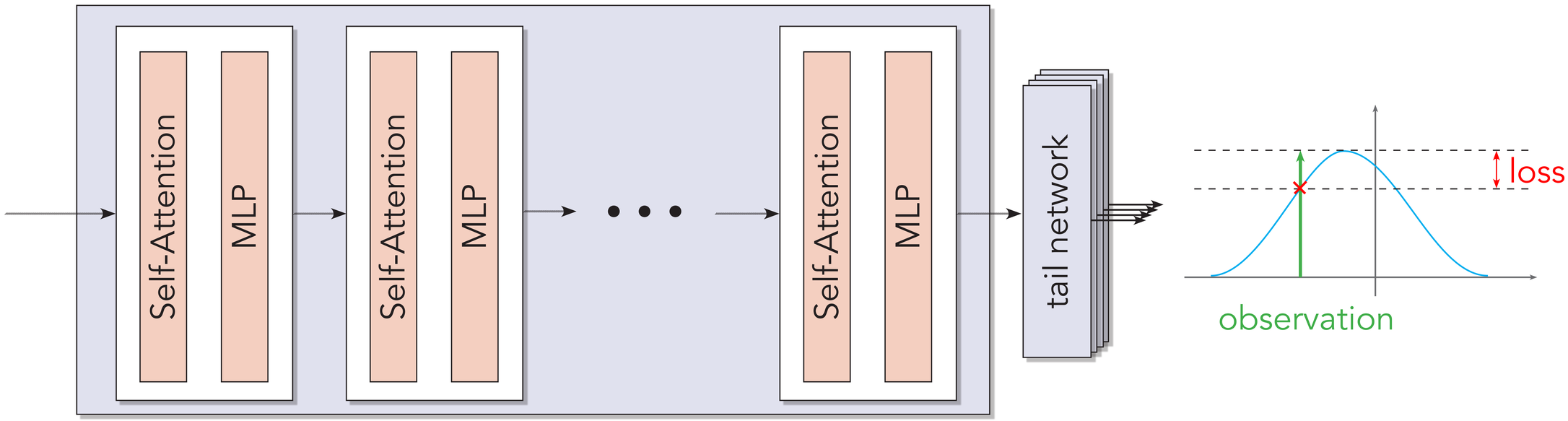
Statistical loss: respect the stochasticity



Statistical loss: respect the stochasticity



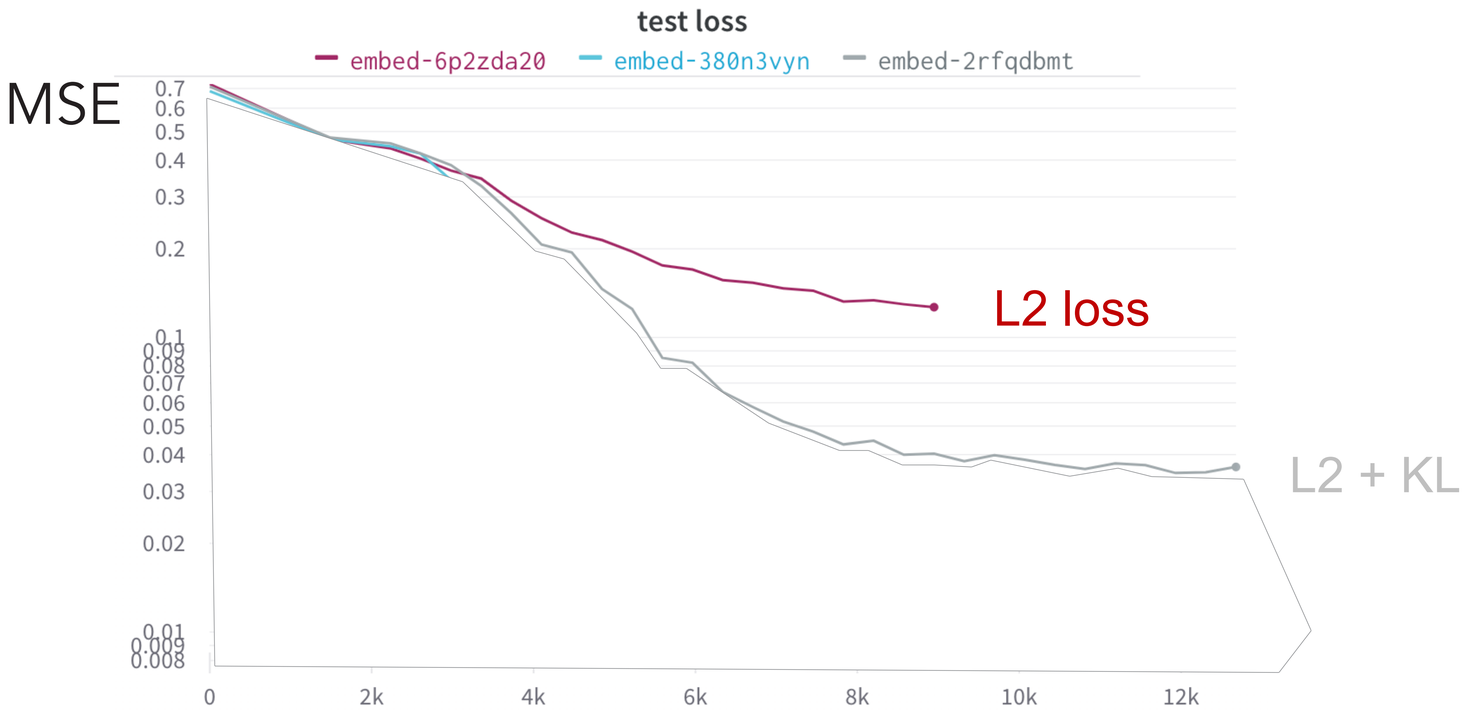
Statistical loss: respect the stochasticity



training encourages statistical representation

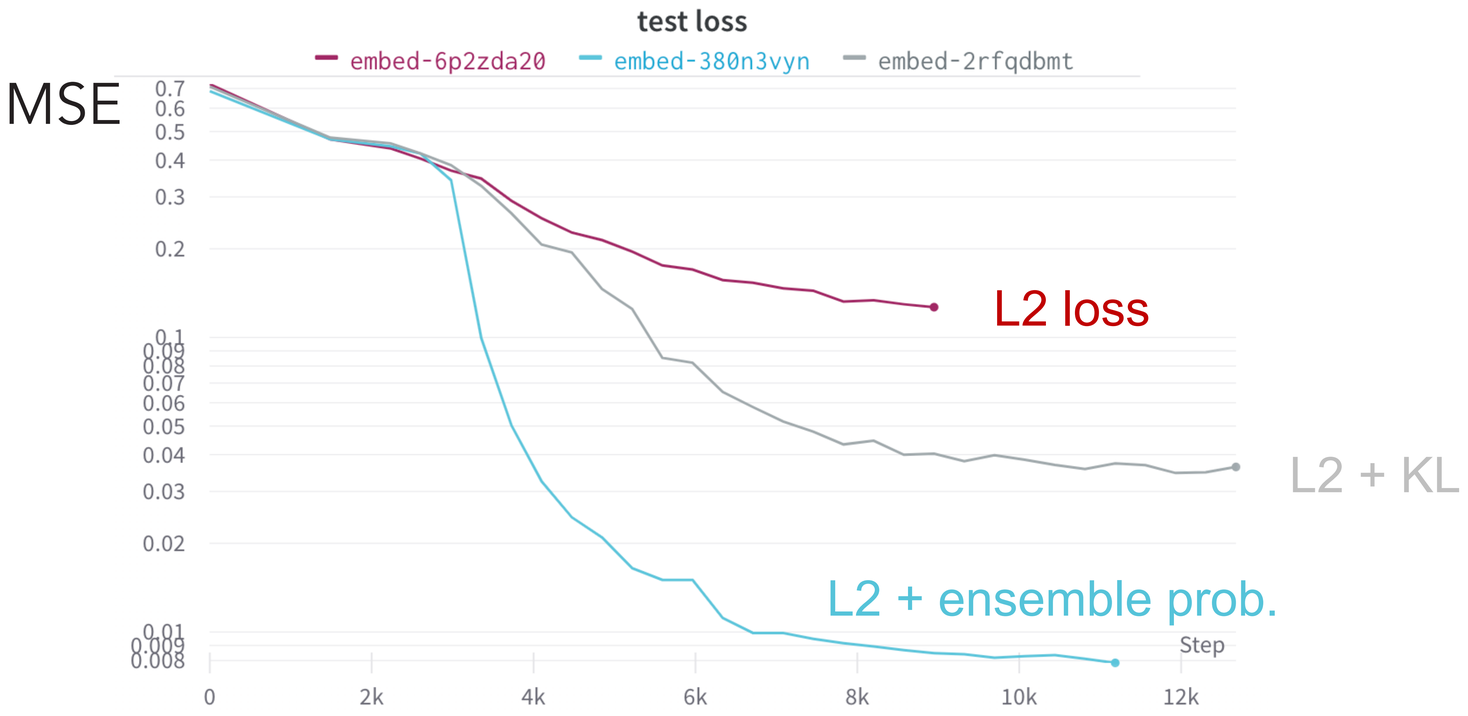
Statistical loss: respect the stochasticity

- Machine learning: Training on MSE/L_2 loss is problematic in terms of training dynamics

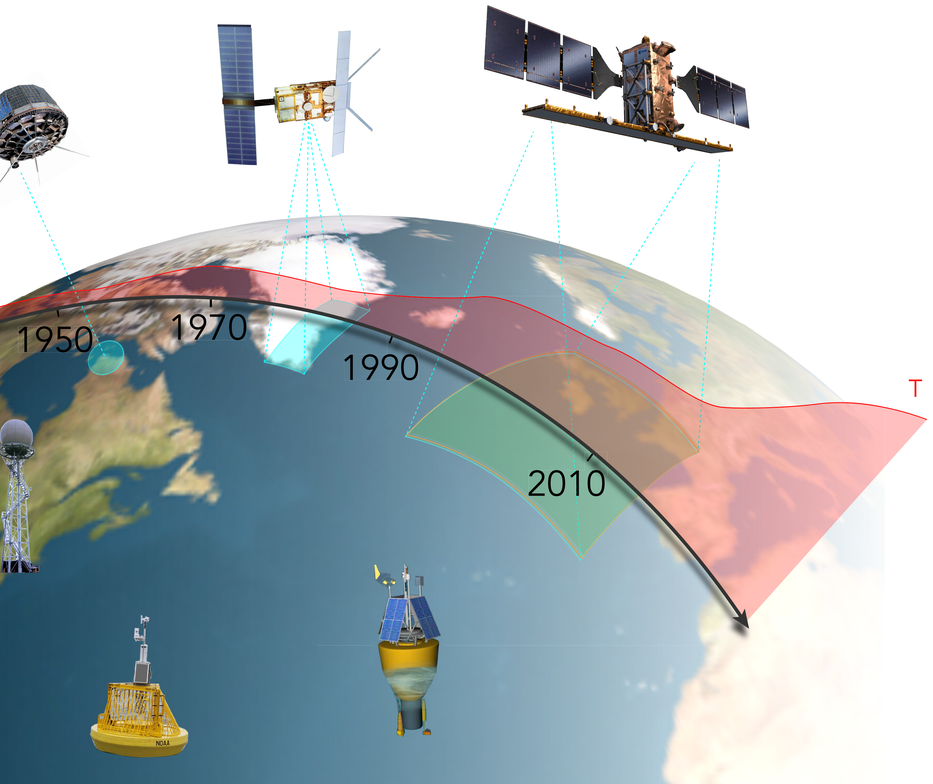


Statistical loss: respect the stochasticity

- Machine learning: Training on MSE/L_2 loss is problematic in terms of training dynamics



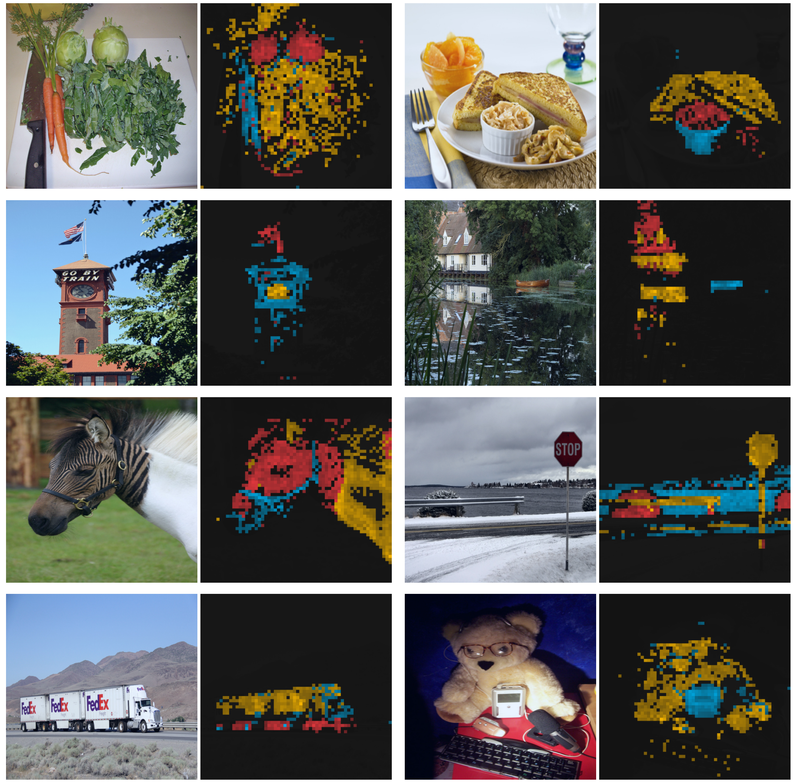
AtmoRep



- Large scale representation learning for atmospheric data
- Machine learning methodology needs to be completed and many open questions to be investigated
- Forecasting and projections
- How to integrate “raw” observations?
- Coupled atmosphere+ocean system?
- …

Network: attention maps

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, Advances in Neural Information Processing Systems, volume 30. Curran Associates, Inc., 2017.



M. Caron, H. Touvron, I. Misra, H. Jegou, J. Mairal, P. Bojanowski, and A. Joulin. Emerging properties in self-supervised vision transformers. CoRR, abs/2104.14294, 2021.

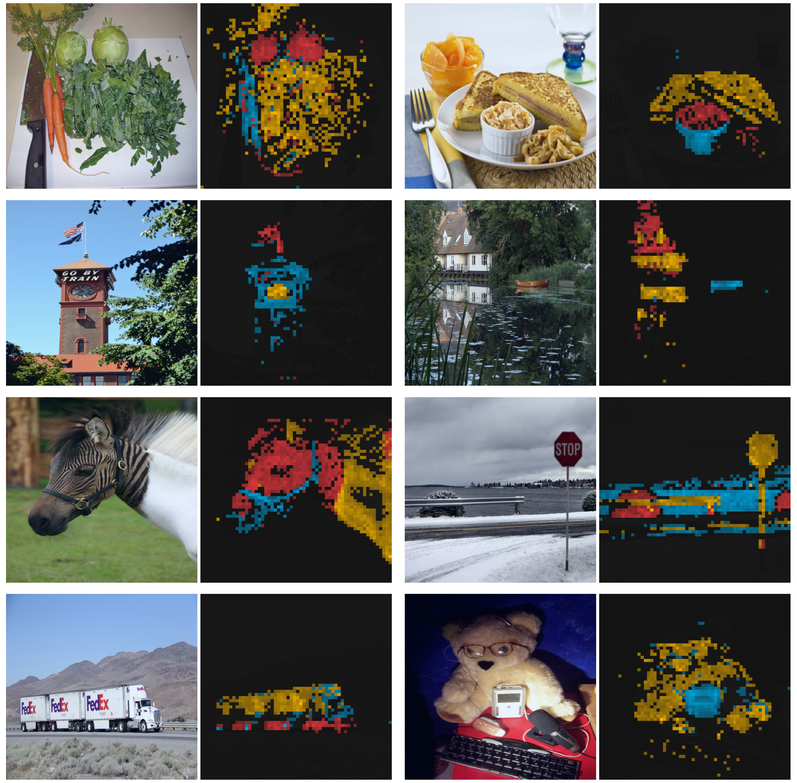
Statistical

Self-supervised training

- Task: Predict atmospheric state at time *t* for a local neighborhood
- Use spatio-temporal neighborhood as data for prediction
  - 2-3 weeks for fine mesoscale predictions
  - Use data from hyperbolic cone around prediction target
  - All fields and all vertical layer (potentially masked for prediction task)

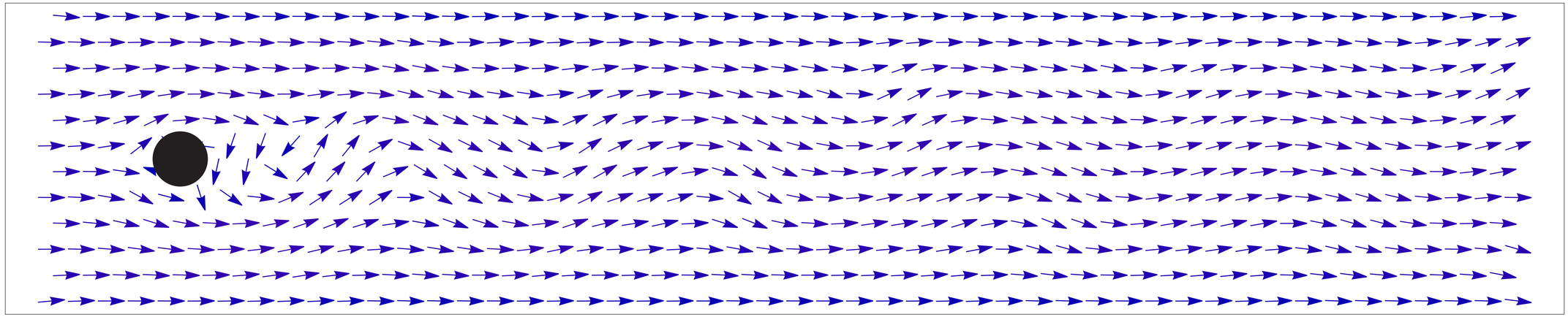
Transformers and attention

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, Advances in Neural Information Processing Systems, volume 30. Curran Associates, Inc., 2017.

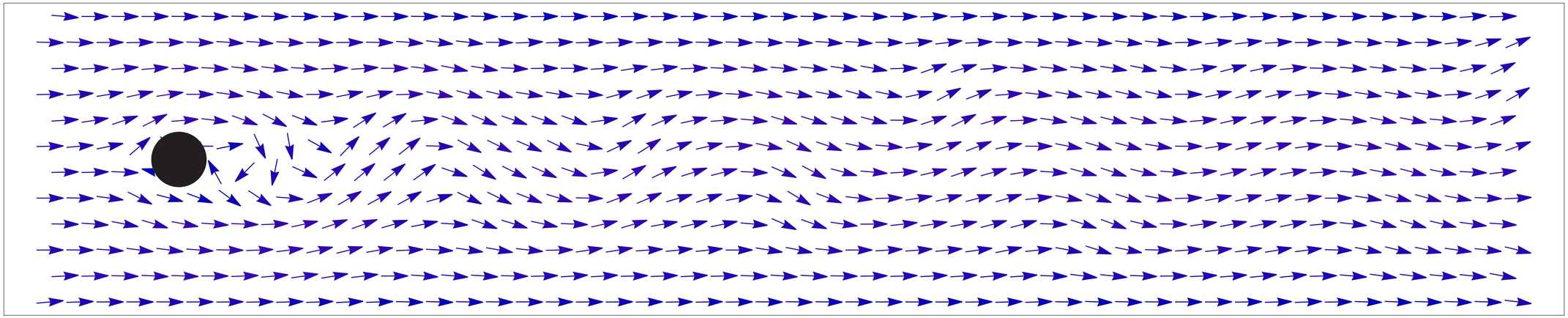


M. Caron, H. Touvron, I. Misra, H. Jegou, J. Mairal, P. Bojanowski, and A. Joulin. Emerging properties in self-supervised vision transformers. CoRR, abs/2104.14294, 2021.

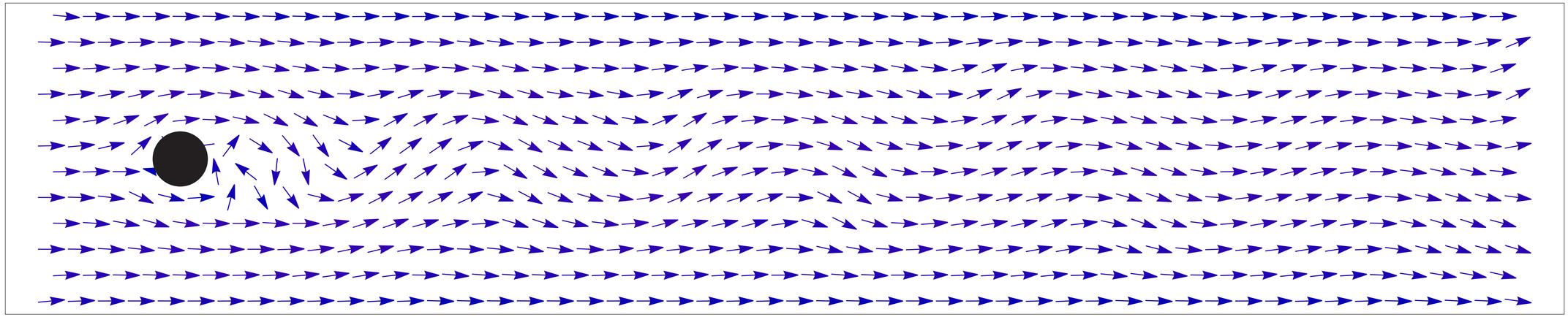
Transformers and attention



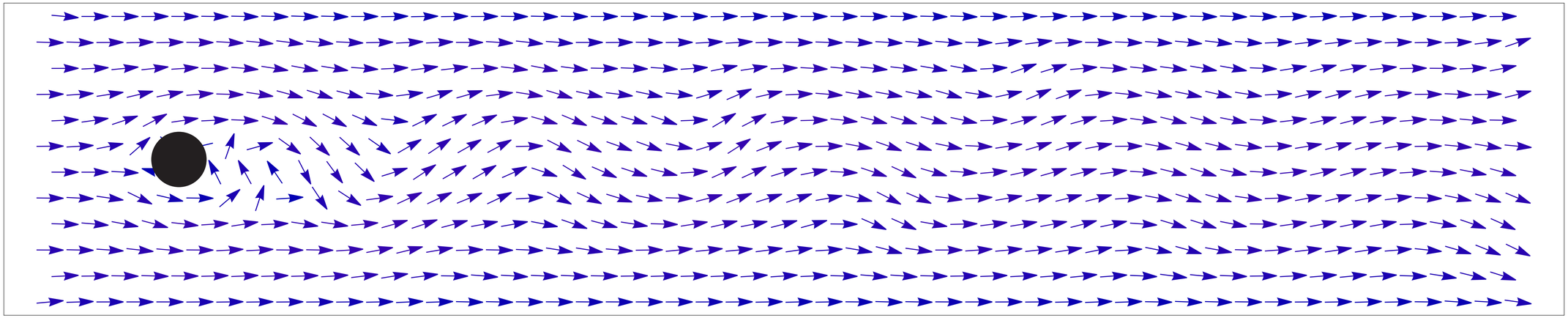
Transformers and attention



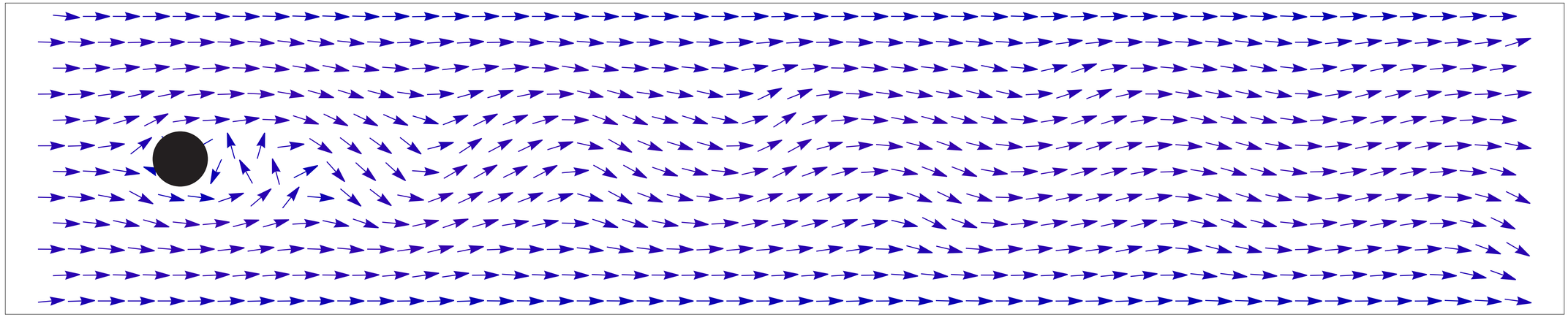
Transformers and attention



Transformers and attention



Transformers and attention



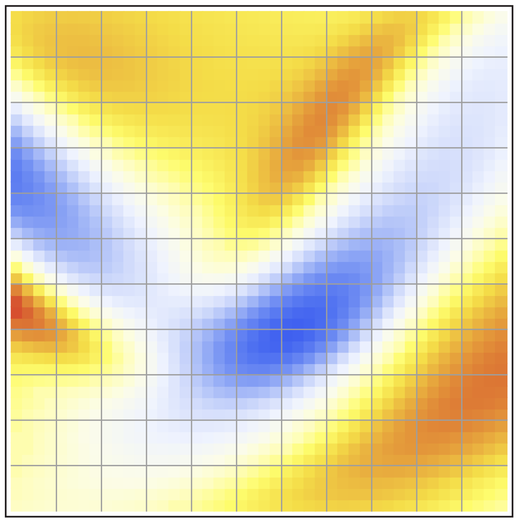
Fluid flow (vorticity)

Fluid flow (vorticity)

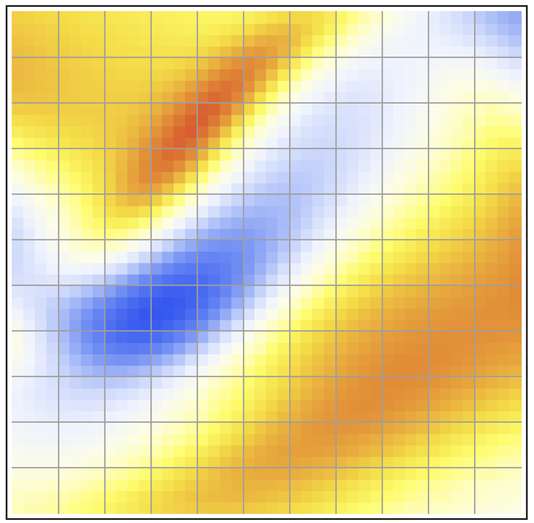
training with varying position
and spherical eccentrity

Fluid flow (vorticity)

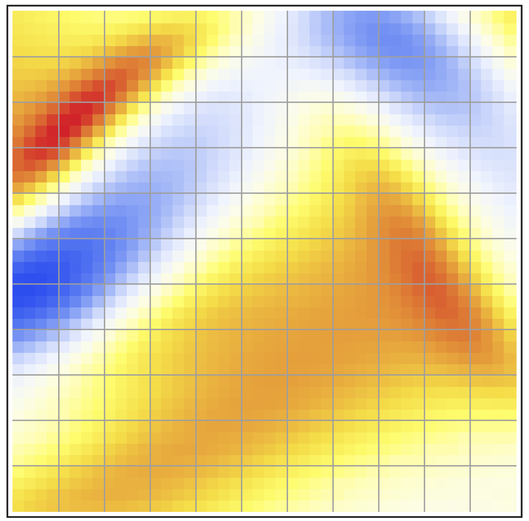
t-2



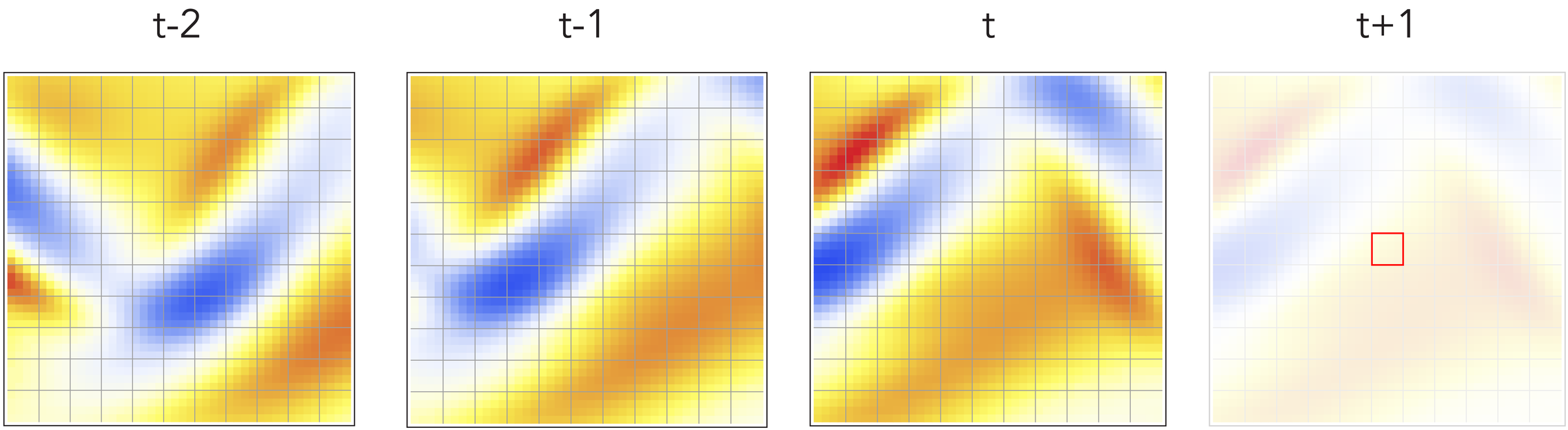
t-1



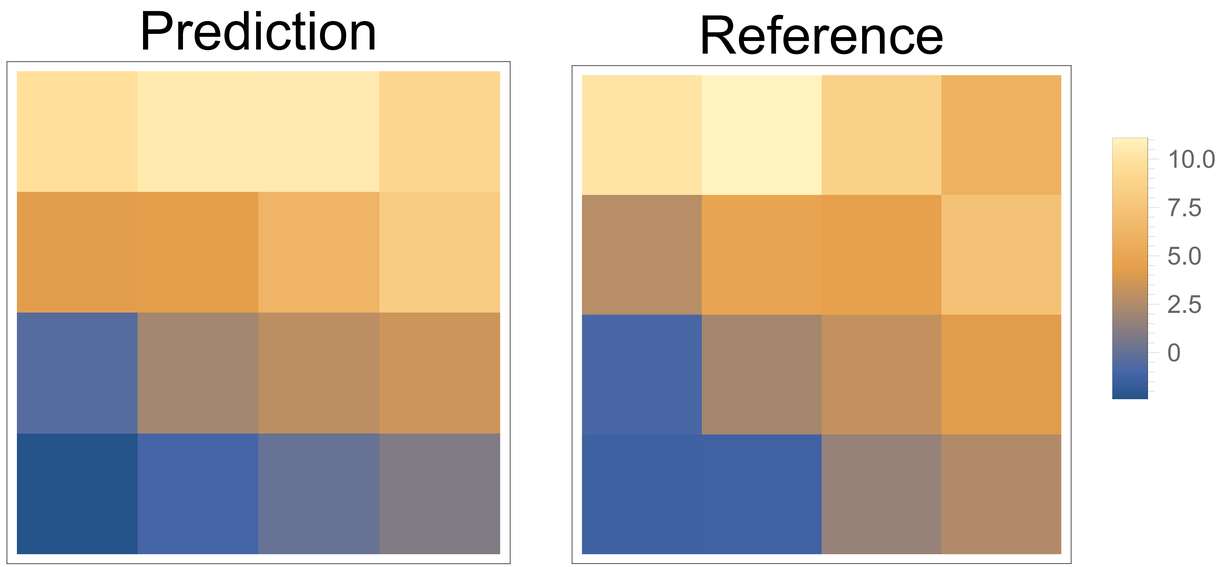
t

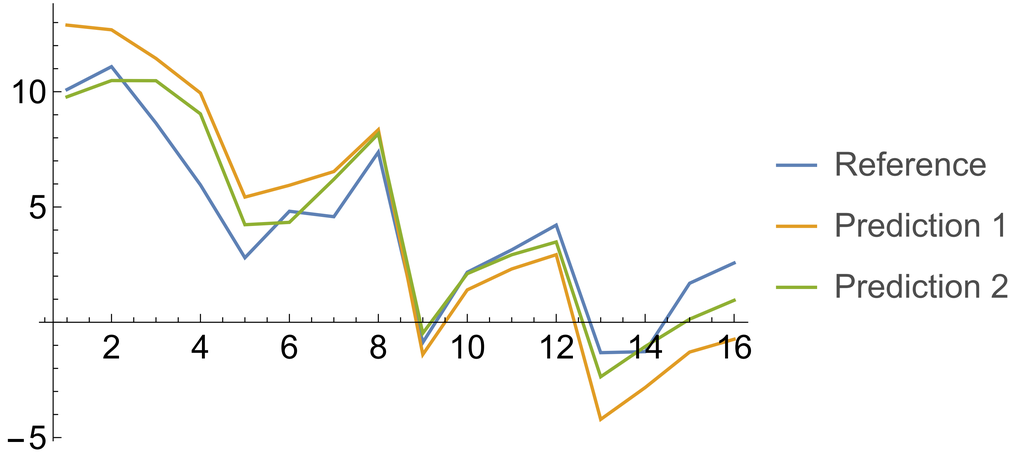


Fluid flow (vorticity)

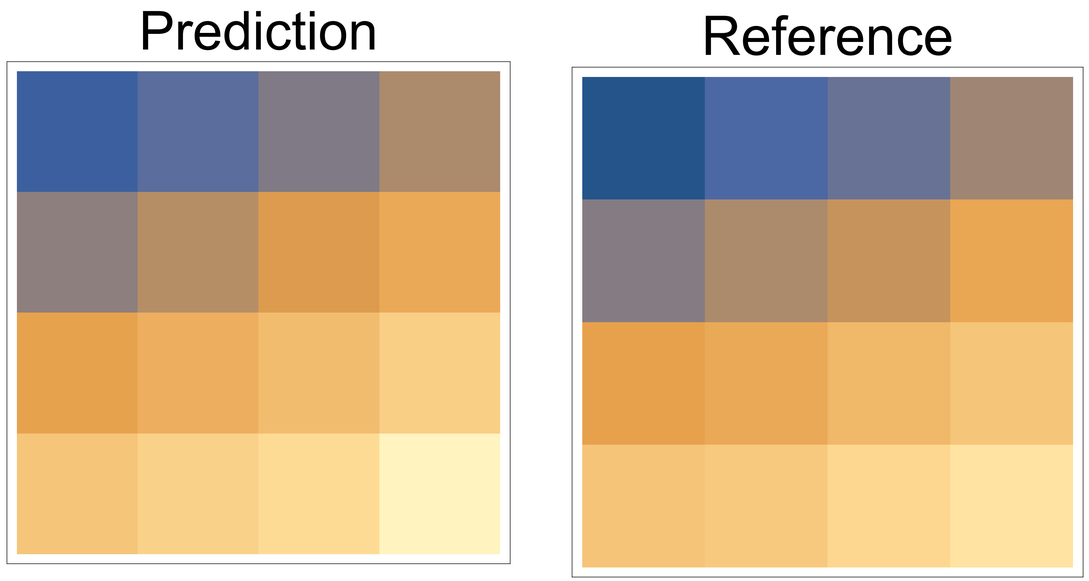


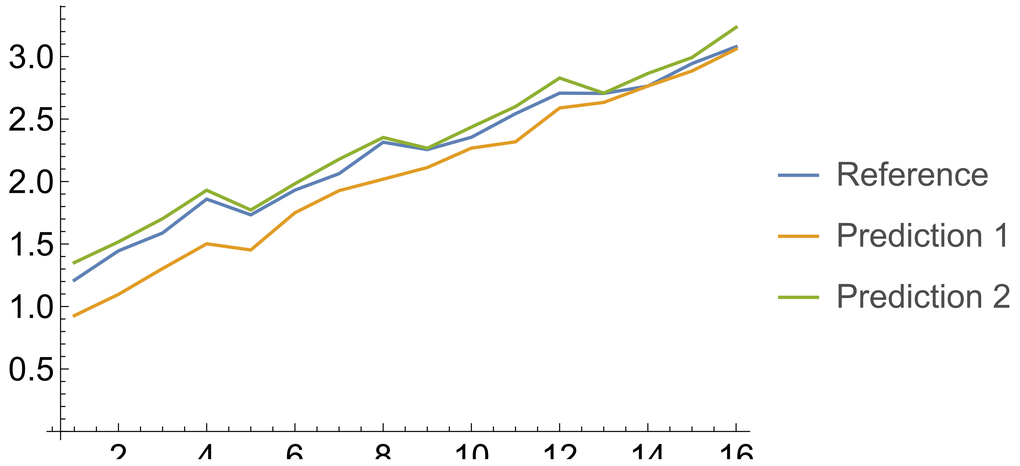
Fluid flow



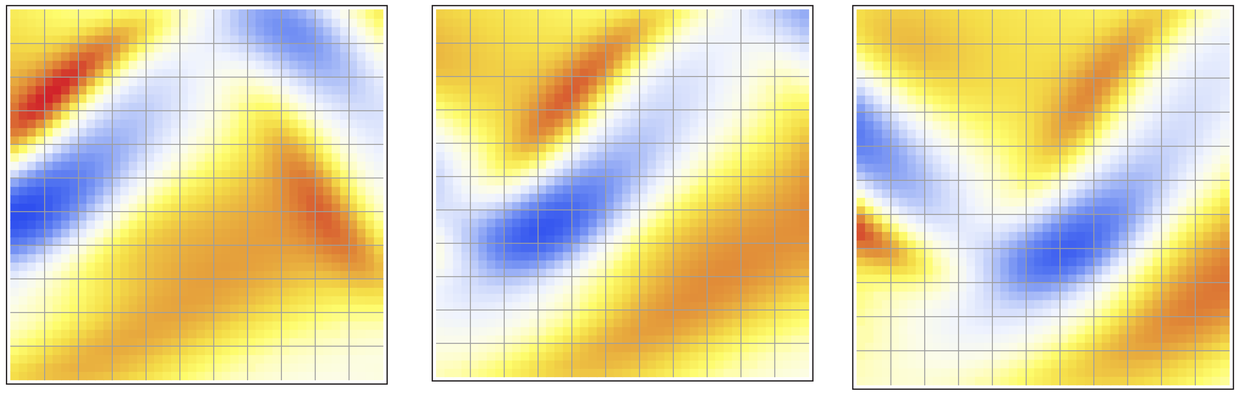


Fluid flow



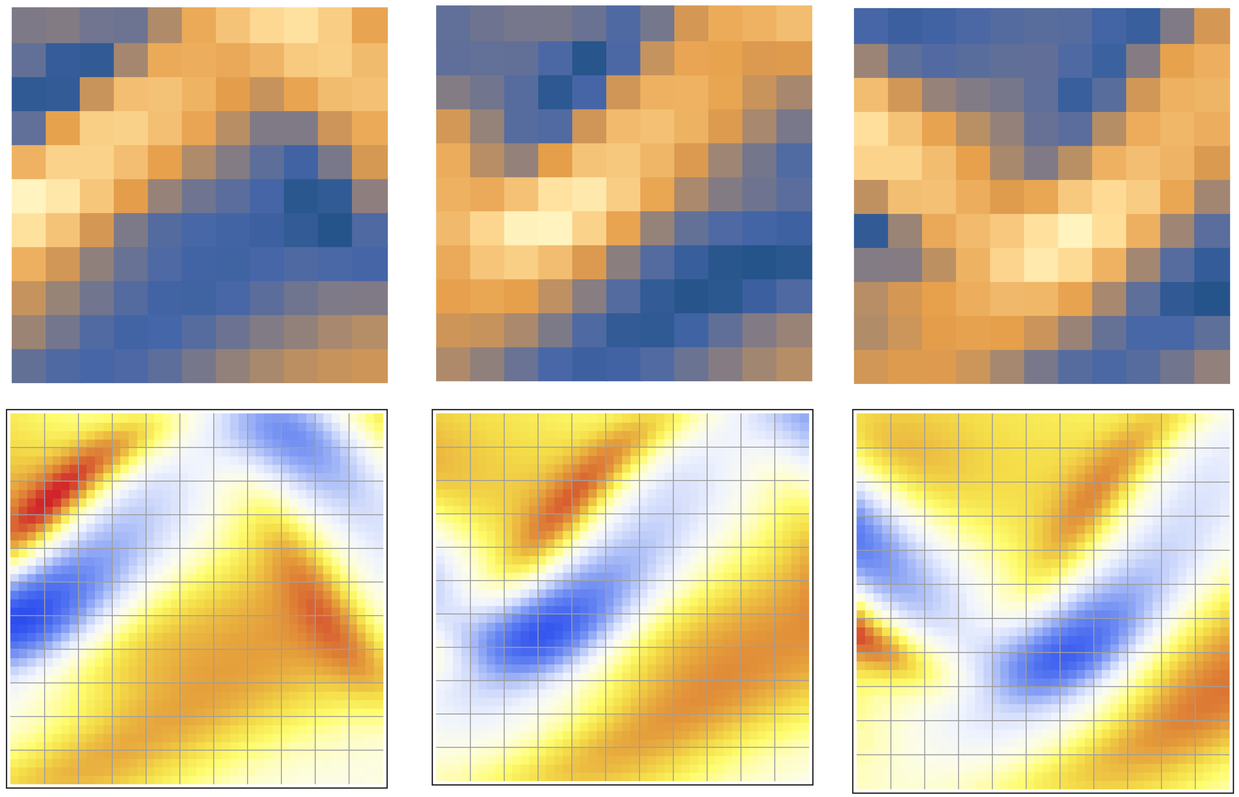


Fluid flow

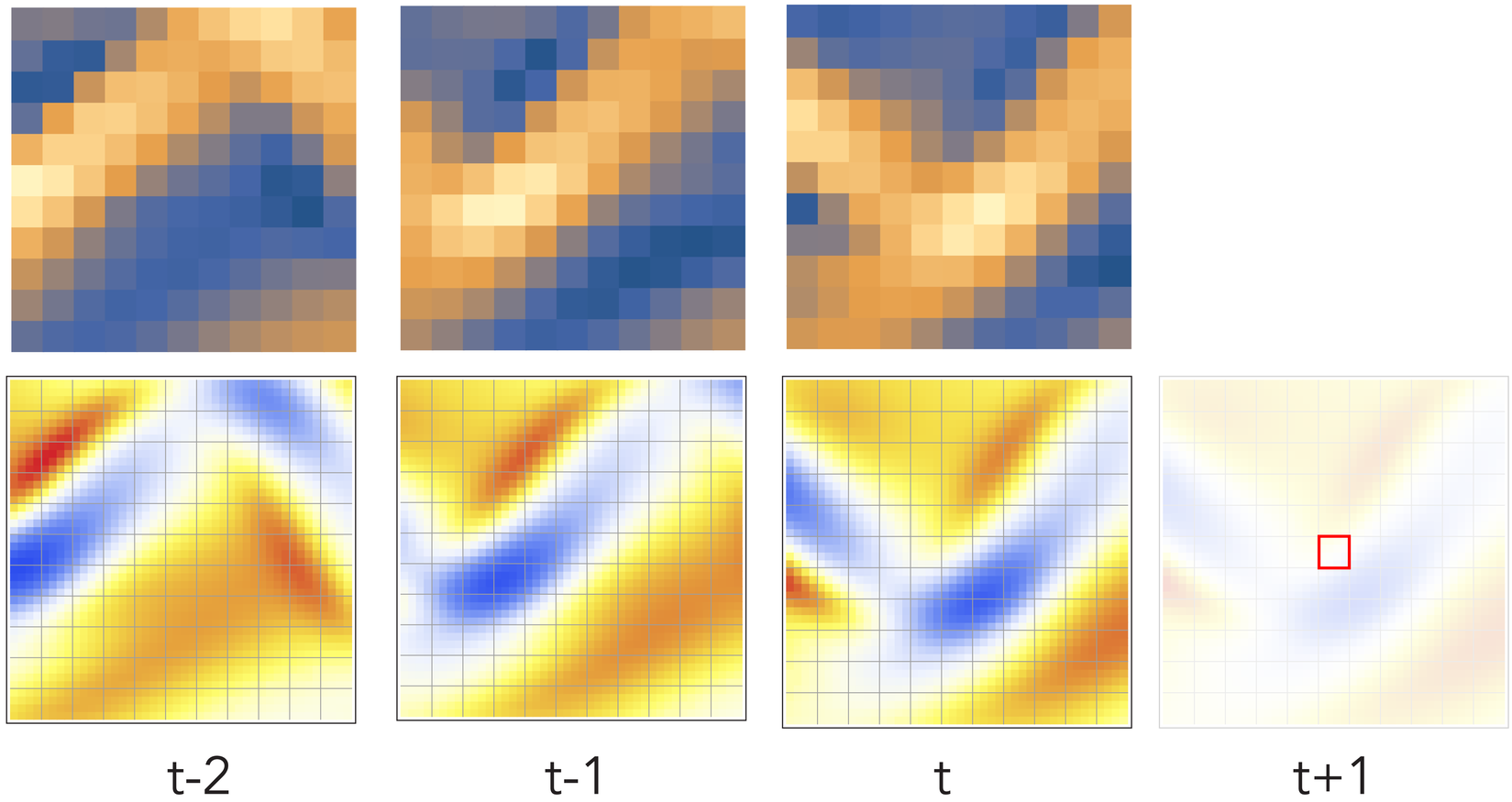


t-2 t-1 t

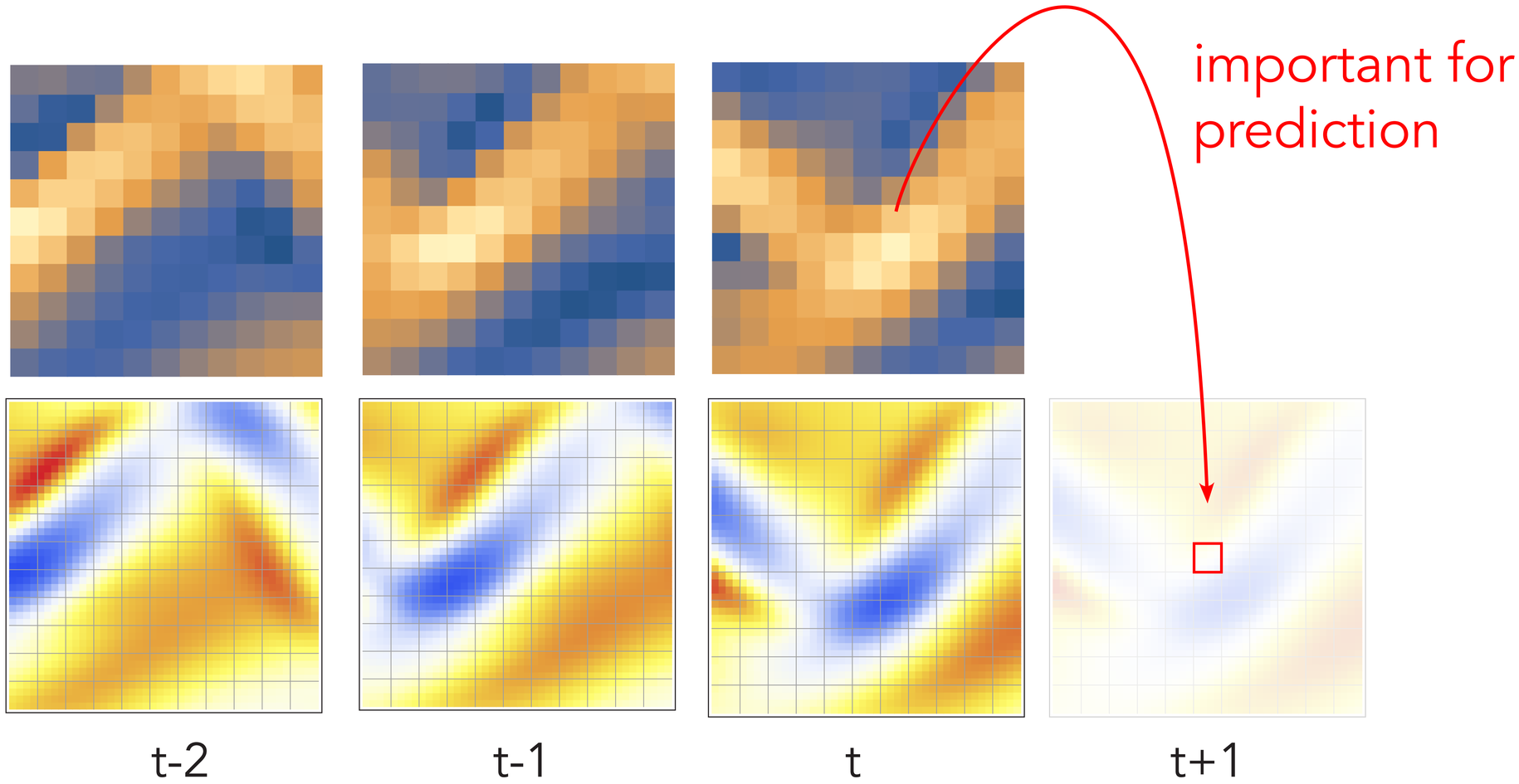
Fluid flow



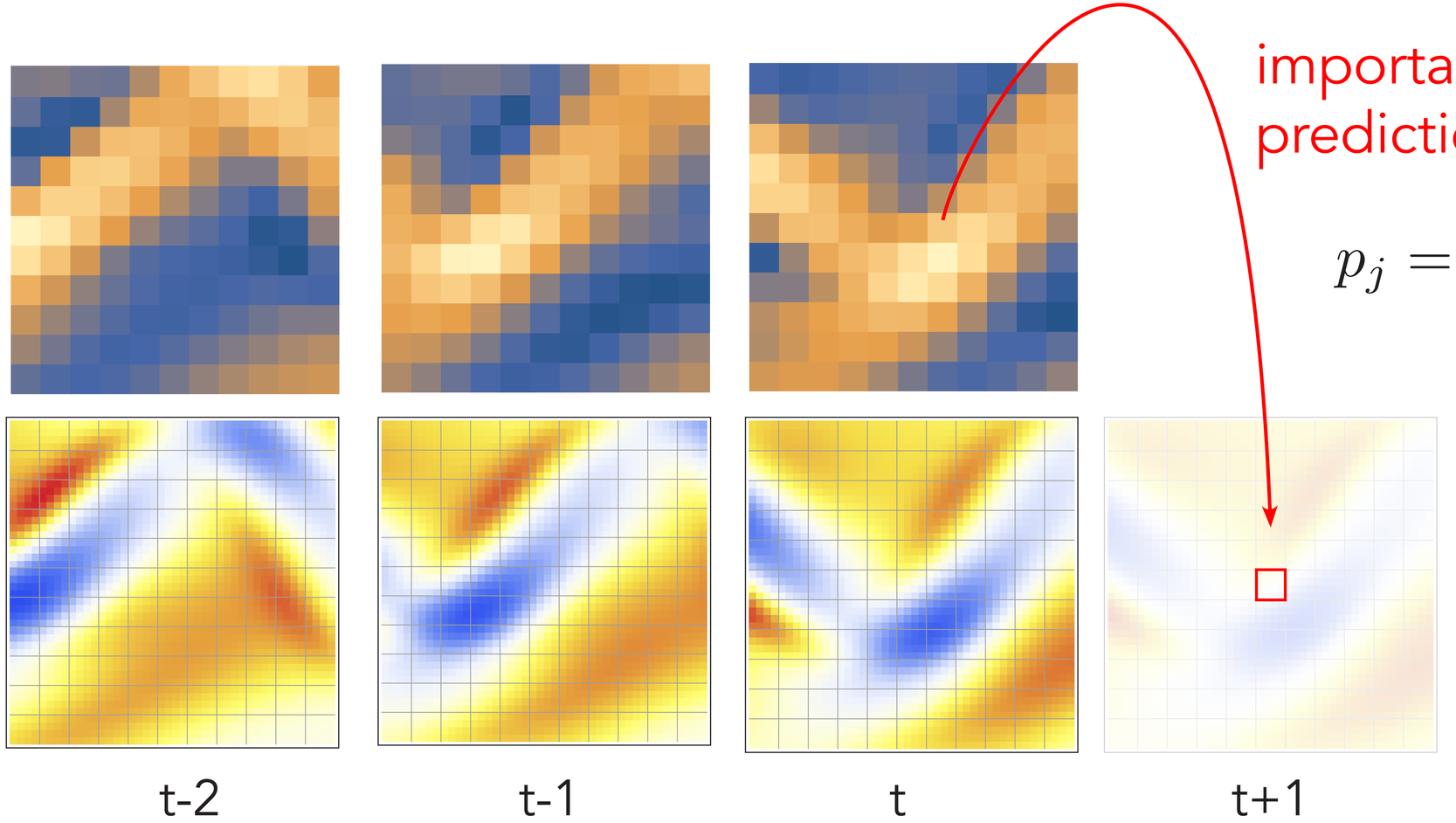
Fluid flow



Fluid flow



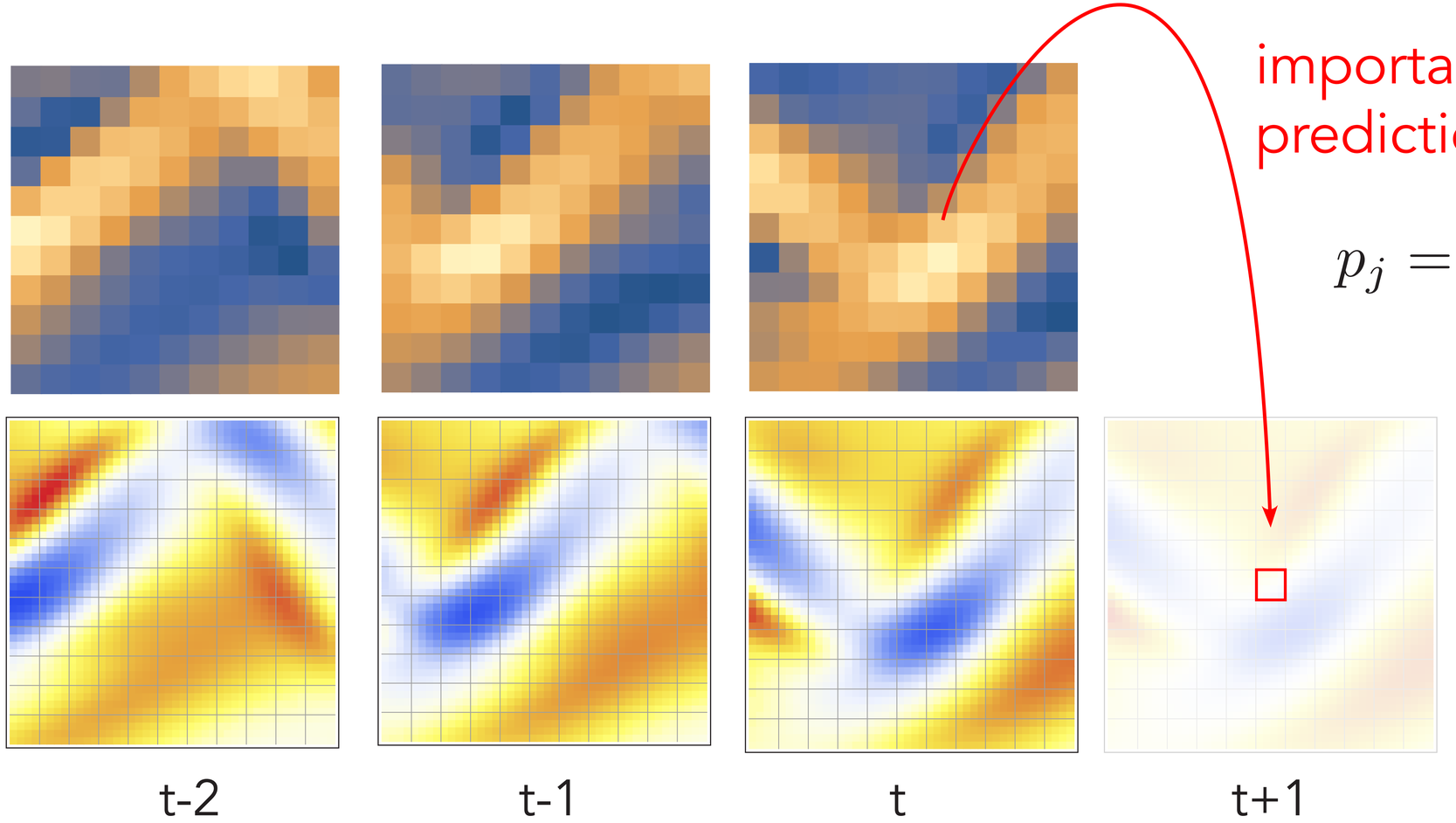
Fluid flow



important for prediction

$$p_j = \sum_j \sigma(a_{ij})\, v_j$$

Fluid flow

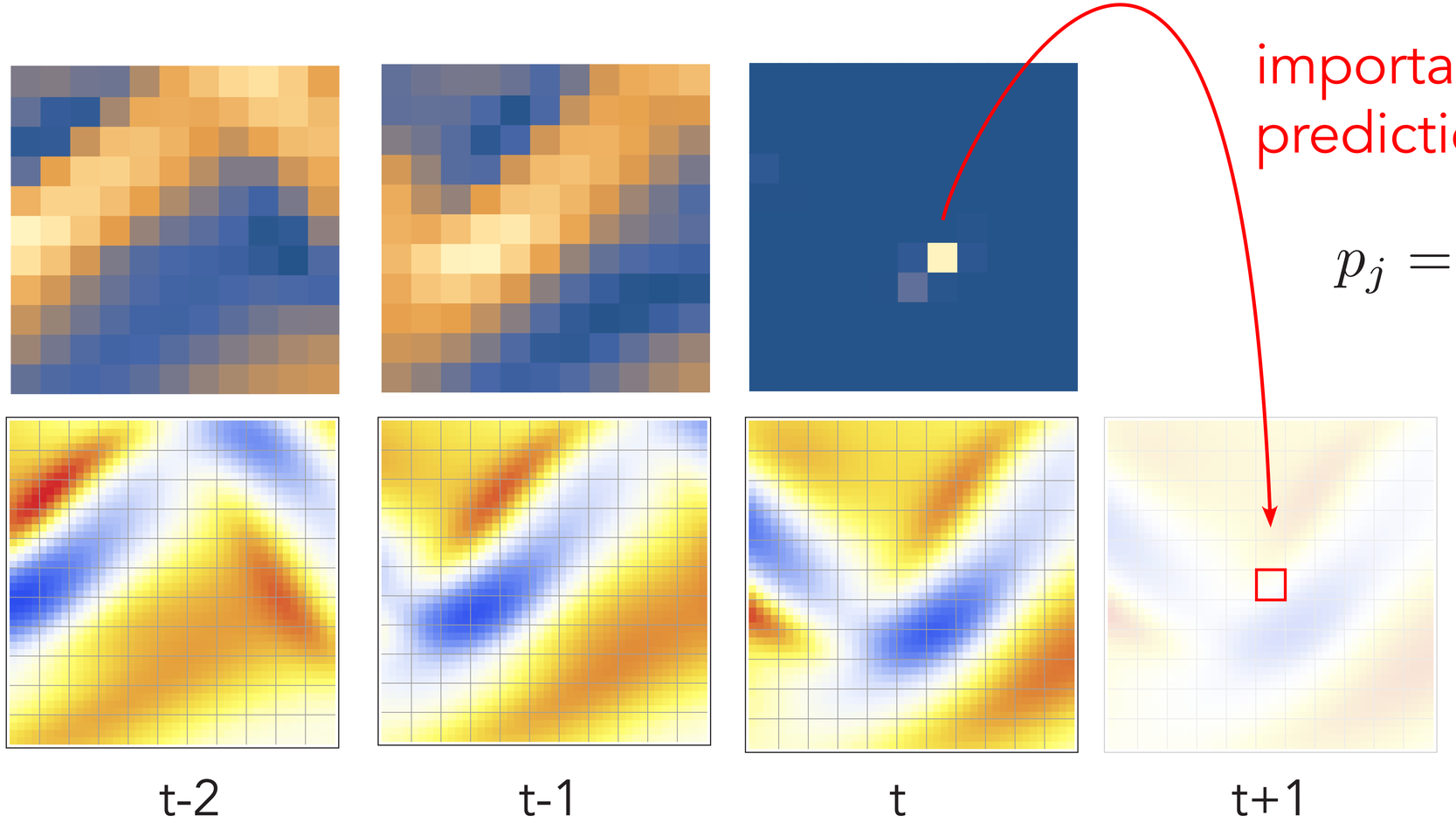


important for prediction

$$p_j = \sum_j \sigma(a_{ij})\, v_j$$

attention map values

Fluid flow

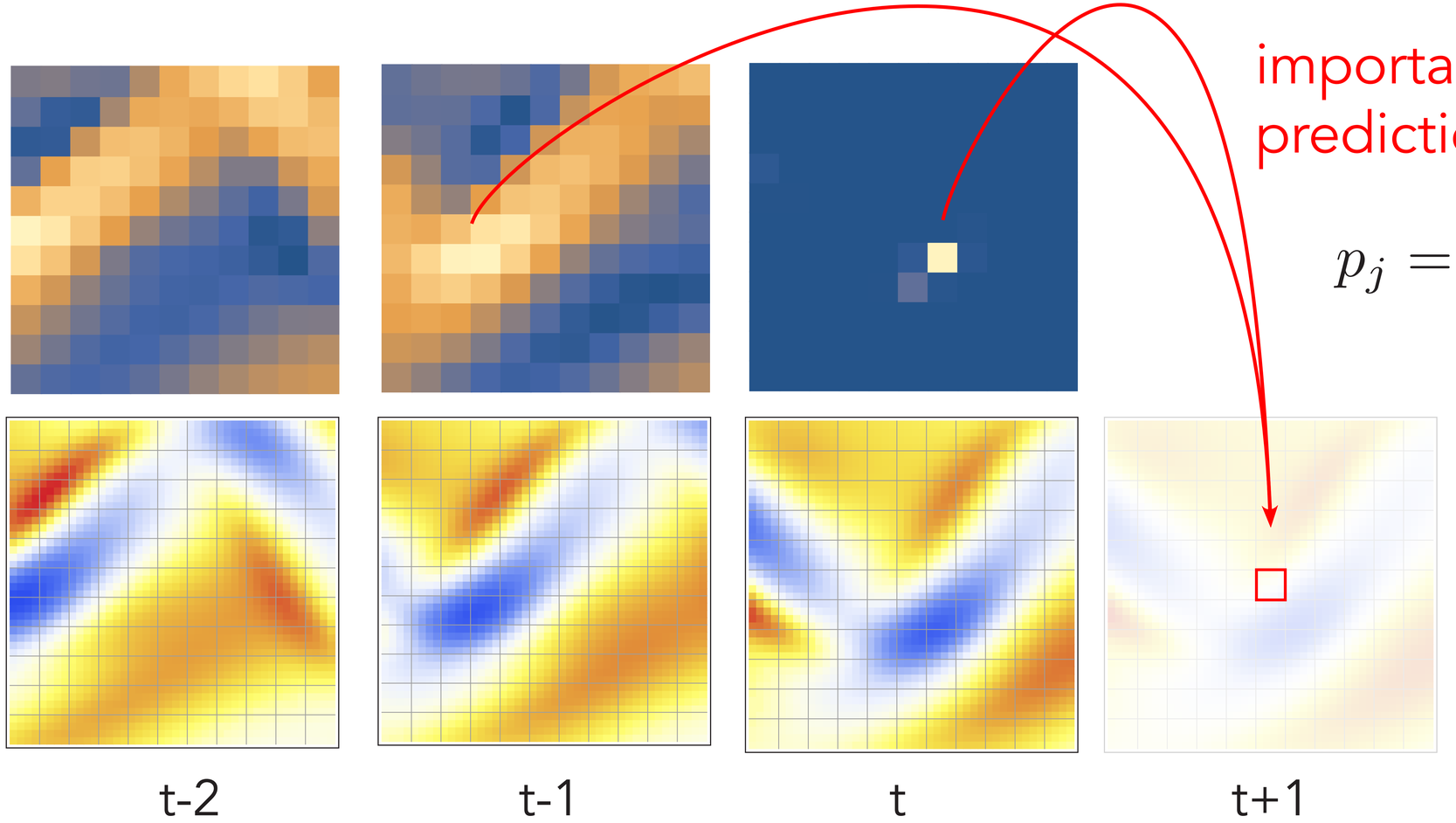


important for prediction

$$p_j = \sum_j \sigma(a_{ij})\, v_j$$

attention map values

Fluid flow

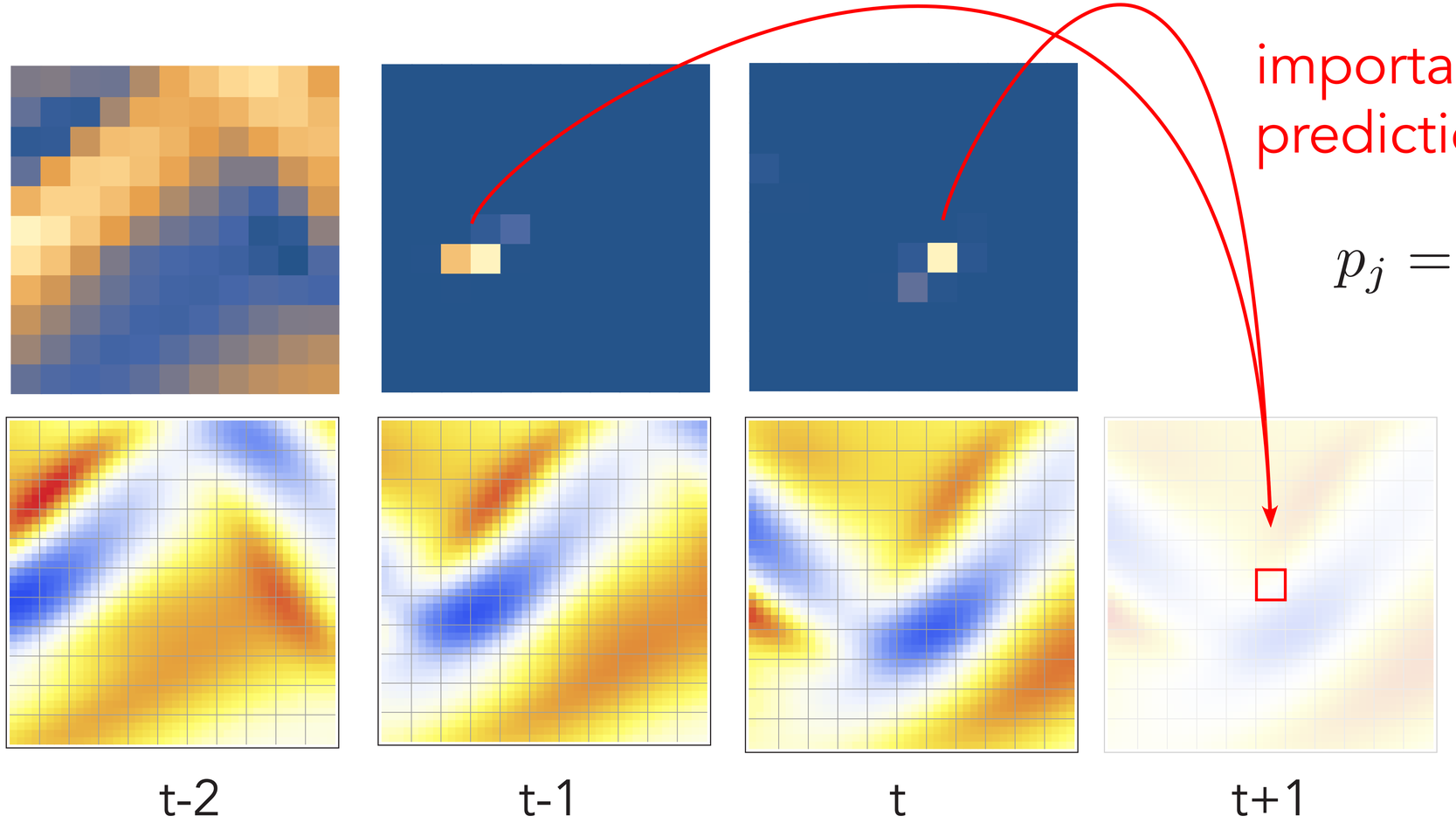


important for prediction

$$p_j = \sum_j \sigma(a_{ij})\, v_j$$

attention map values

Fluid flow

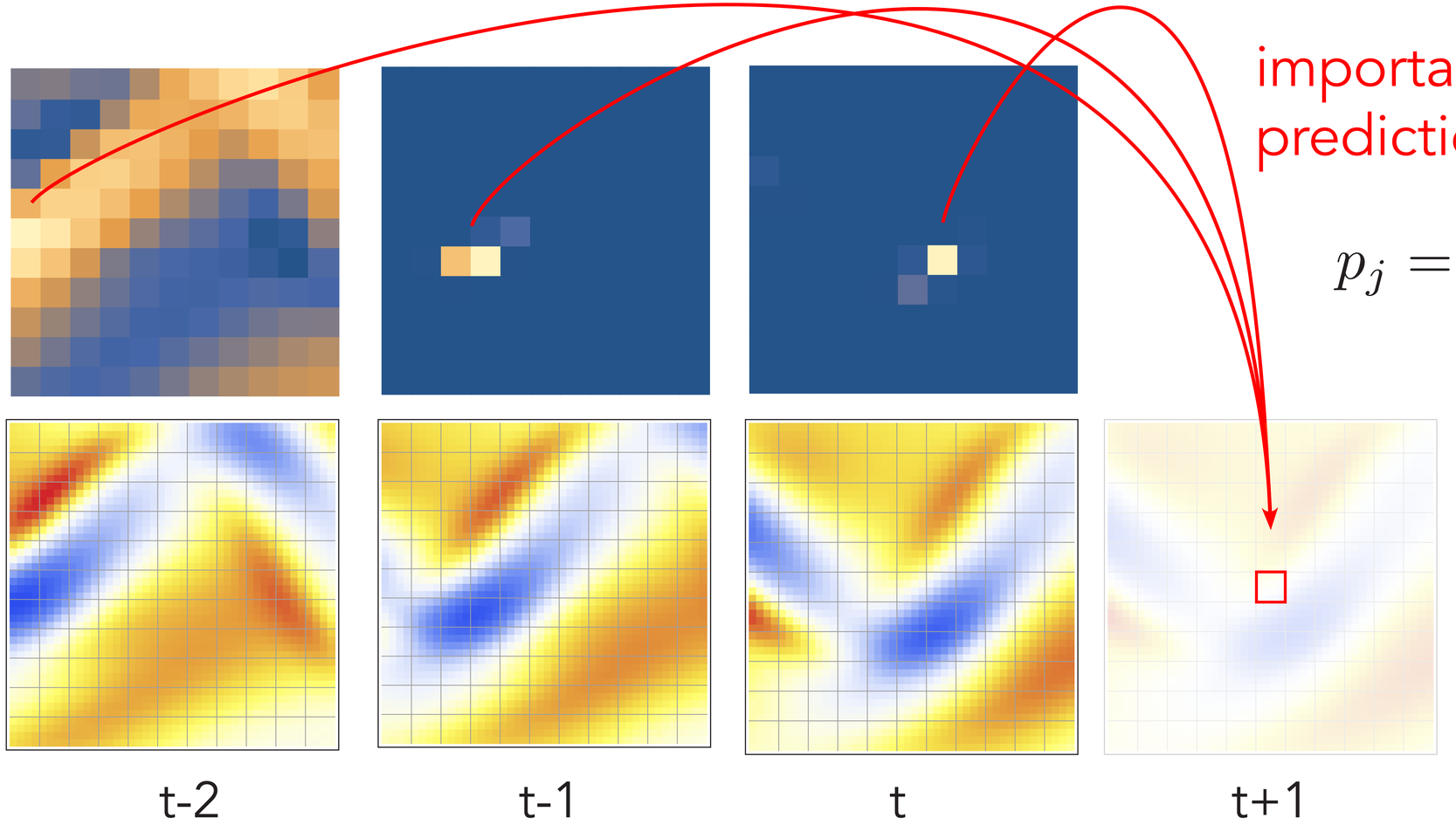


important for prediction

$$p_j = \sum_j \sigma(a_{ij})\, v_j$$

attention map values

Fluid flow

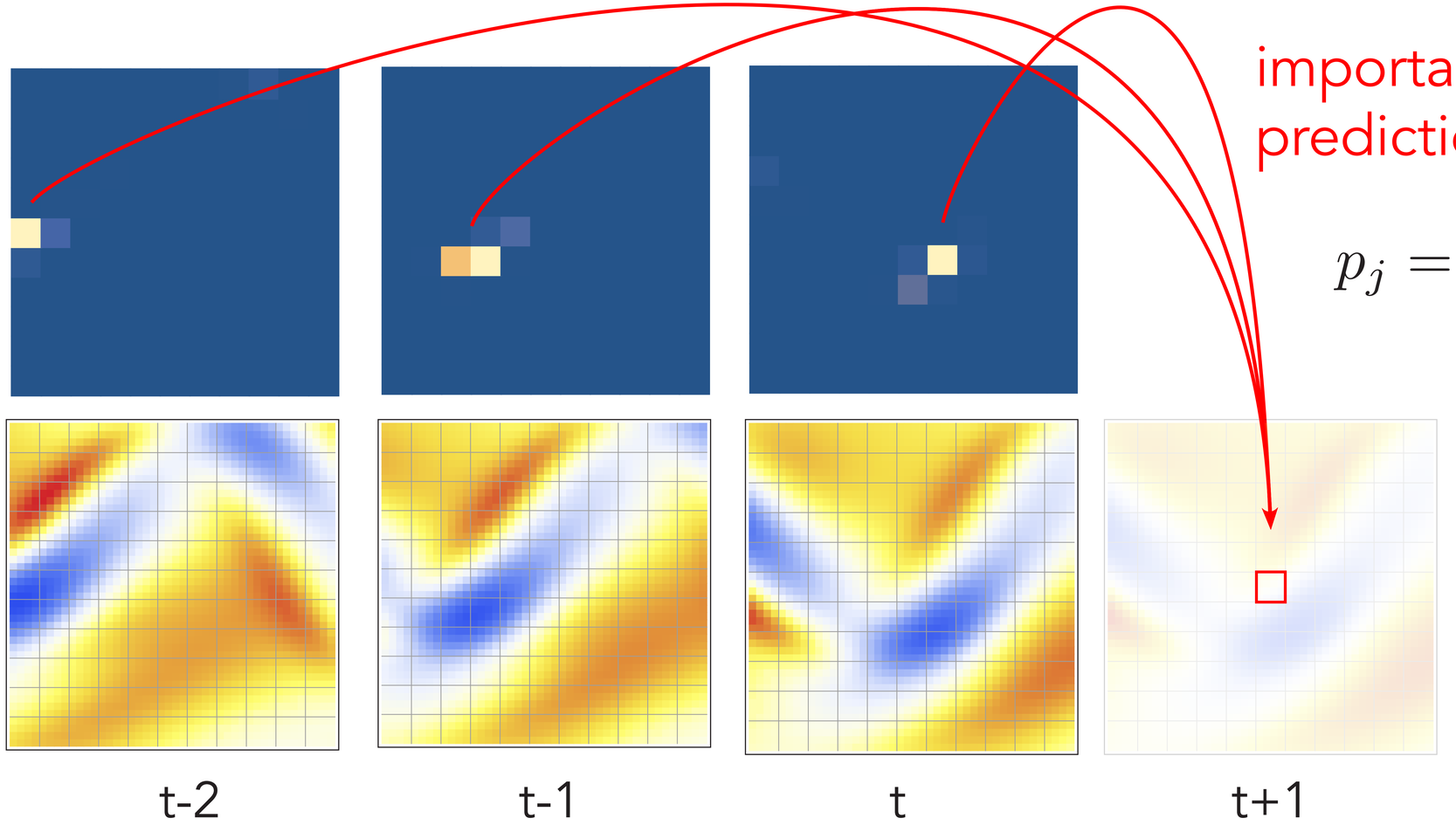


important for prediction

$$p_j = \sum_j \sigma(a_{ij})\, v_j$$

attention map values

Fluid flow

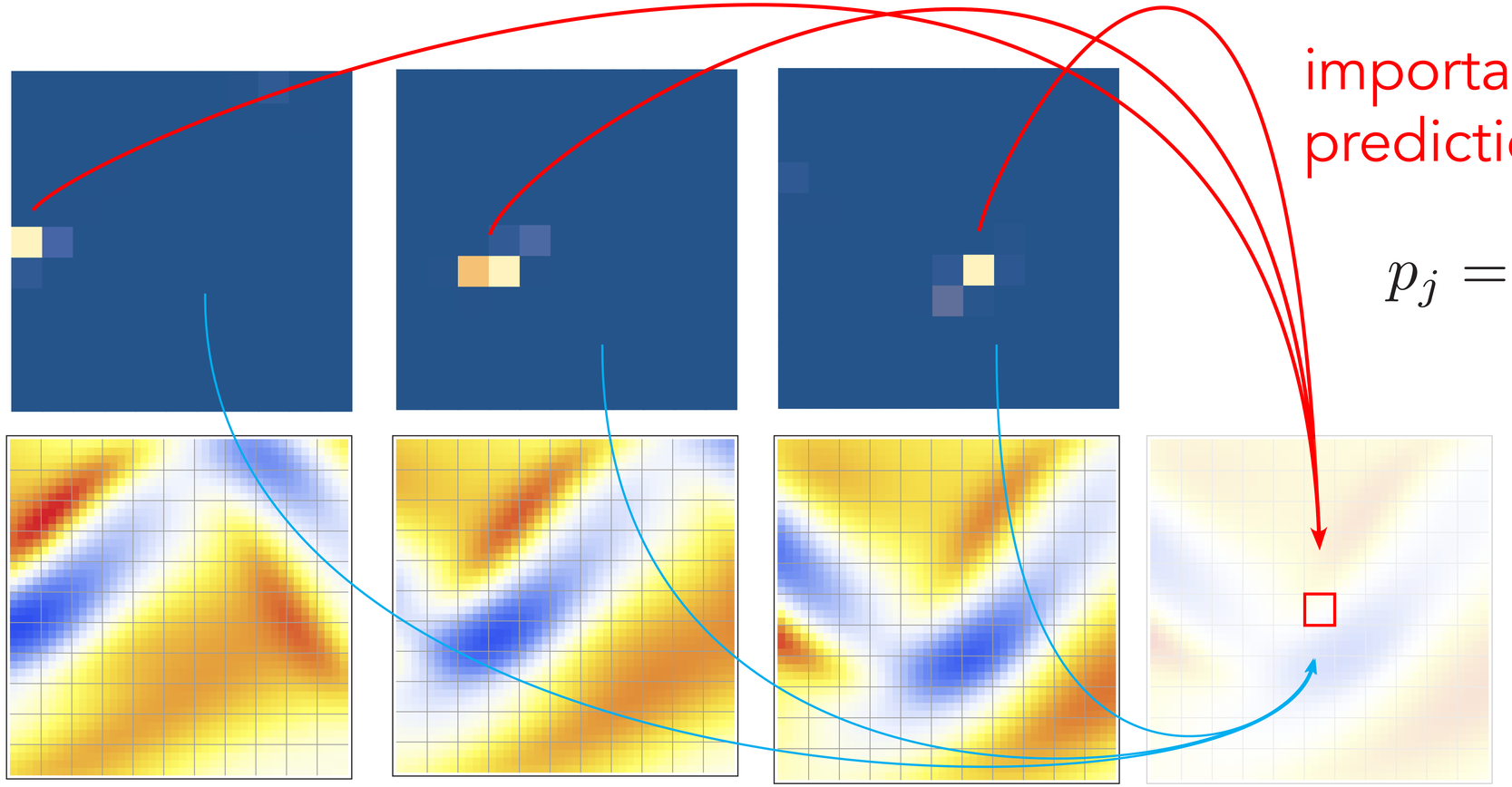


important for prediction

$$p_j = \sum_j \sigma(a_{ij})\, v_j$$

attention map values

Fluid flow



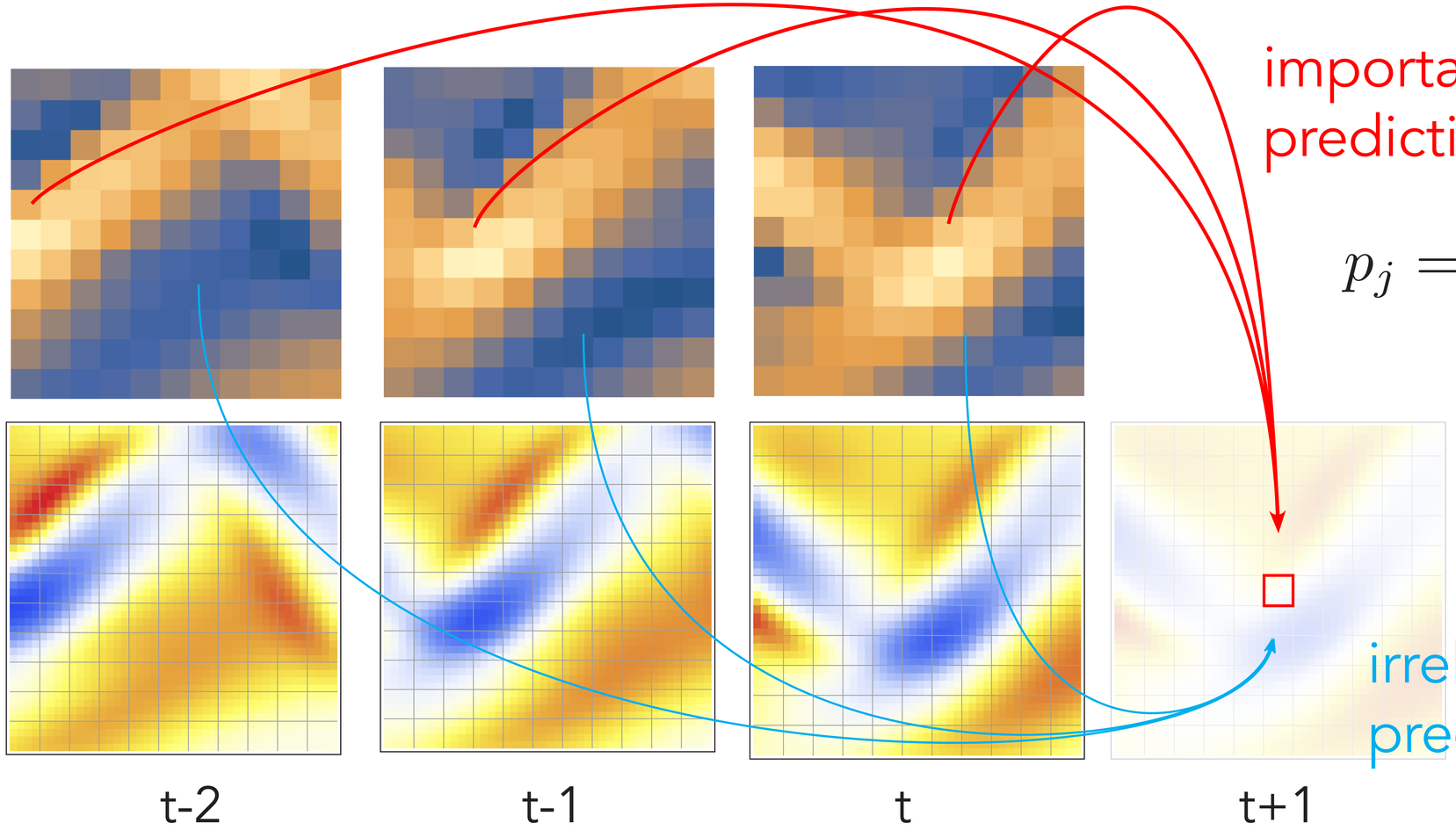
important for prediction

$$p_j = \sum_j \sigma(a_{ij})\, v_j$$

attention map values

irrelevant for prediction

Fluid flow



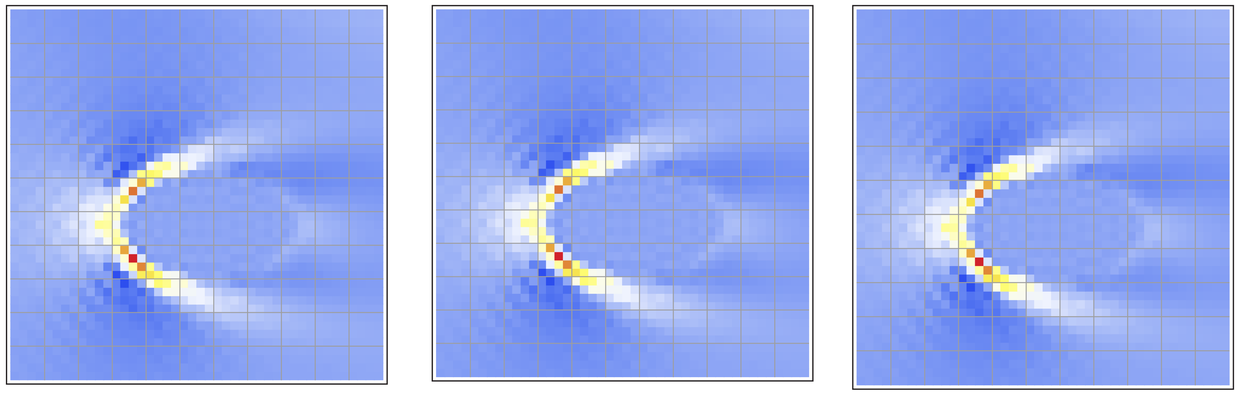
important for prediction

$$p_j = \sum_j \sigma(a_{ij})\, v_j$$

attention map values

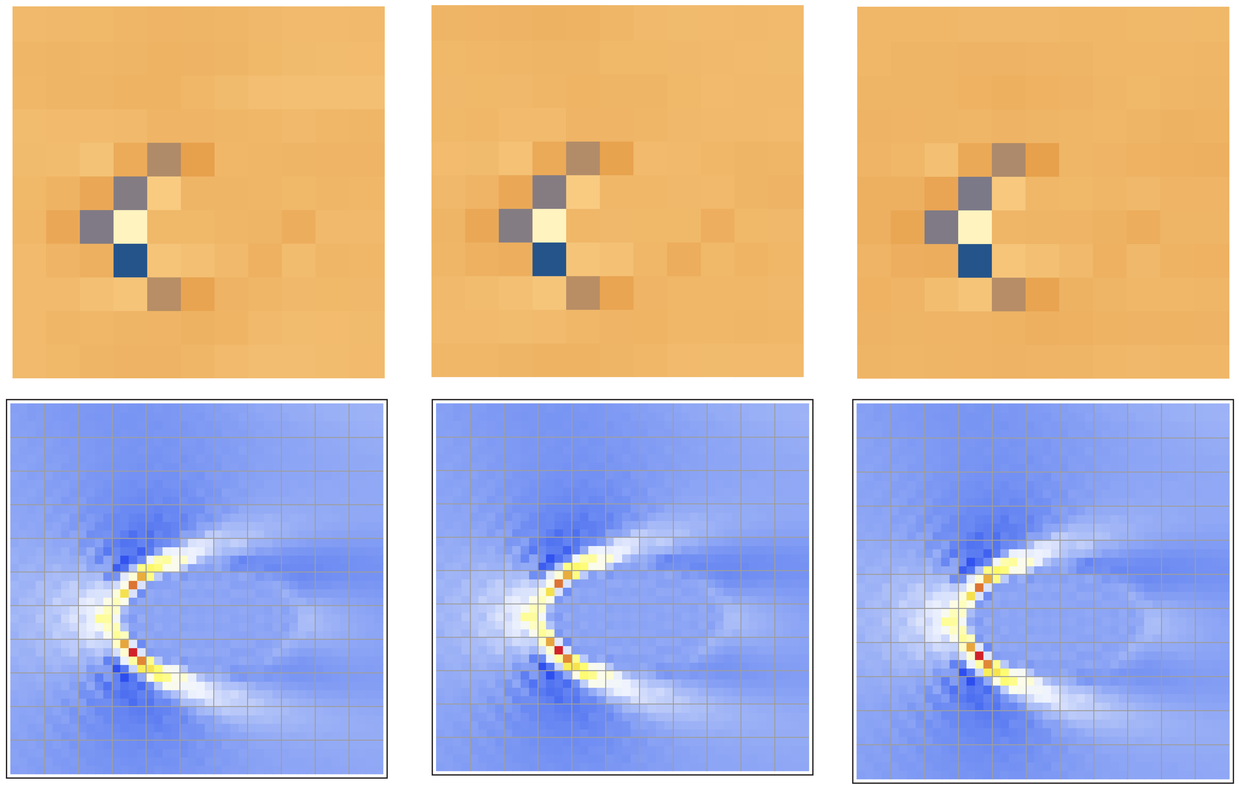
irrelevant for prediction

Fluid flow

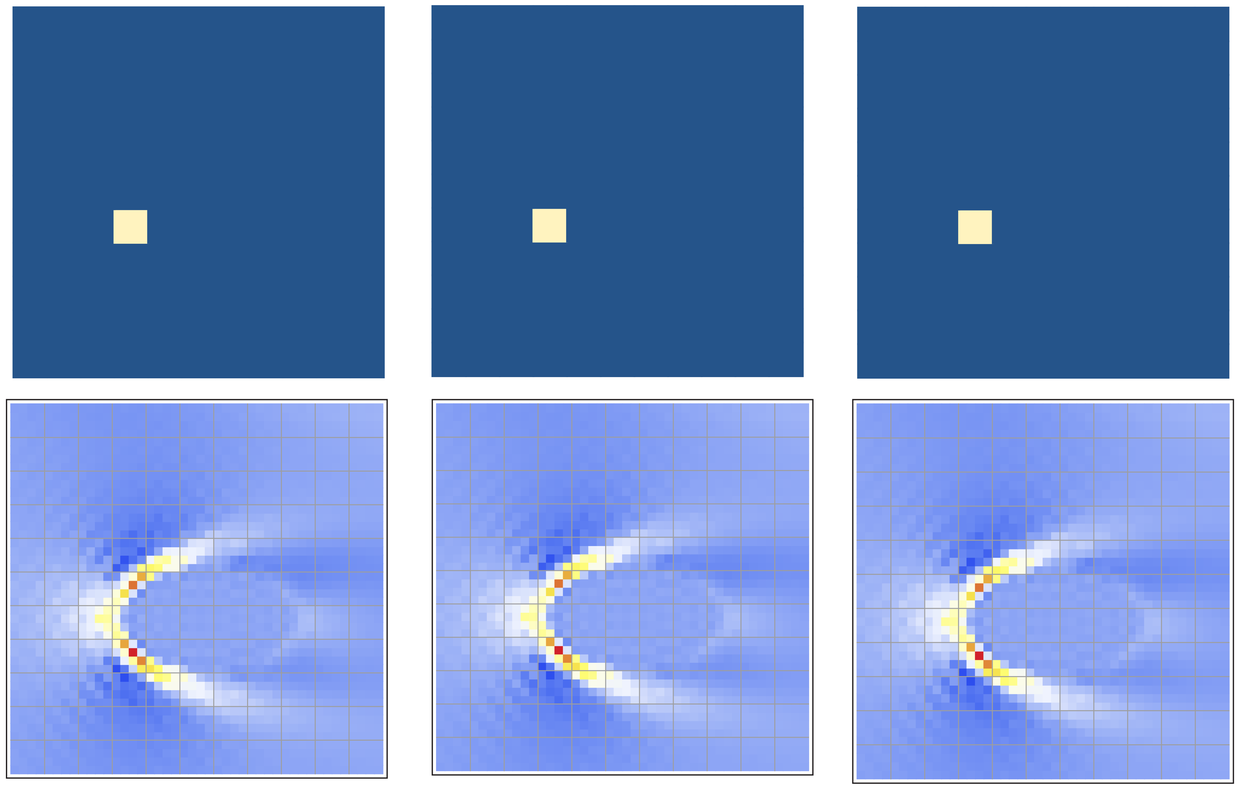


t-2 t-1 t

Fluid flow

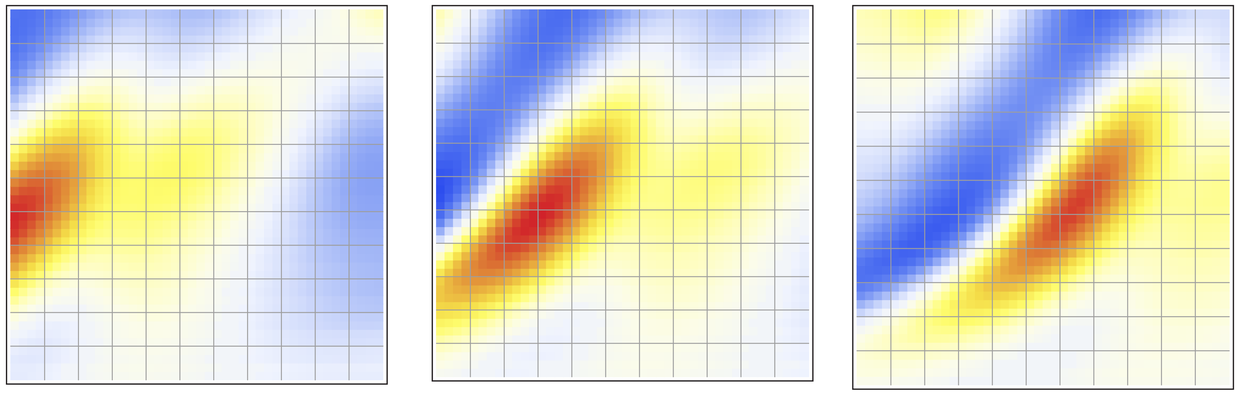


Fluid flow



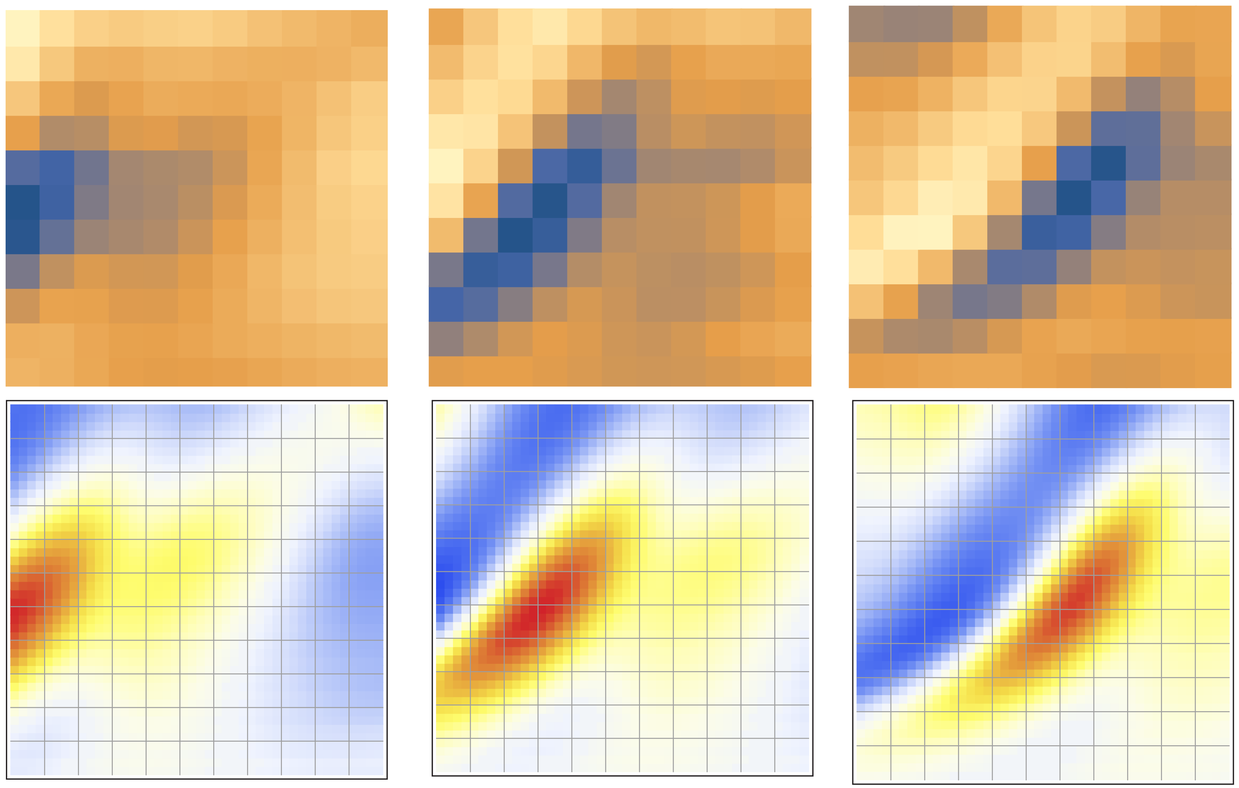
t-2 t-1 t

Fluid flow



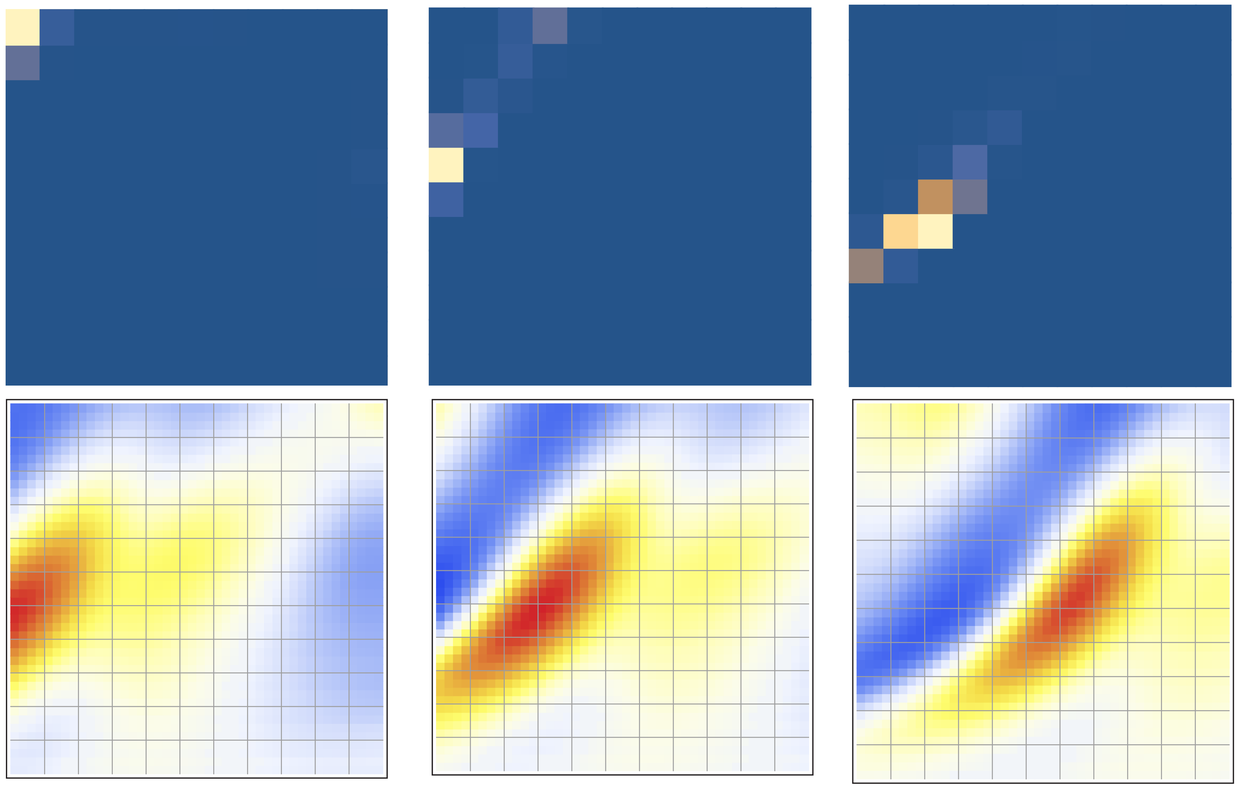
t-2 t-1 t

Fluid flow



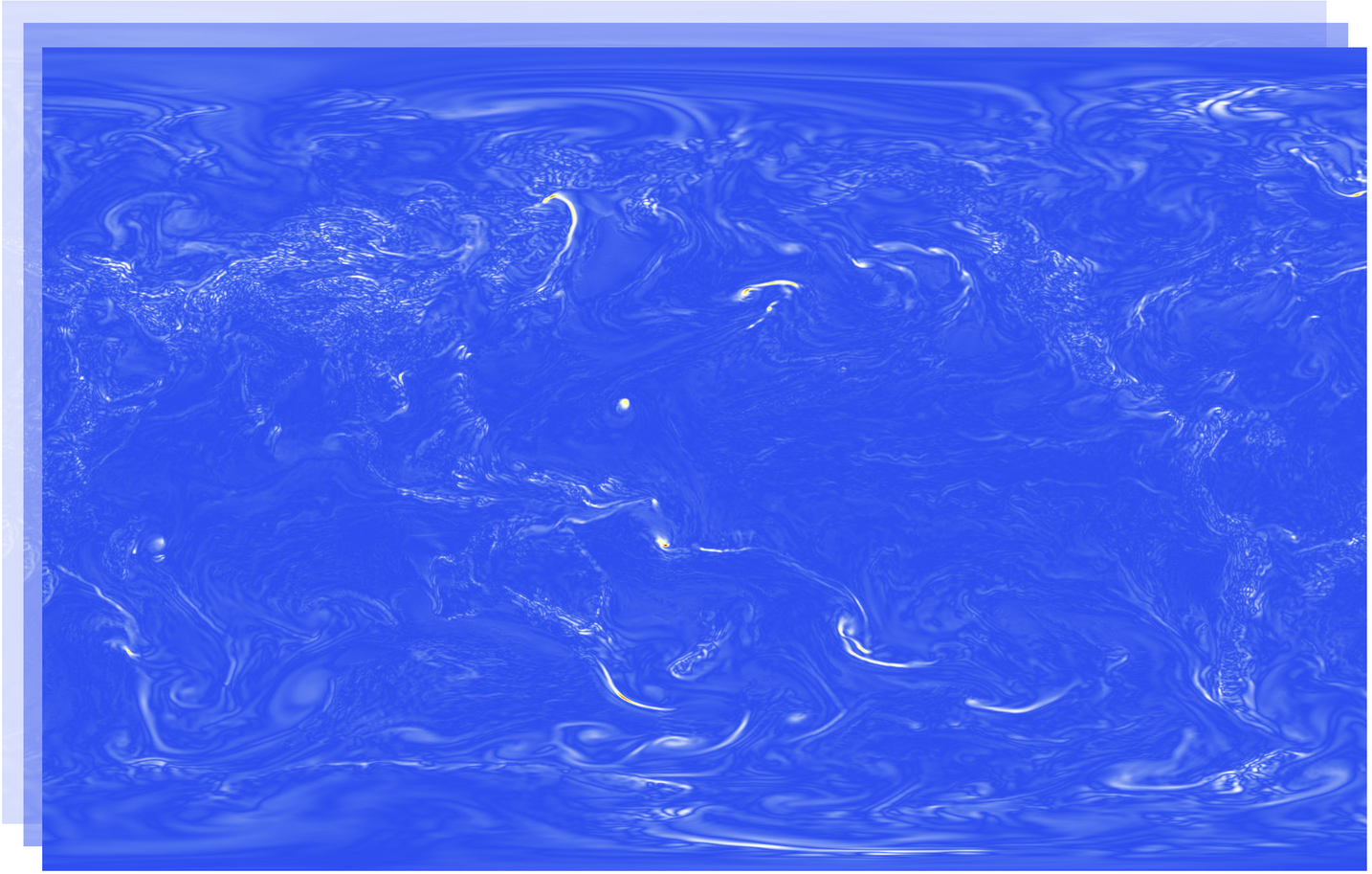
t-2 t-1 t

Fluid flow

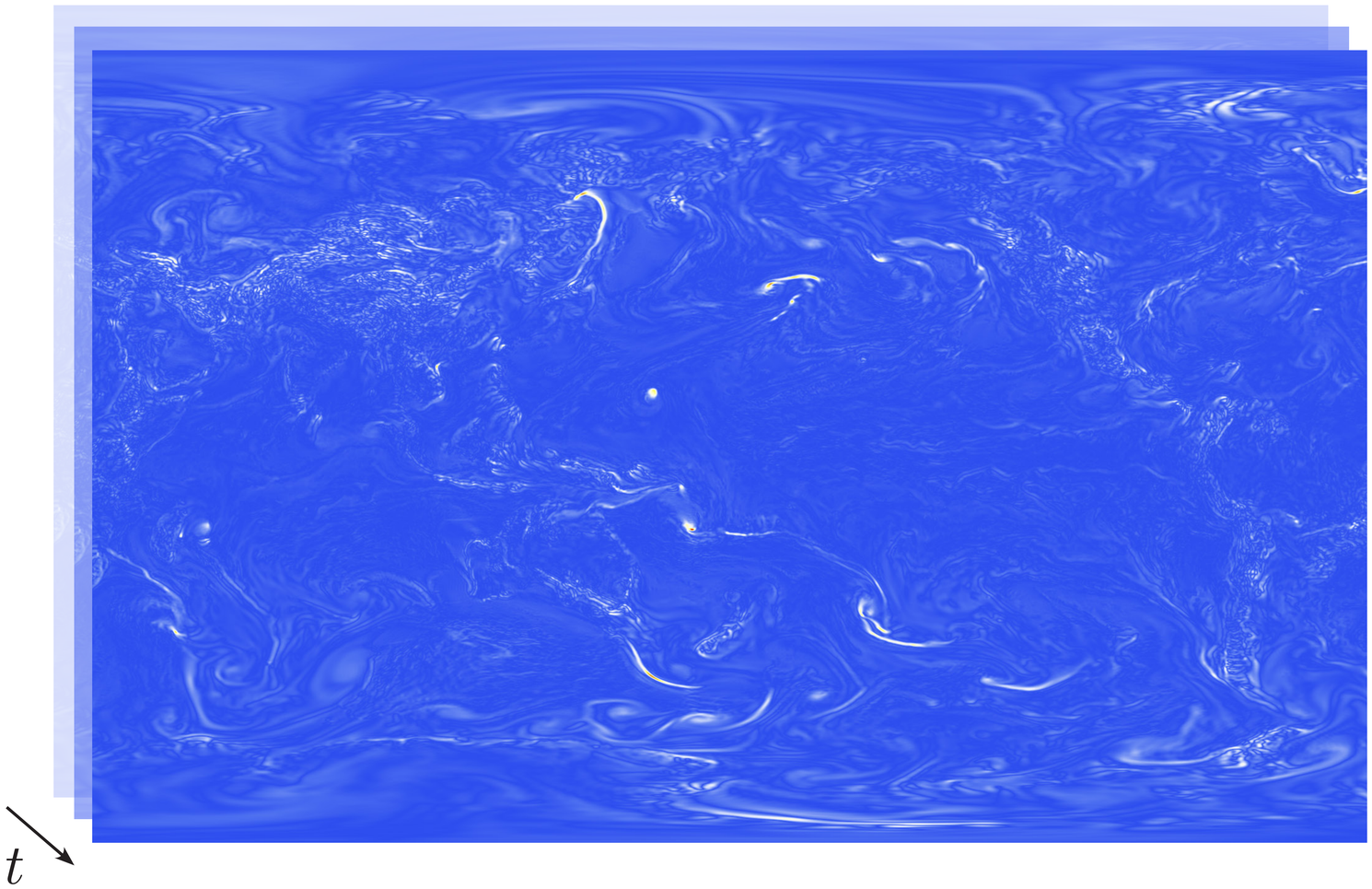


t-2 t-1 t

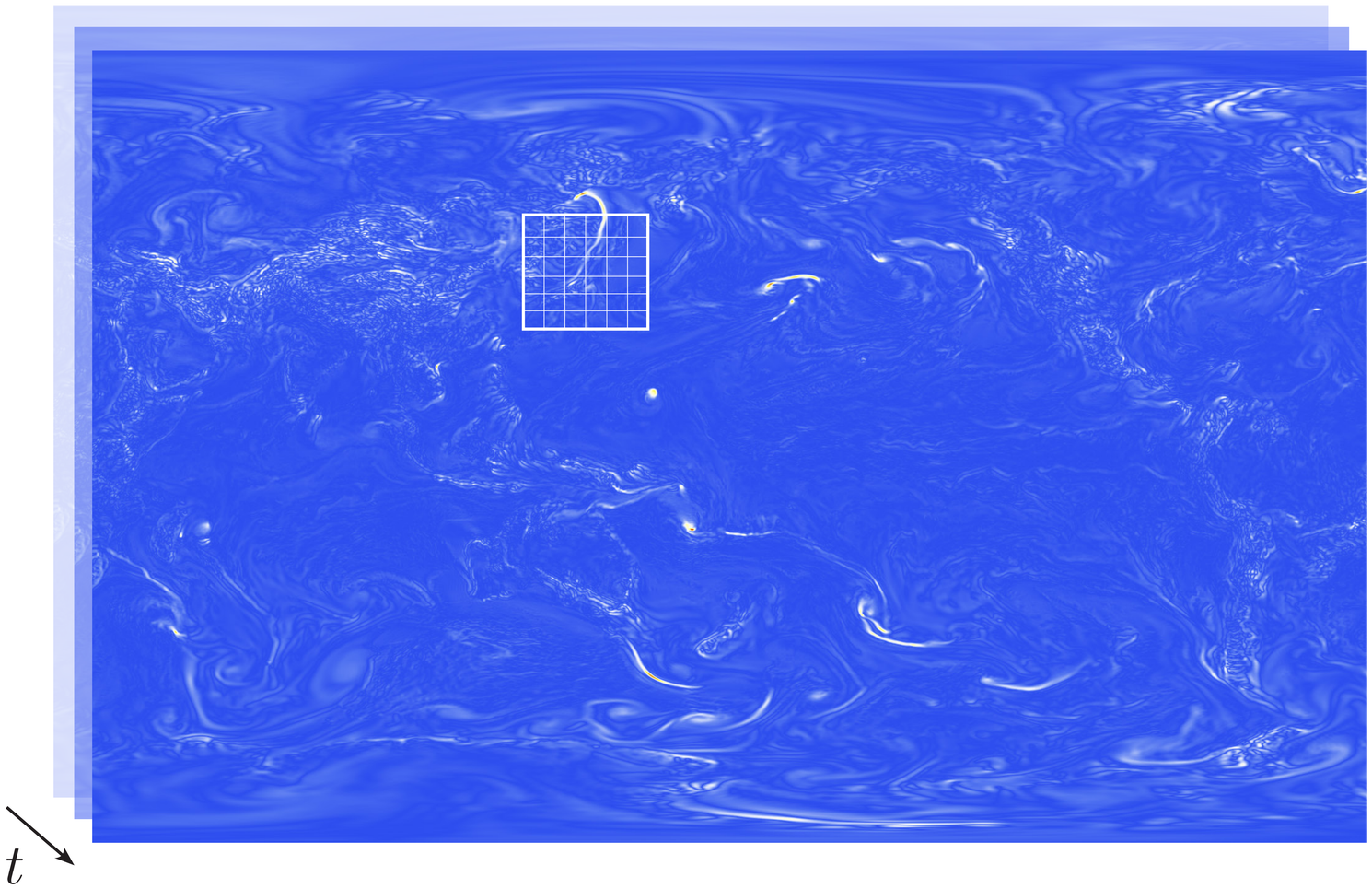
Atmospheric vorticity (ERA5)



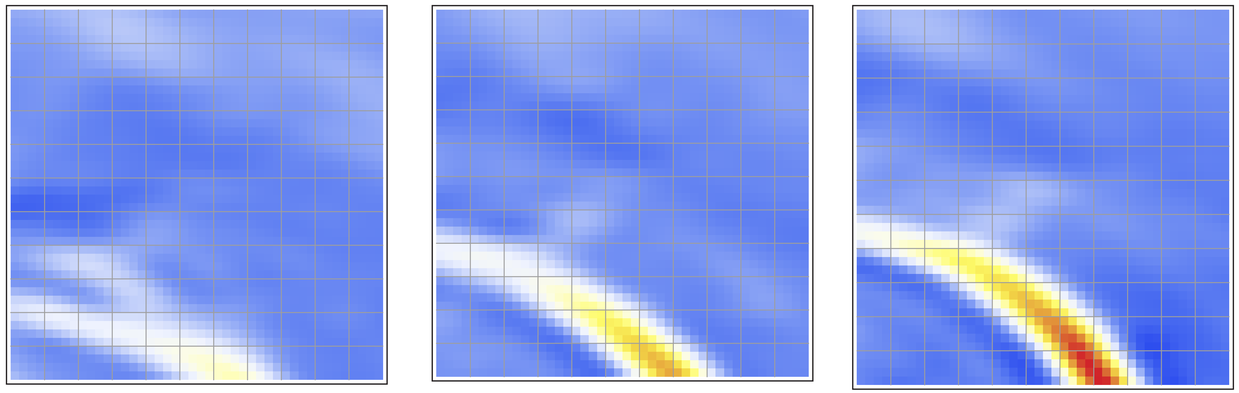
Atmospheric vorticity (ERA5)



Atmospheric vorticity (ERA5)

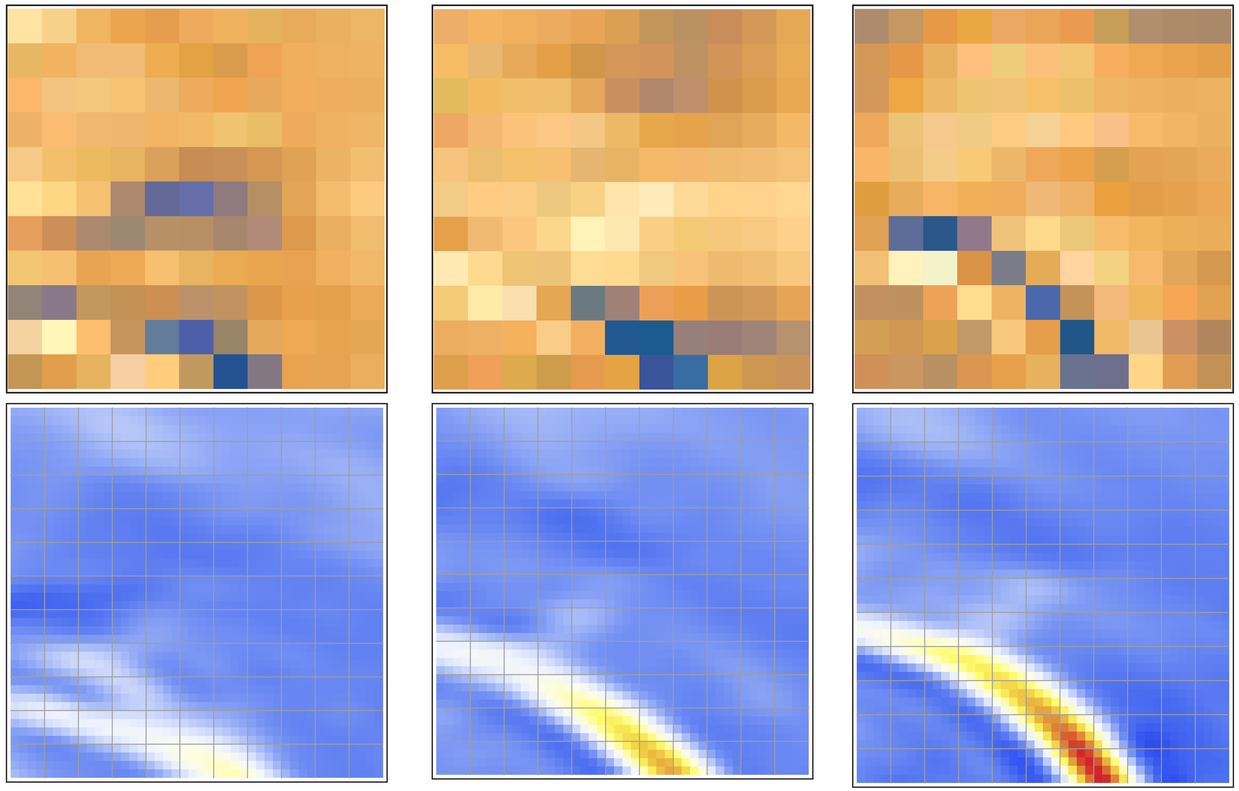


Atmospheric vorticity (ERA5)



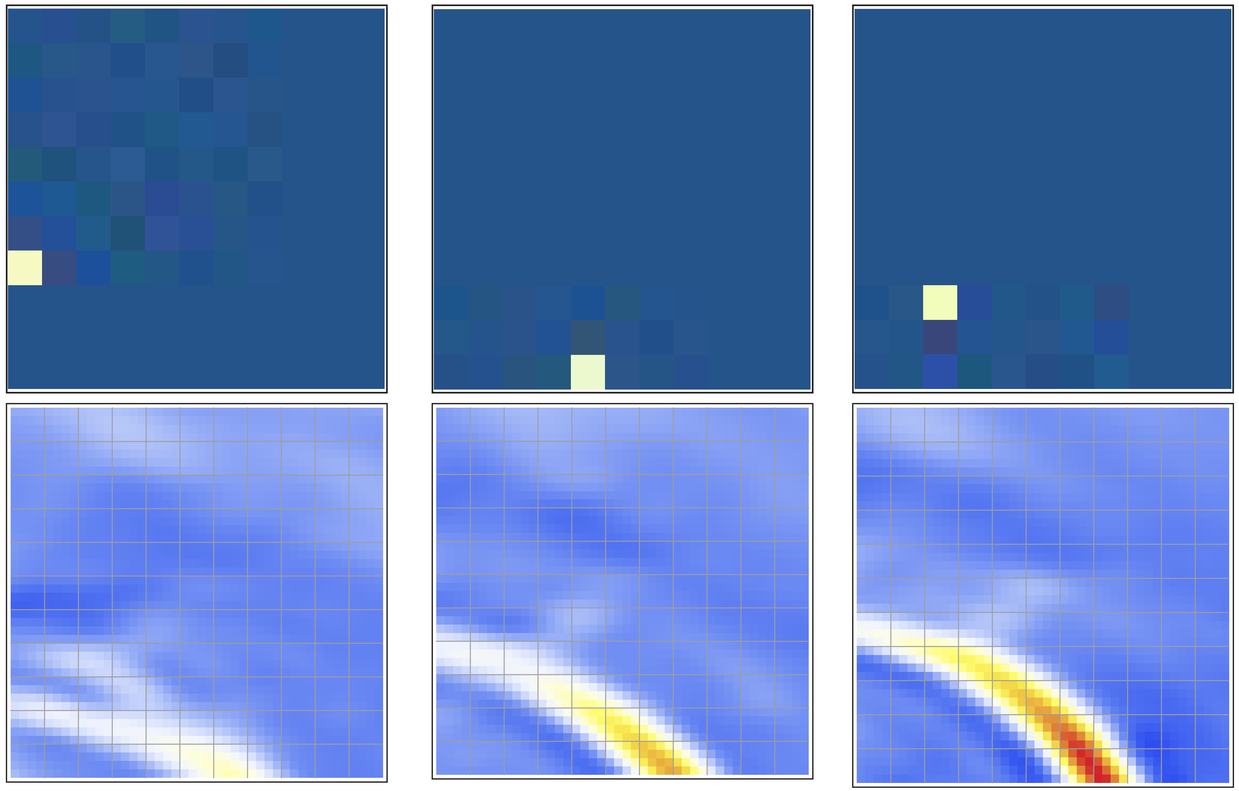
t-2 t-1 t

Atmospheric vorticity (ERA5)



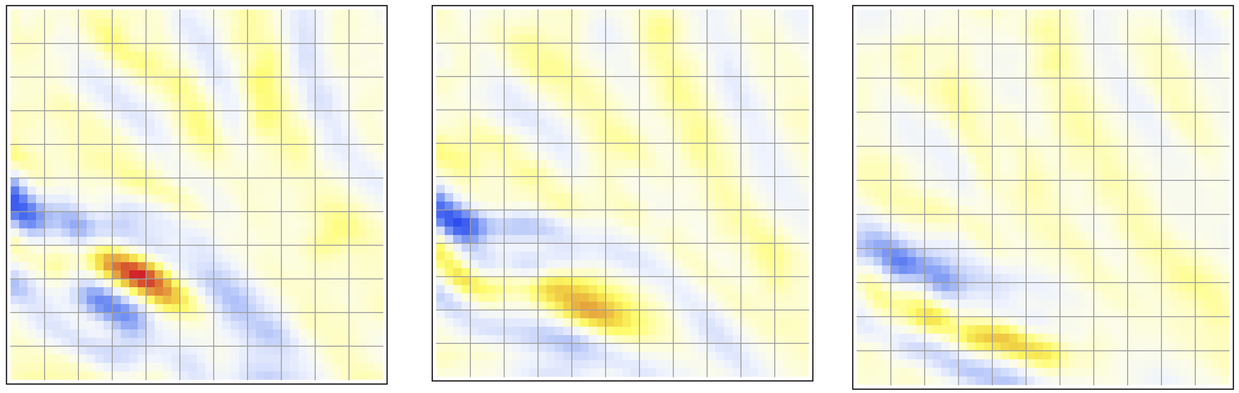
t-2 t-1 t

Atmospheric vorticity (ERA5)



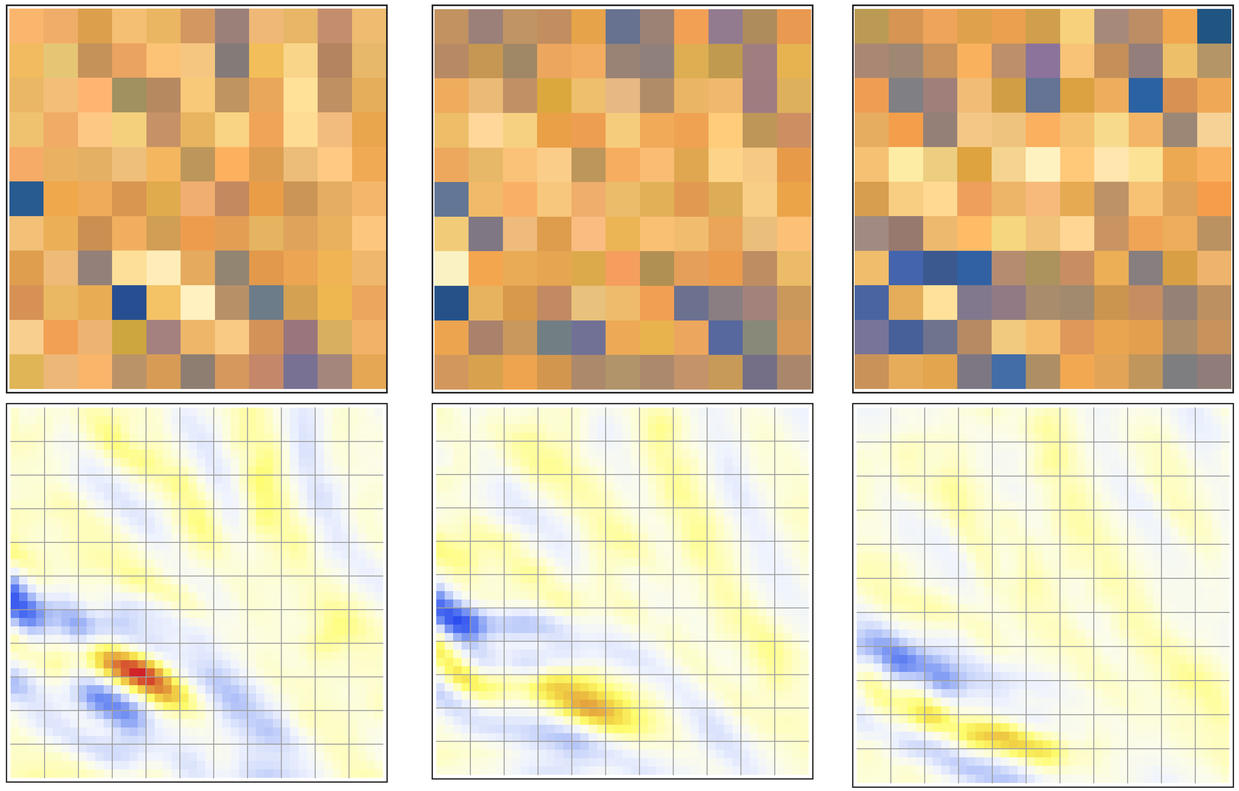
t-2 t-1 t

Atmospheric vorticity (ERA5)



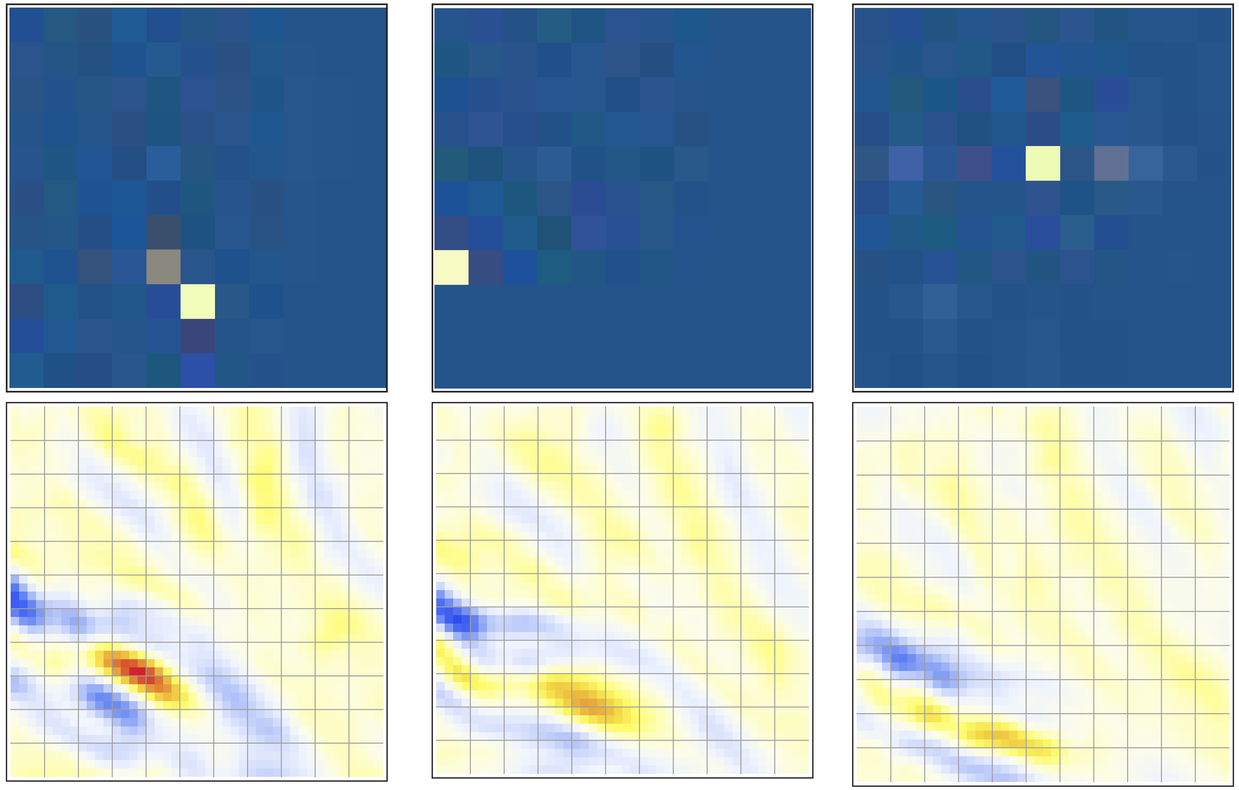
t-2 t-1 t

Atmospheric vorticity (ERA5)



t-2 t-1 t

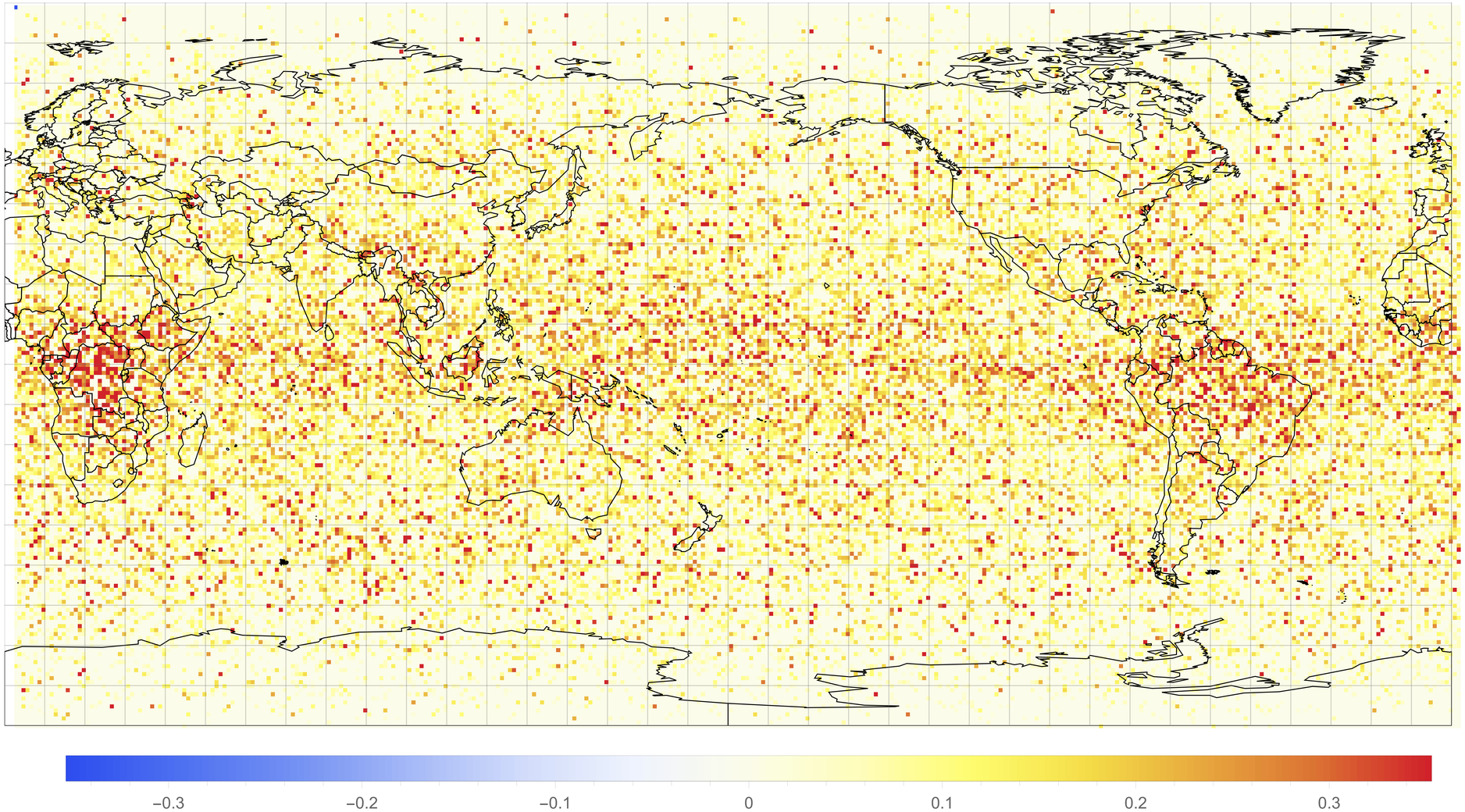
Atmospheric vorticity (ERA5)



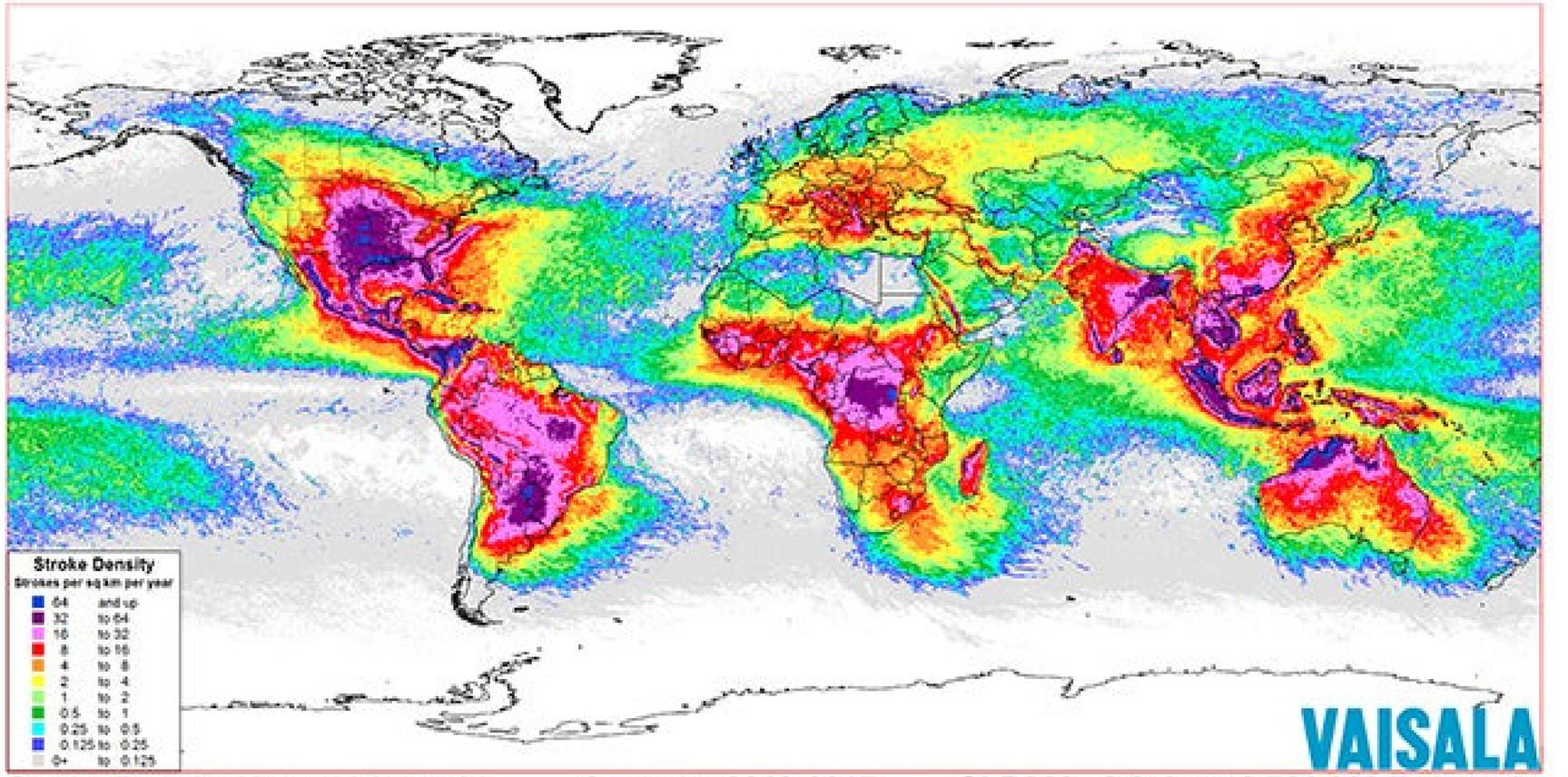
t-2 t-1 t

3 August 2022

Prediction (relative error, 1979)

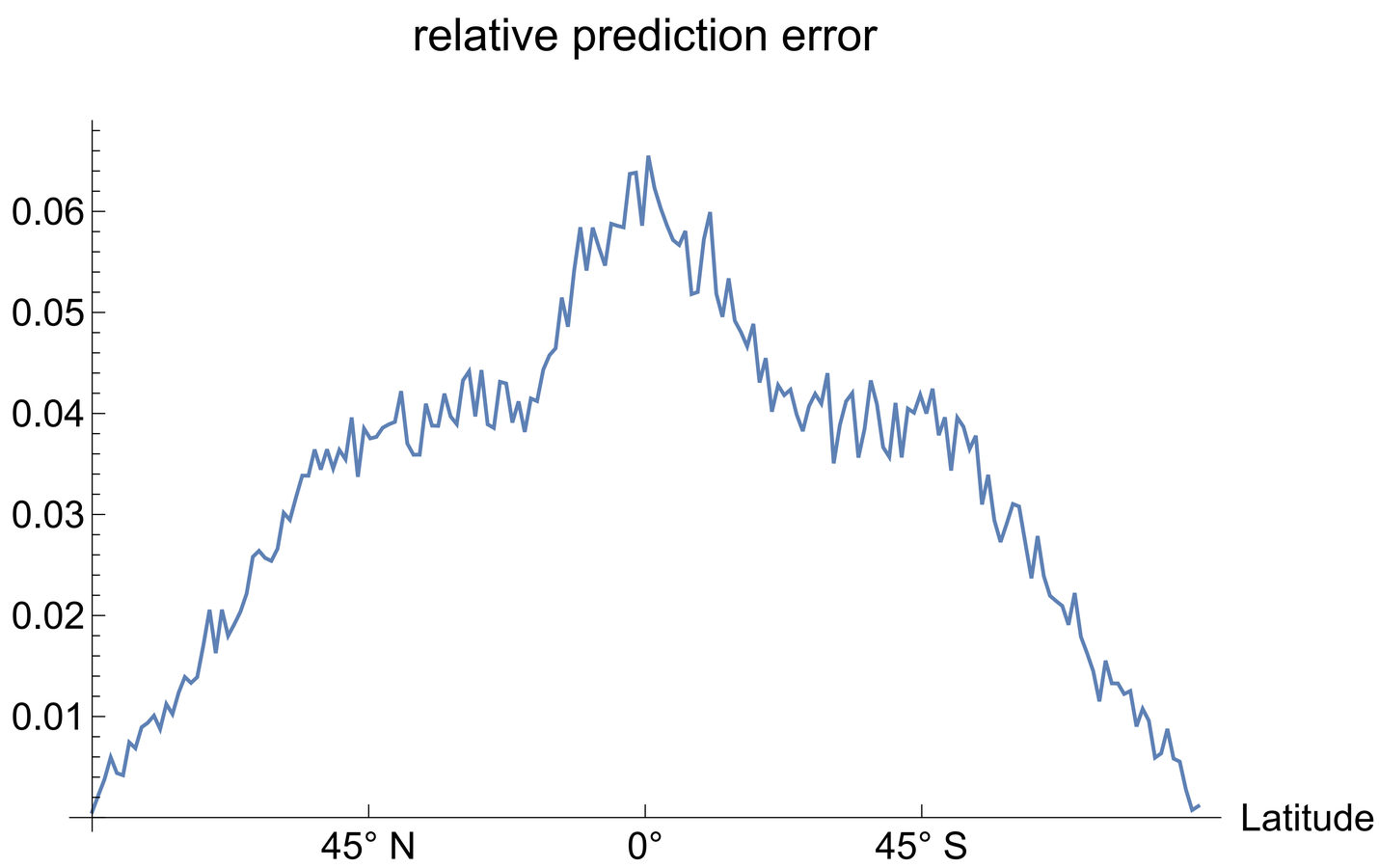


Prediction

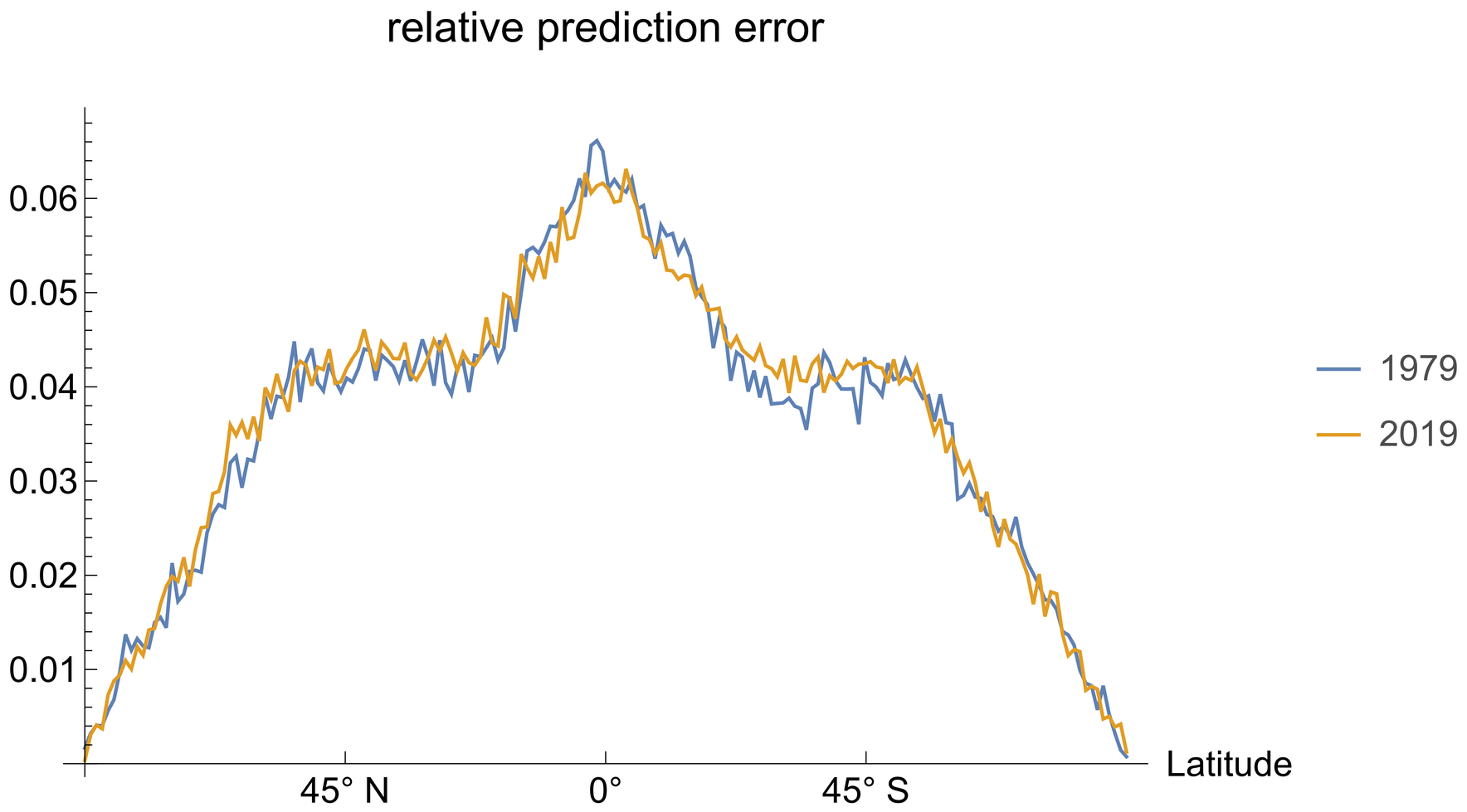


Stroke Density Map - 20 km grid Average: 2013-2017 GLD360 v2.0 data: 8,761,390,744 strokes

Prediction



Prediction



Code

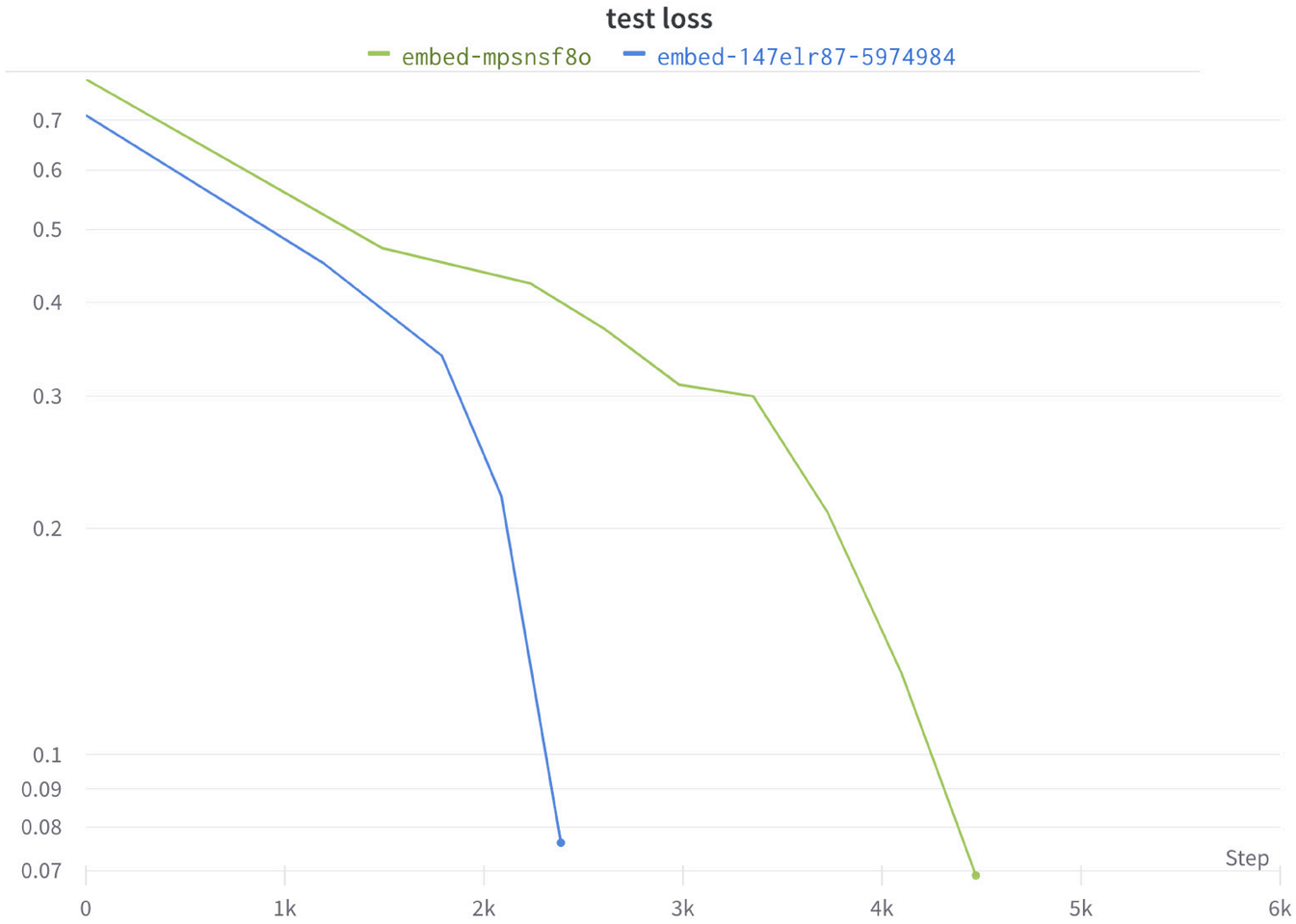
- Code is stabilizing
- Infrastructure code is implemented and working
- Parallelization shows very good (weak) scaling
- Multiformer code
  - Other users: Yi Deng, Bing
  - Sharing currently ad-hoc and by hand
  - There seems to be a clear demand

Data pipeline

- Local approach: work on local spatio-temporal neighborhoods
- Size of neighborhoods is (overall) pre-determined, center in space and time is sampled randomly with area preserving sampling (correction for spherical distortion)
- Embarrassingly parallel: different parallel tasks sample independently
- Data is stored in monthly files: allows good randomization while still having sufficiently large files

Training: 1 task (local) versus 8 tasks (booster)

blur=5

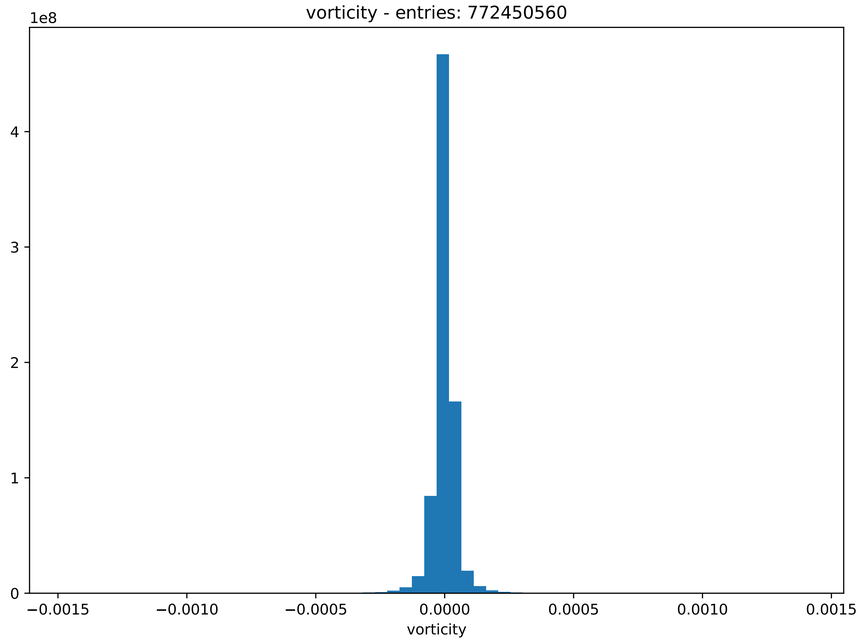


Data normalization

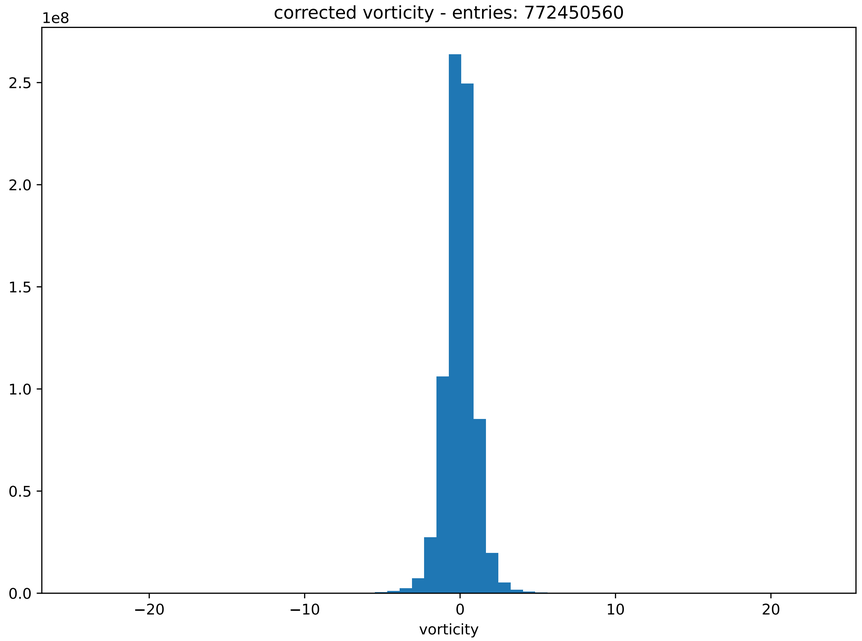
- All fields are scaled to zero mean and unit variance
  - Per grid point
  - Per month
- Computations completed (1979-2019) and on booster.

Data normalization: vorticity

uncorrected

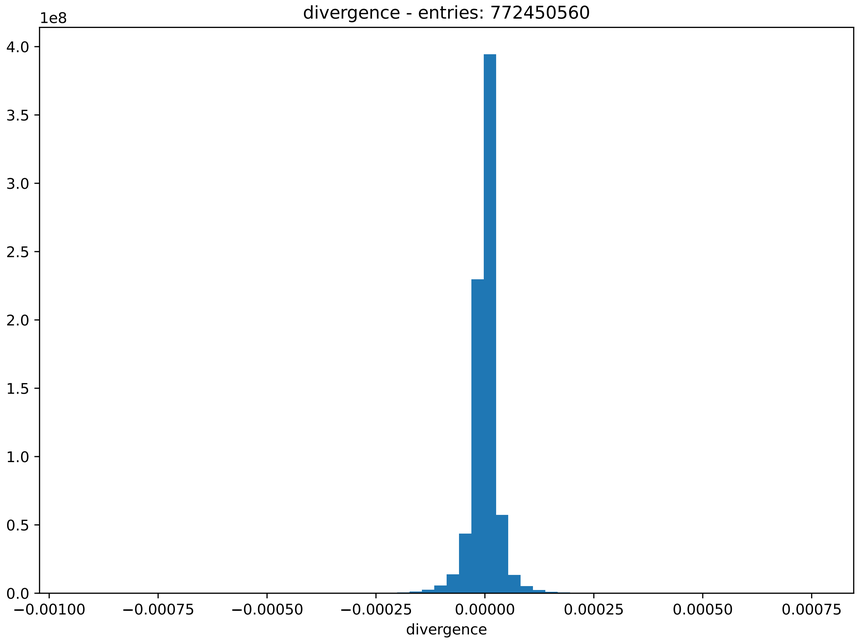


corrected

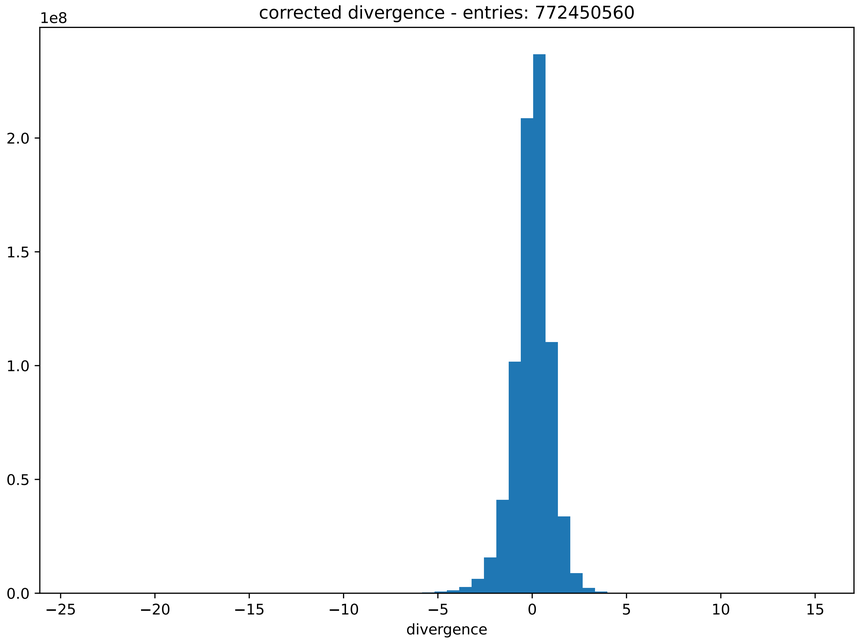


Data normalization: divergence

uncorrected

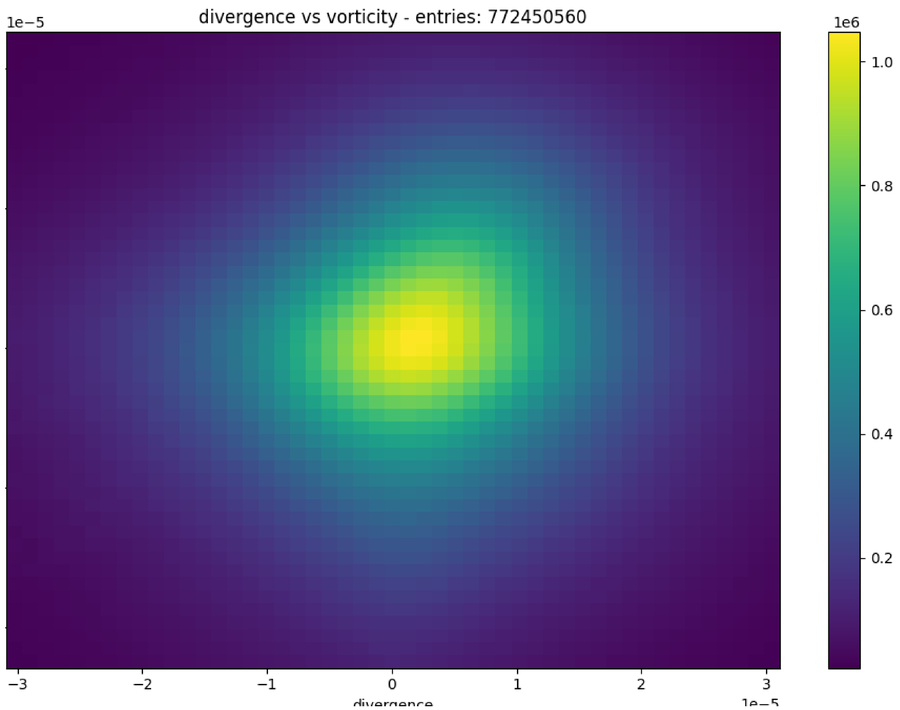


corrected

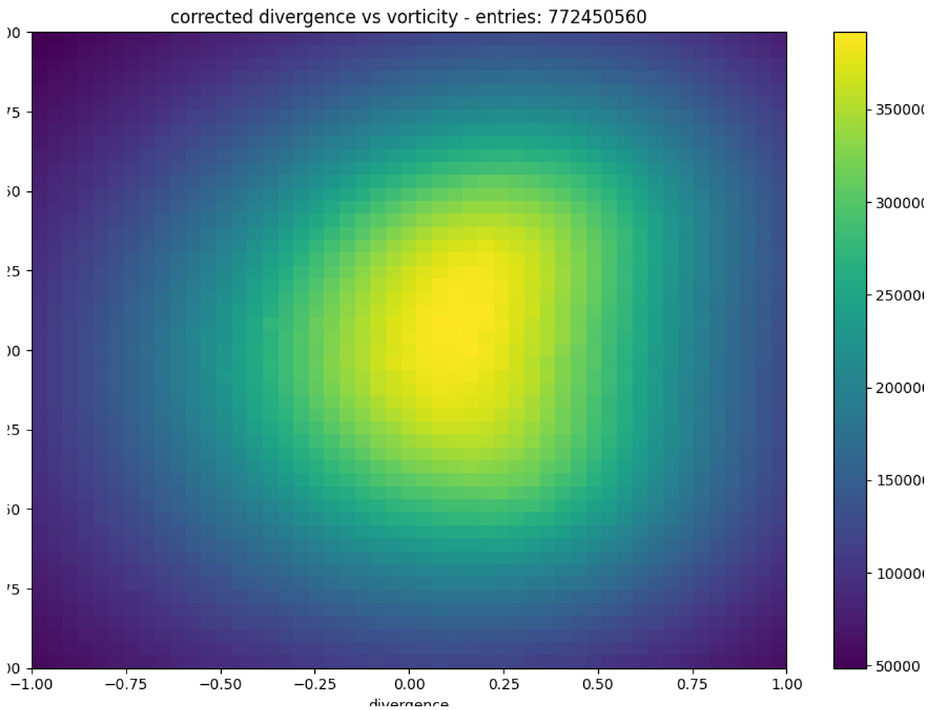


Data normalization

uncorrected



corrected



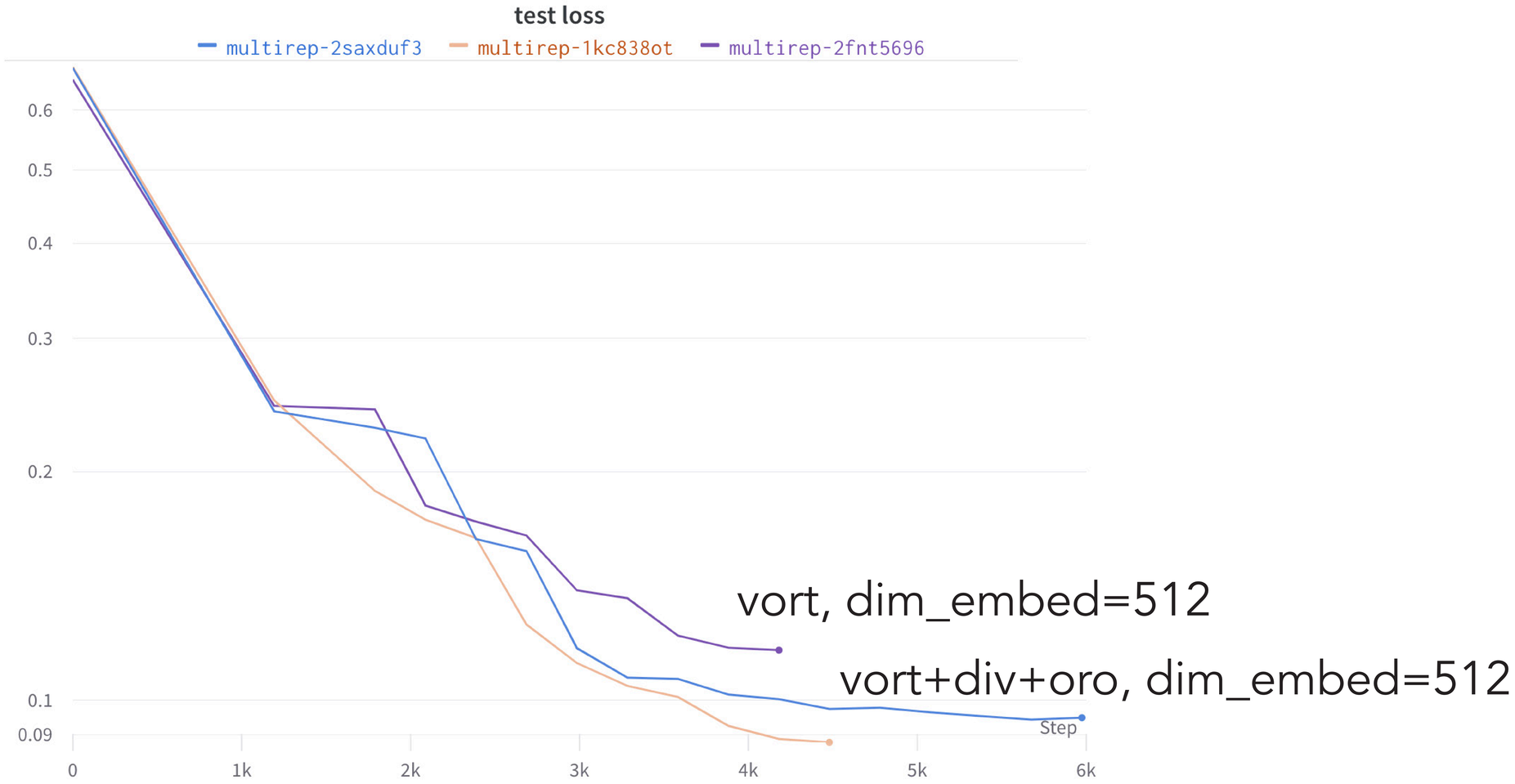
Multiformer

- Implemented and funtional
  - Some minor features could be added if necessary, e.g. variable number of coupling heads
  - Details of attention coupling still under investigation (see below)

```
cf.fields = { 'vorticity'  : [1, 768, ['divergence', 'orography'] ],
              'divergence' : [1, 256, ['vorticity', 'orography'] ],
              'orography'  : [0, 96, []] }
```

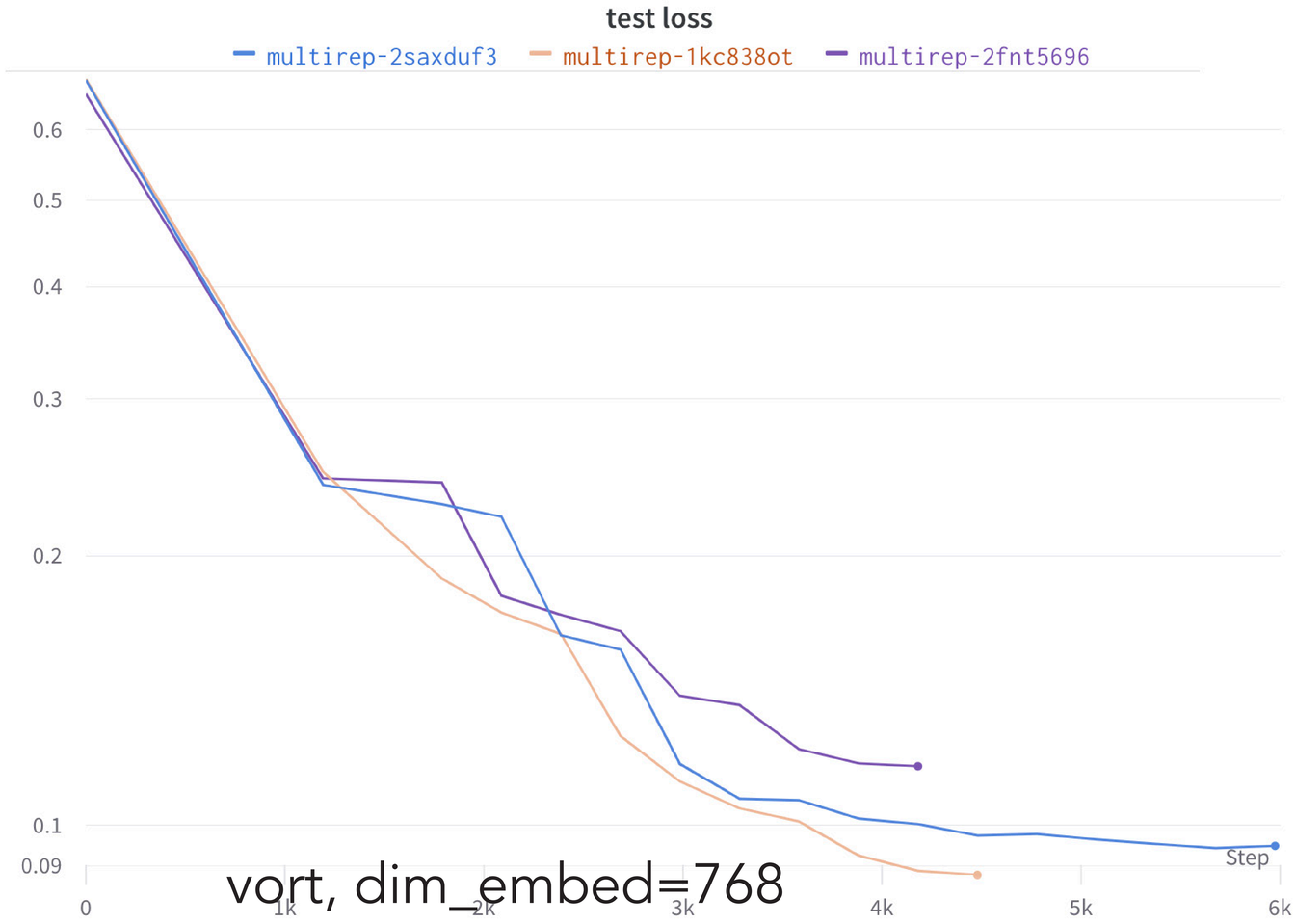
Training: vorticity vs. vorticity + divergence

blur=5



Training: vorticity vs. vorticity + divergence

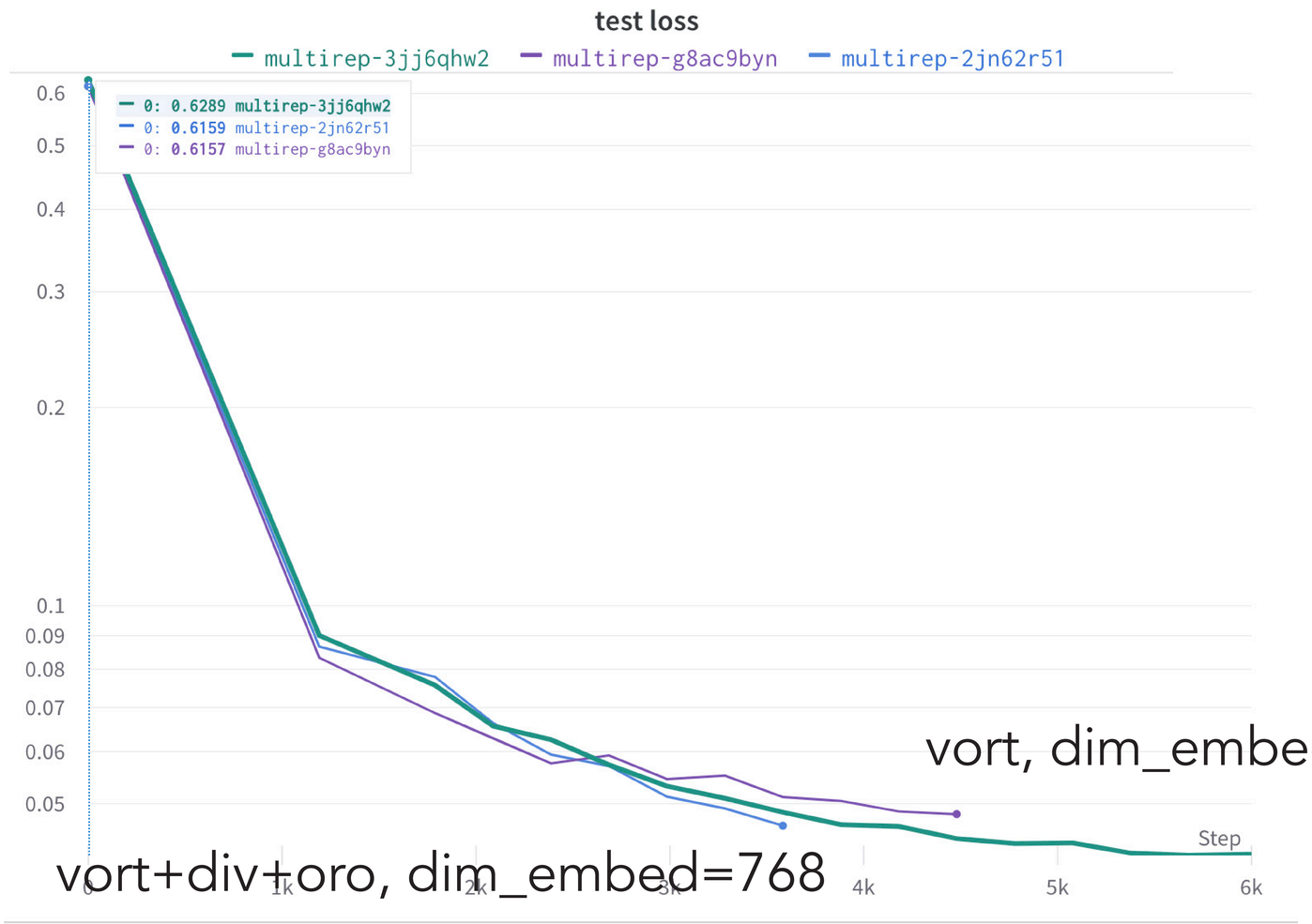
blur=5



vort, dim_embed=768

Training: vorticity vs. vorticity + divergence

blur=5



vort, dim_embed=1024

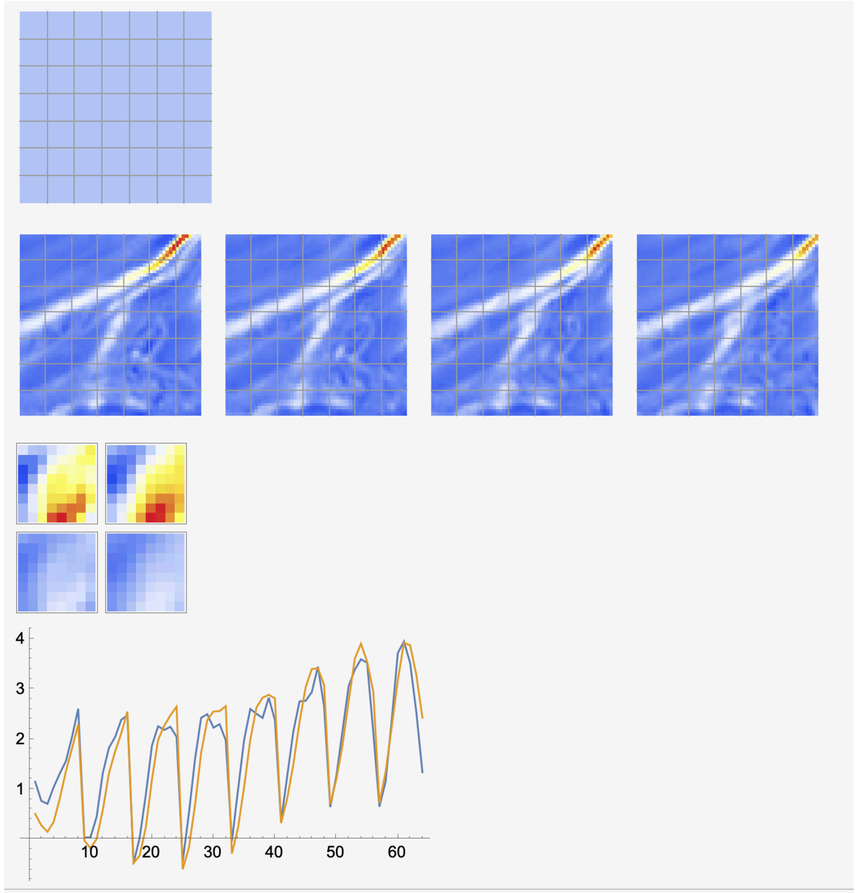
vort+div+oro, dim_embed=768

Multiformer

orography

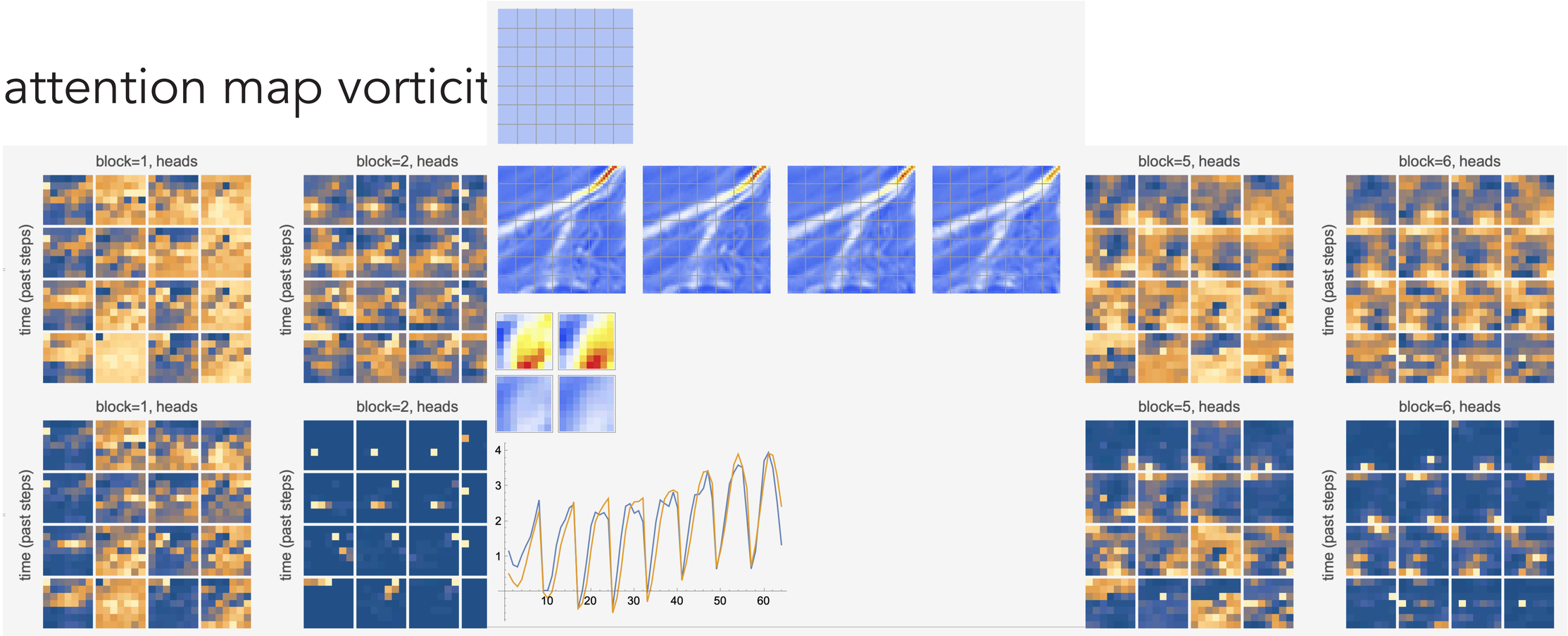
vorticity

prediction



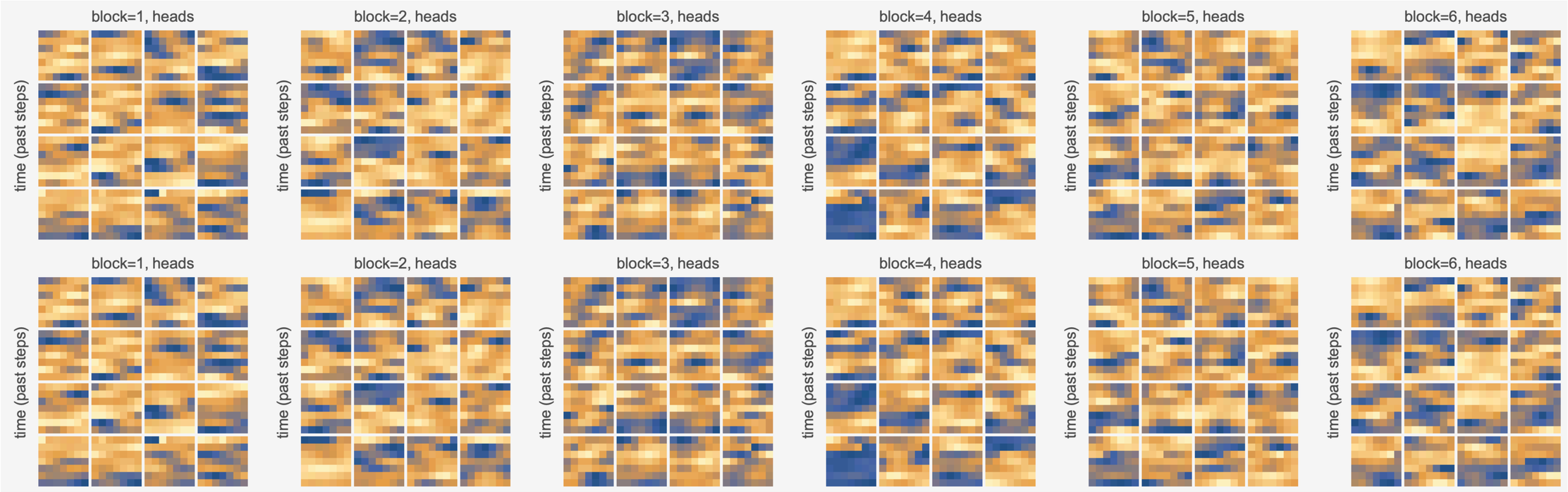
Multiformer

attention map vorticit



Multiformer

attention map orography

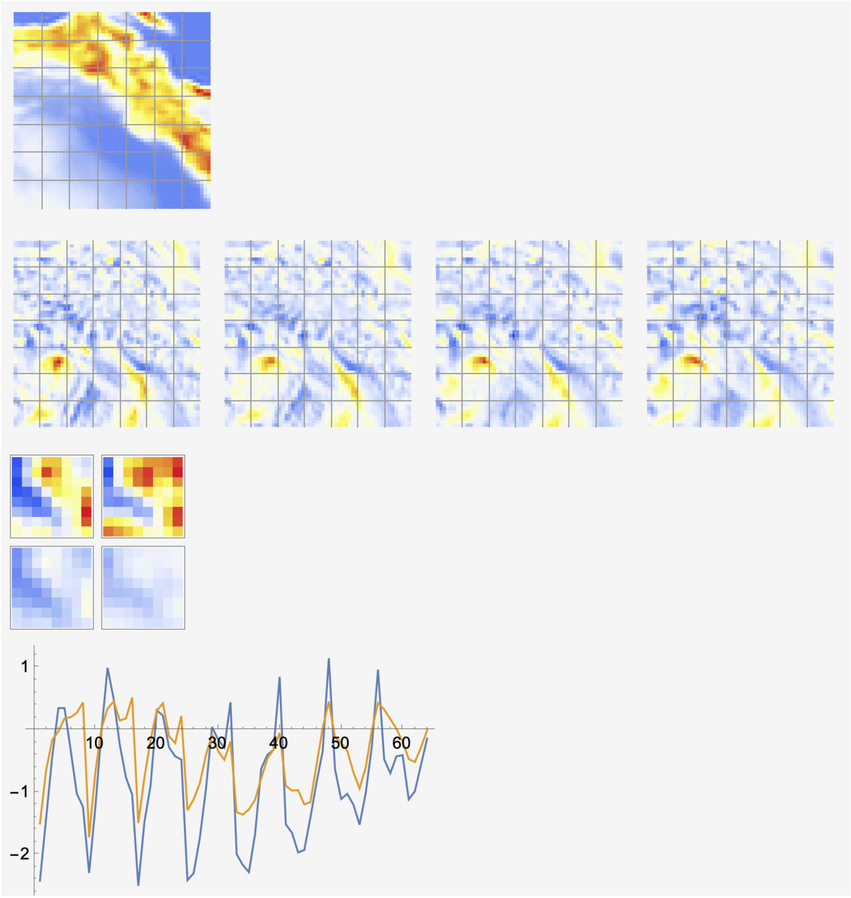


Multiformer

orography

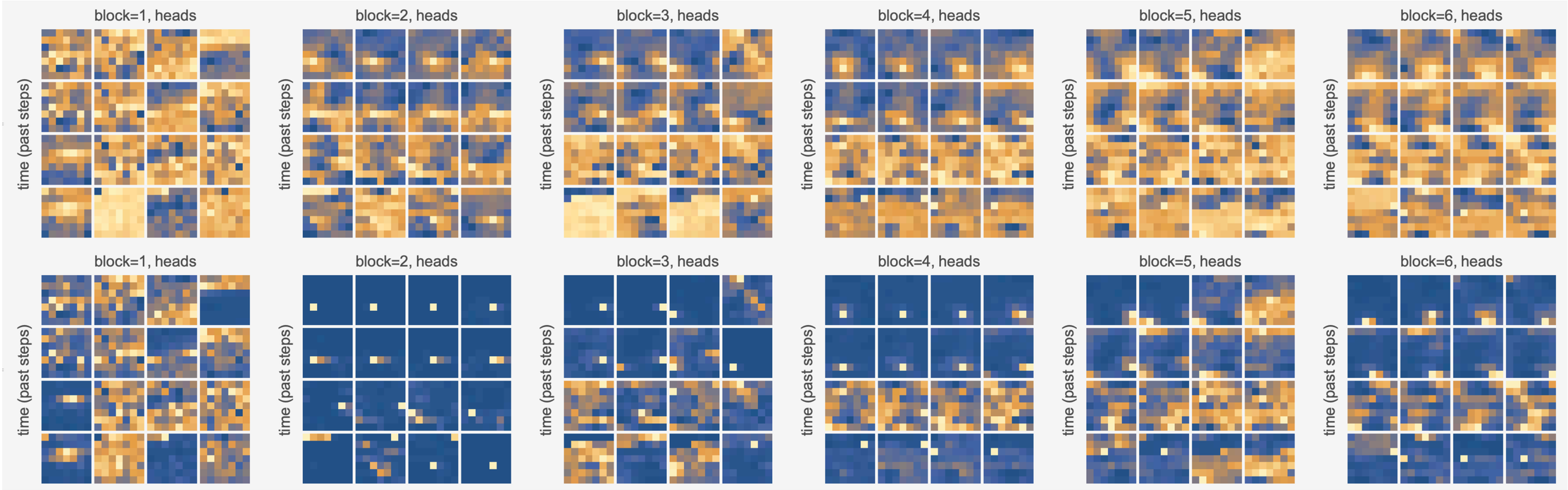
vorticity

prediction



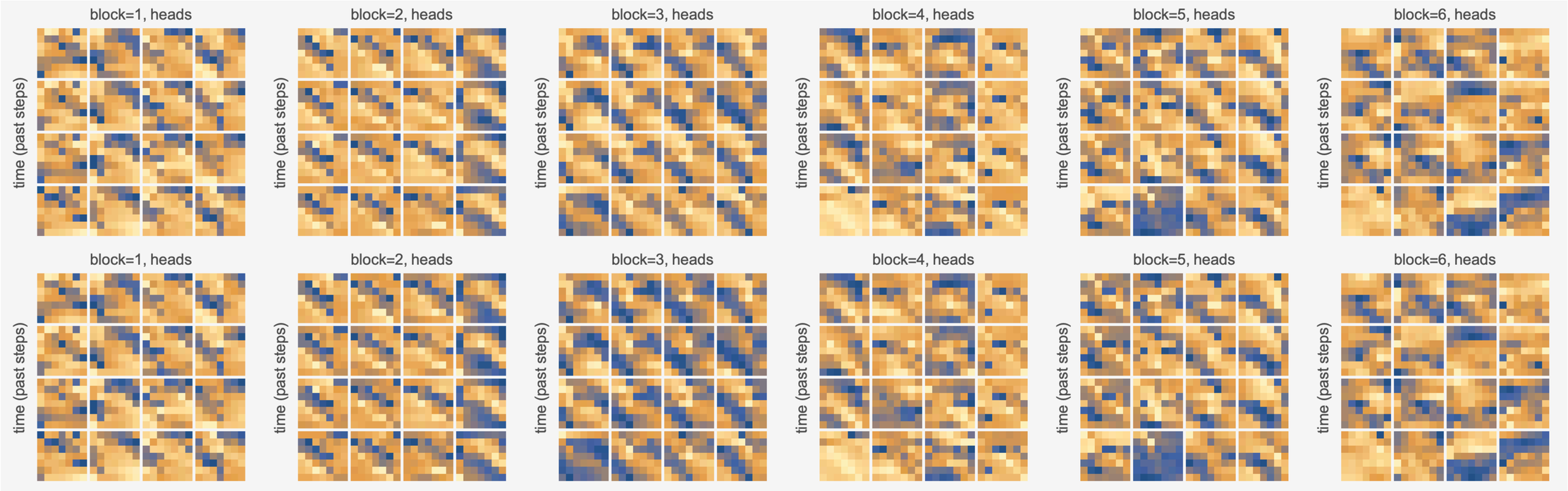
Multiformer

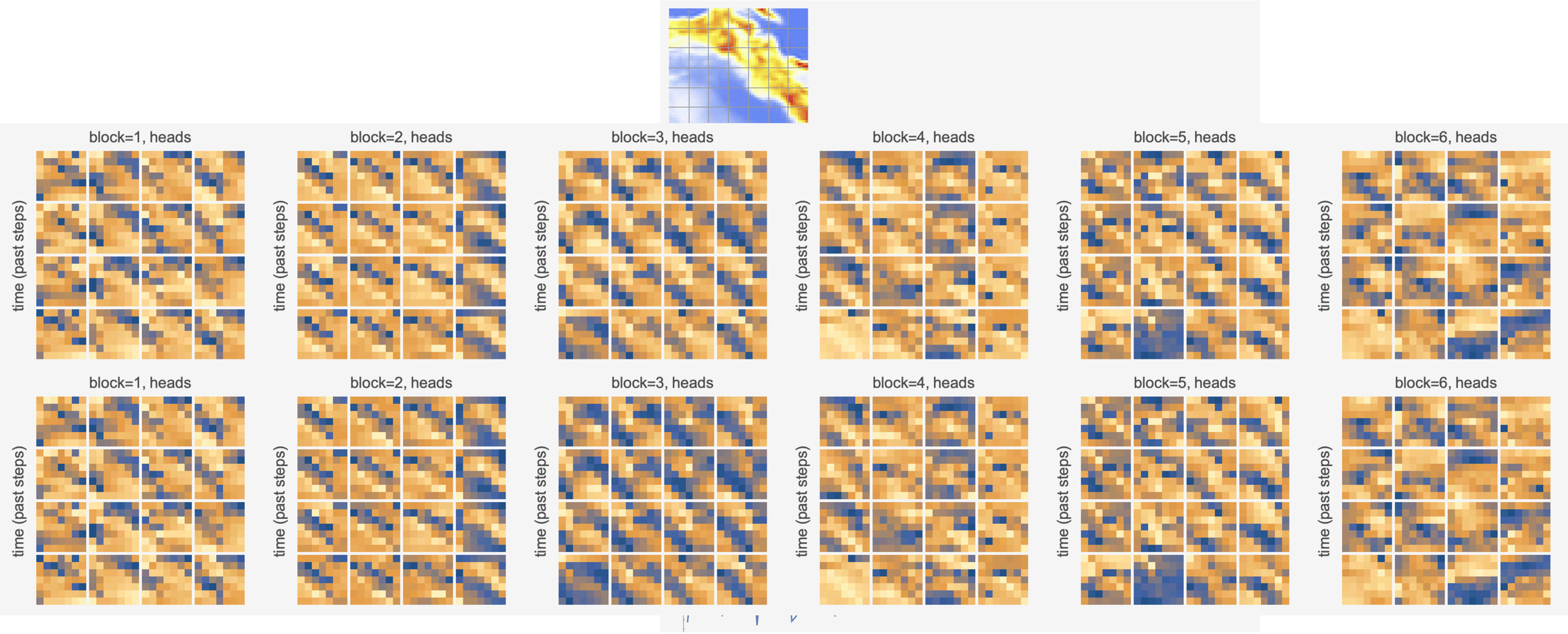
attention map vorticity



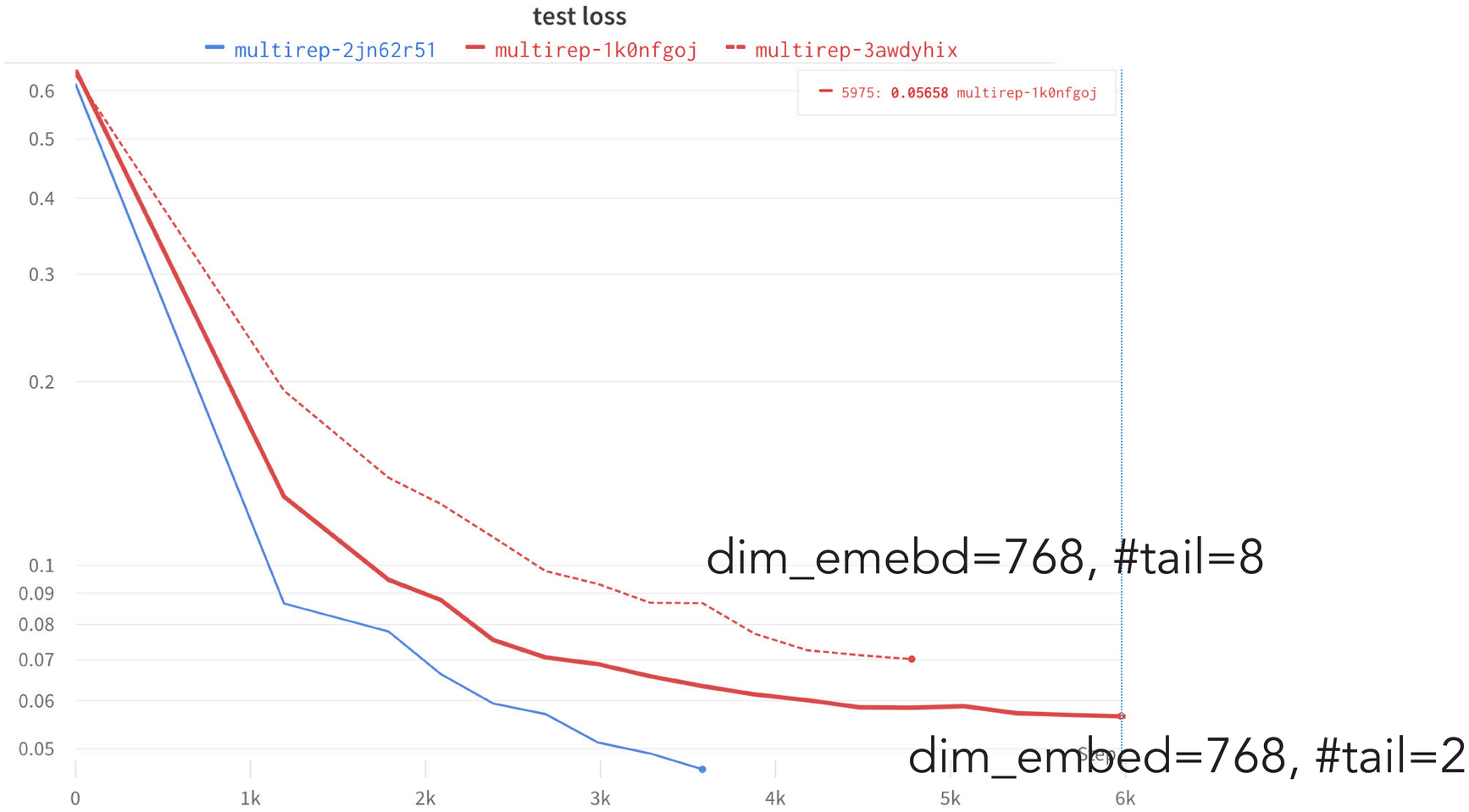
Multiformer

attention map orography



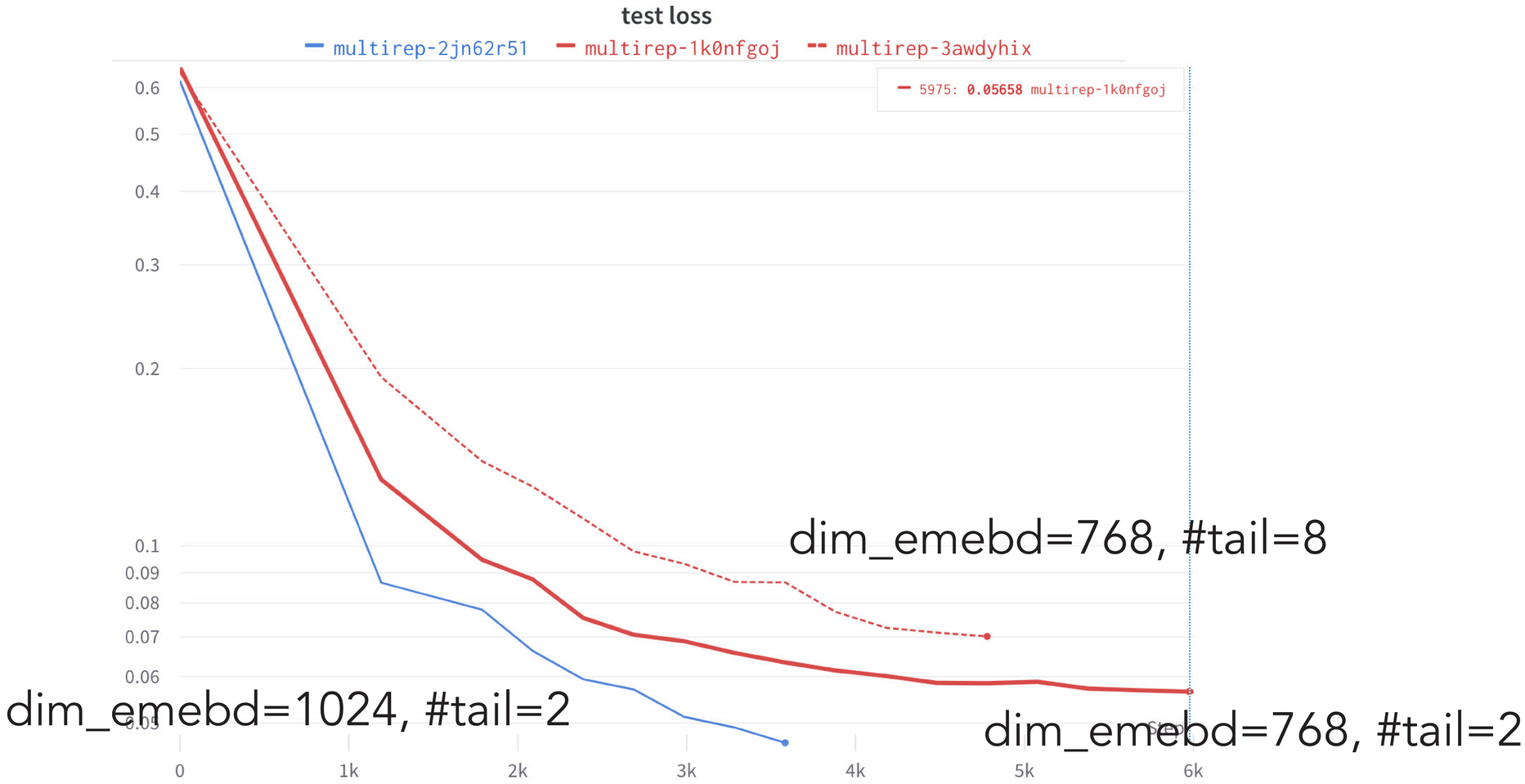


Training: tail net and embedding dimension



Training: tail net and embedding dimension

blur=5



Attention mechanism

- Standard encoder-decoder coupling (keys, value are fed in)
  - Mixes fields (but the same happens in PDEs)
  - In principle more expressive
  - Focus on field can be kept through loss

Attention mechanism

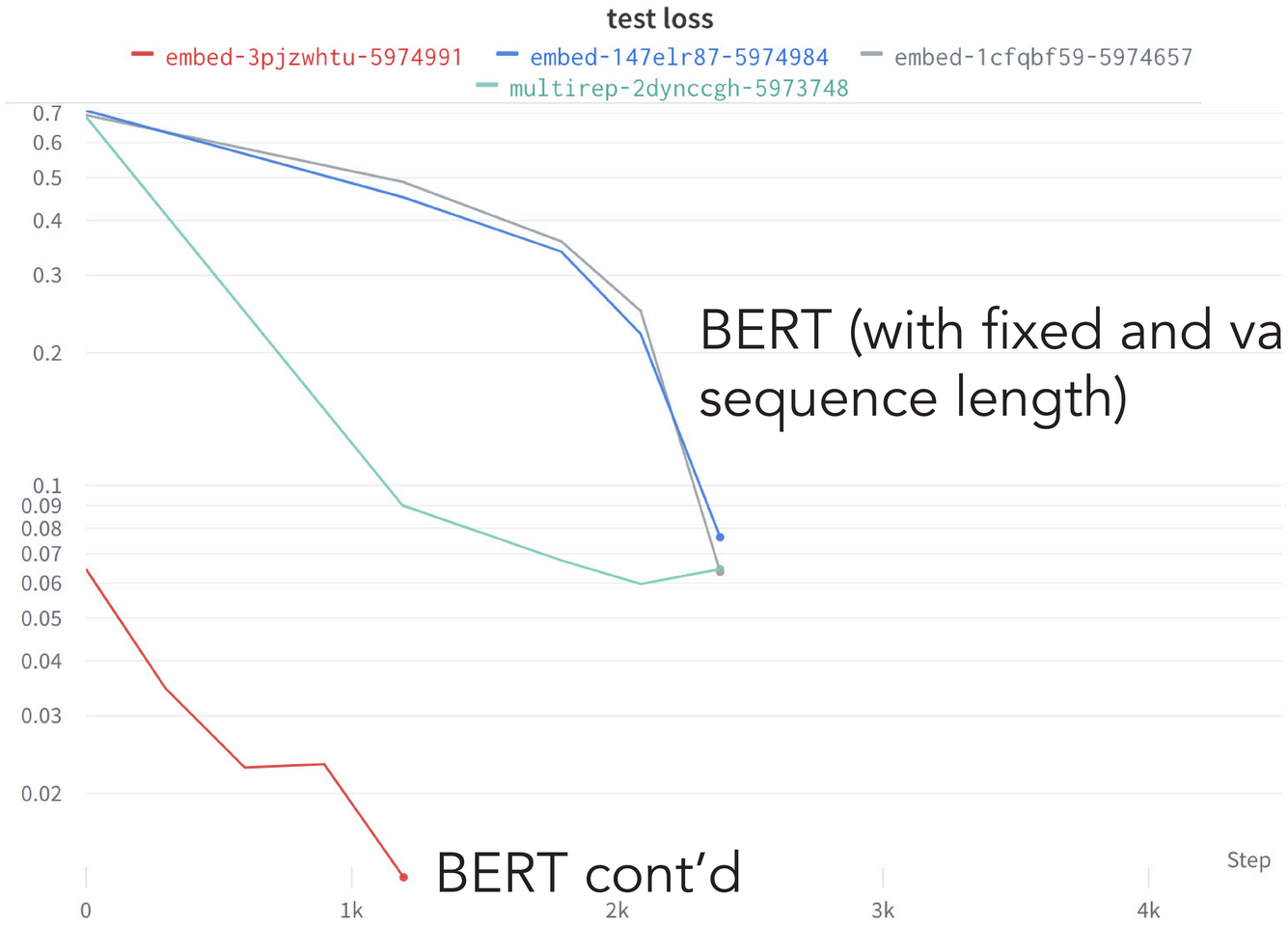
- Standard encoder-decoder coupling (keys, value are fed in)
  - Mixes fields (but the same happens in PDEs)
  - In principle more expressive
  - Focus on field can be kept through loss
- Query coupling (queries influence keys)
  - Fields are kept separate but still influence each other
  - Facilitates physical interpretation (fields are “clean”)
  - Difficulties when token size differs

Embedding network

- Implemented and functional
- Exploits that a transformer can have a variable length input
- Class token as compressed representation
- BERT-style training
  - Tail network predicts masked token just using class token
  - Variable size of space-time neighborhood (per batch)
- Proper evaluation still open

Training: BERT versus forecasting

blur=5

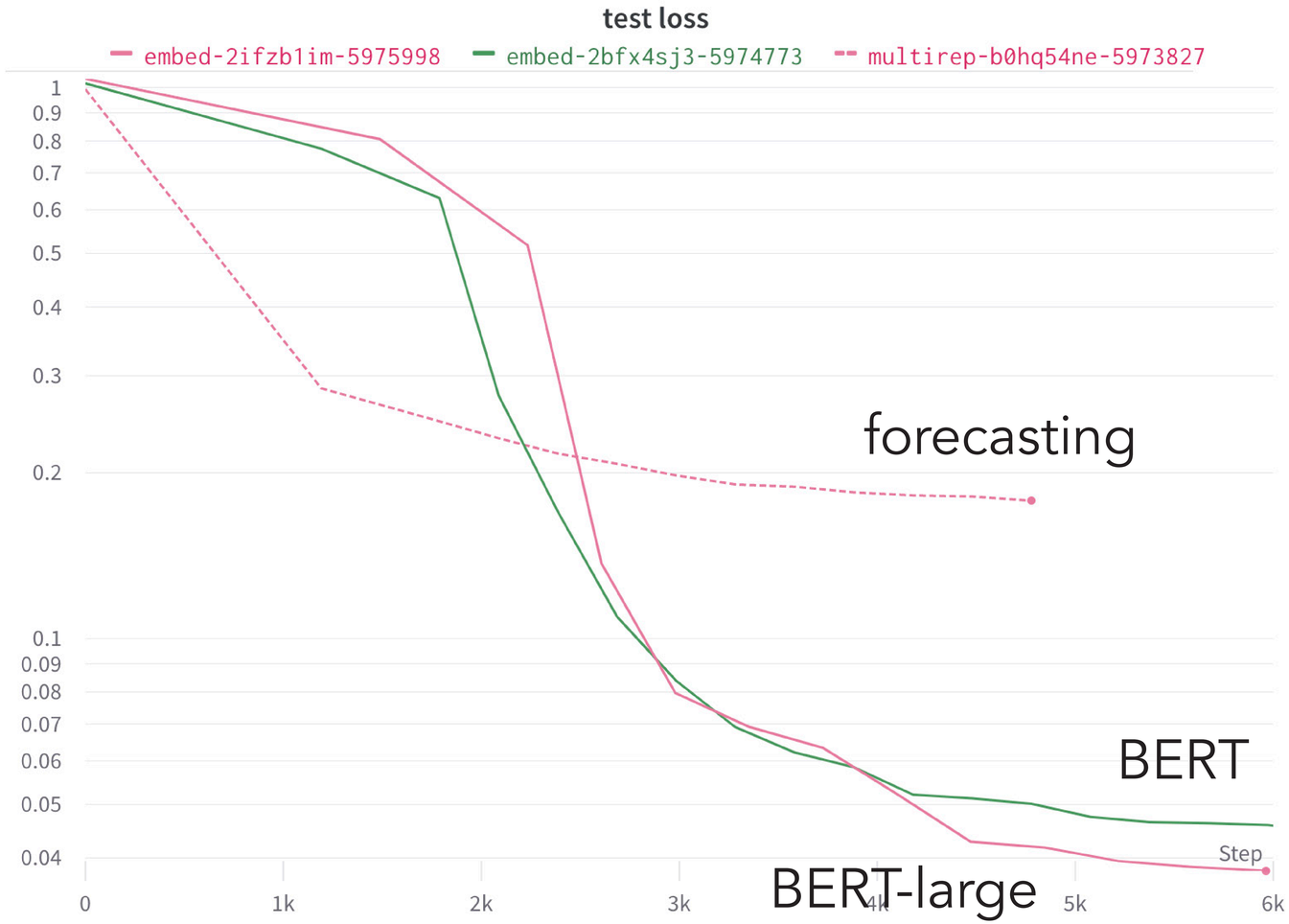


BERT (with fixed and variable sequence length)

BERT cont'd

Training: BERT versus forecasting

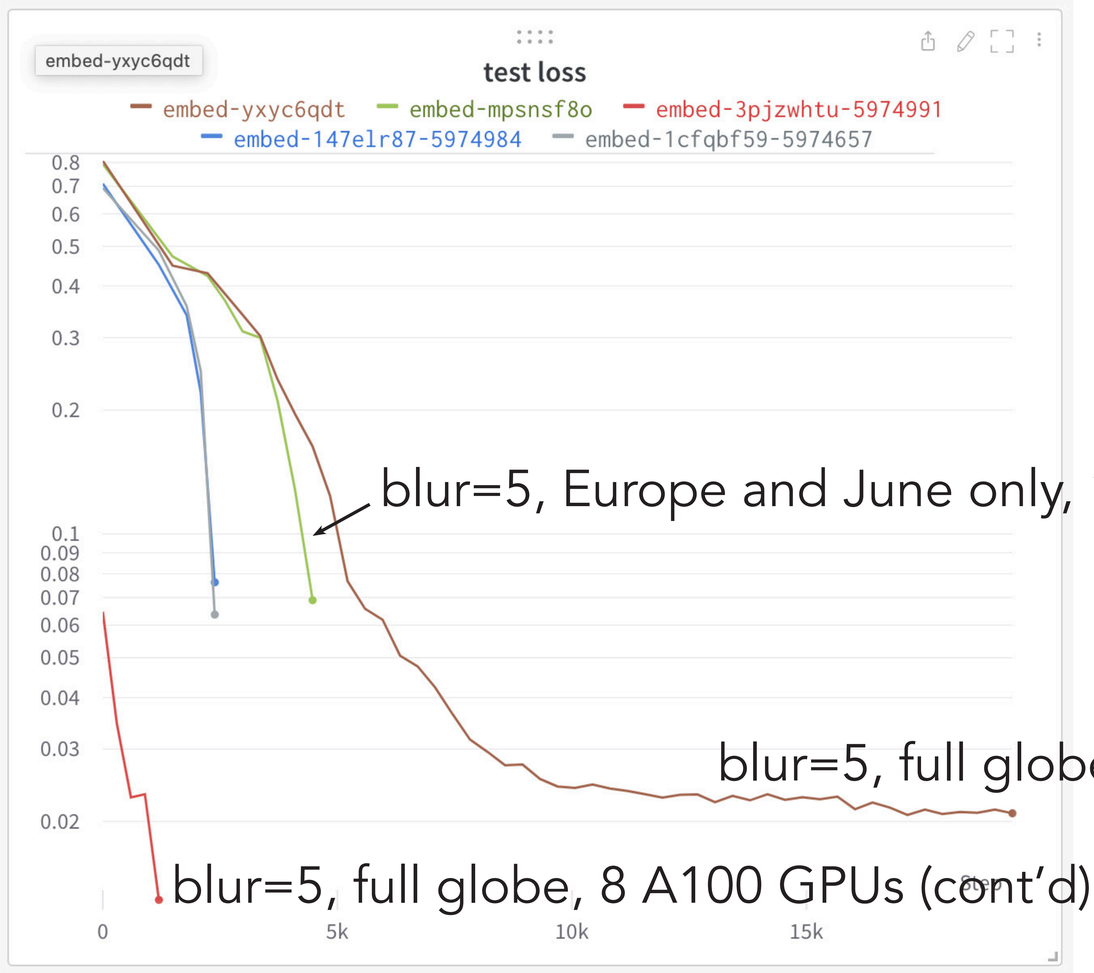
blur=1



Challenges

- Slow development due to wait times on batch system
  - Long (2h+) runs required to see differences between models
  - Unclear how to construct restricted but representative development mode (restriction of space + time is not effective)
- Limited compute time budget

Challenges



Challenges

- Slow development due to wait times on batch system
  - Long (2h+) runs required to see differences between models
  - Unclear how to construct restricted but representative development mode (restriction of space + time is not effective)
- Limited compute time budge

=> ~~Apply for Google cloud credit~~? 2-4 weeks decision time

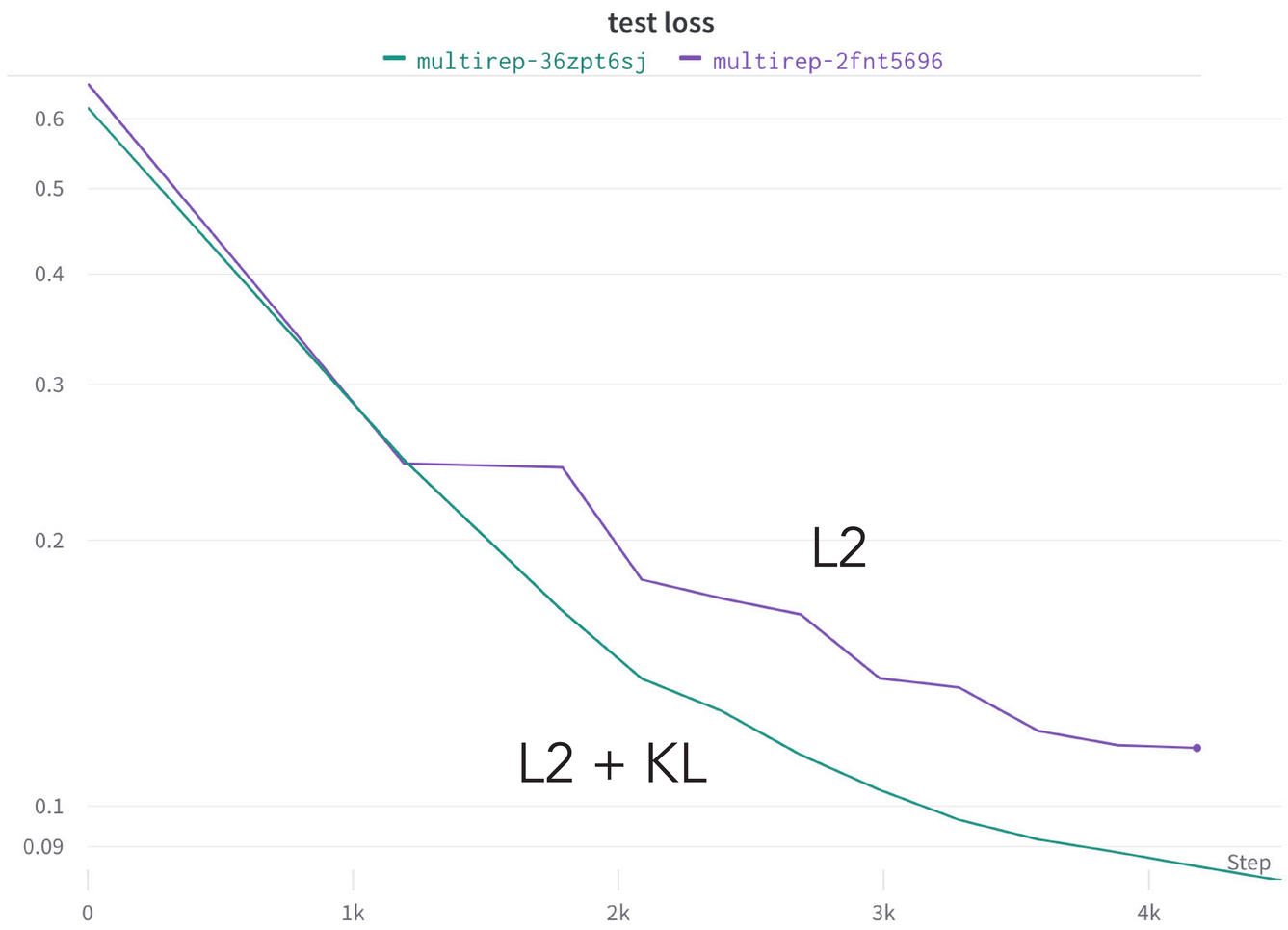
Challenges

- Comparisons between models are often inconclusive
- Architectual changes seem to have little effect
- No over-fitting but larger model also does not improve predictions very much: bottleneck currently unclear

Loss function

- "Never train the mean"
  - Using L2 and KL loss improves results by about 20%

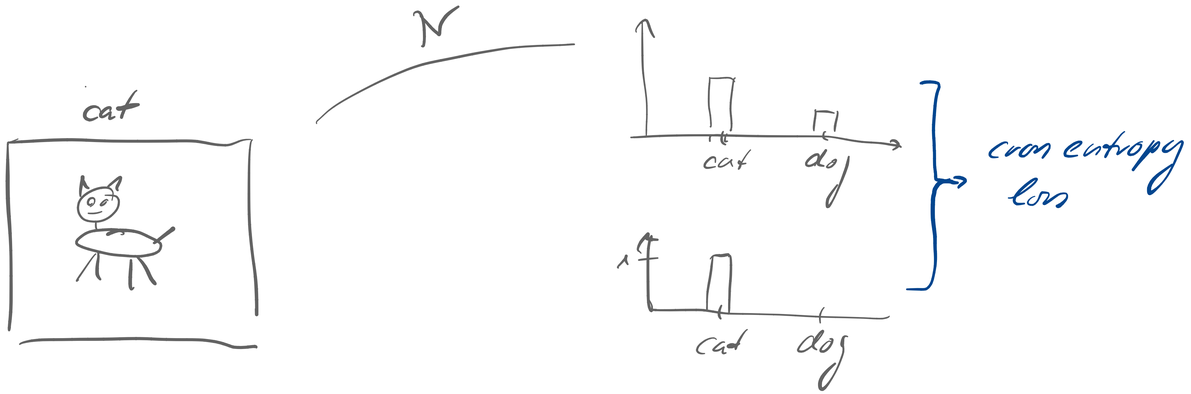
Loss function



Loss function

- "Never train the mean"
  - Using L2 and KL loss improves results by about 20%
- Classification
  - Labels are considered as Kronecker probability distribution
  - Network predicts distribution and cross entropy loss for network training

Loss function



Loss function

arXiv:1609.08144.

Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. 2014. How transferable are features in deep neural networks? In *Advances in neural information processing systems*, pages 3320–3328.

Adams Wei Yu, David Dohan, Minh-Thang Luong, Rui Zhao, Kai Chen, Mohammad Norouzi, and Quoc V Le. 2018. QANet: Combining local convolution with global self-attention for reading comprehension. In *ICLR*.

Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 2018. Swag: A large-scale adversarial dataset for grounded commonsense inference. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

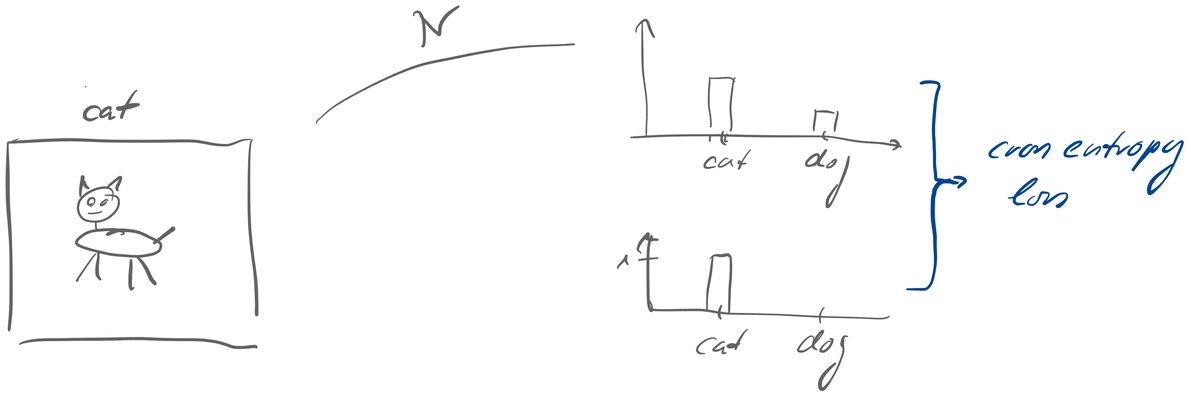
Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE international conference on computer vision*, pages 19–27.

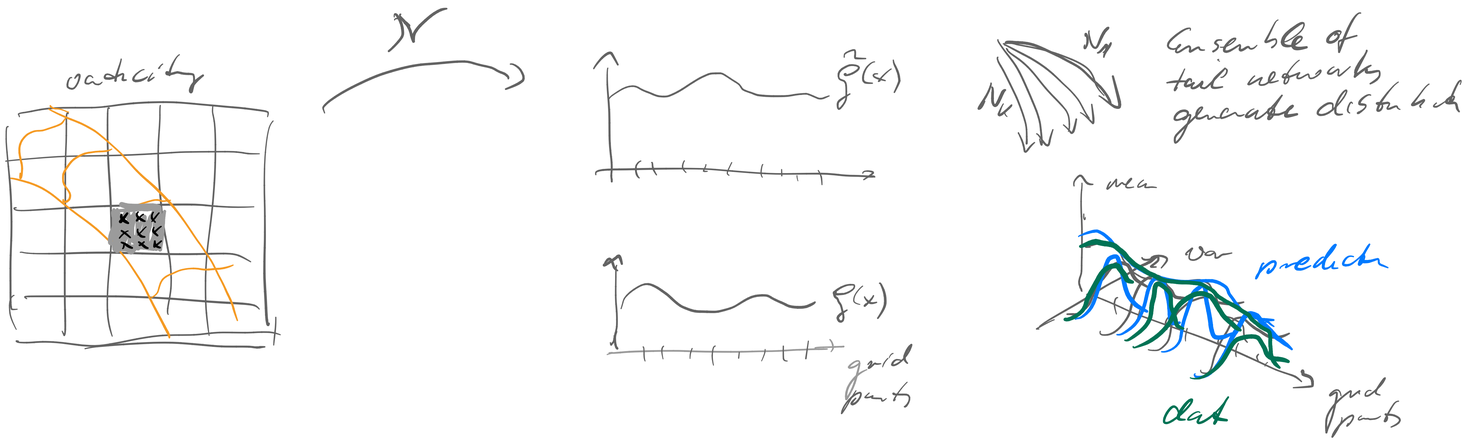
Appendix for "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding"

- 80% of the time: Replace the word with the `[MASK]` token, e.g., `my dog is hairy` → `my dog is [MASK]`
- 10% of the time: Replace the word with a random word, e.g., `my dog is hairy` → `my dog is apple`
- 10% of the time: Keep the word unchanged, e.g., `my dog is hairy` → `my dog is hairy`. The purpose of this is to bias the representation towards the actual observed word.

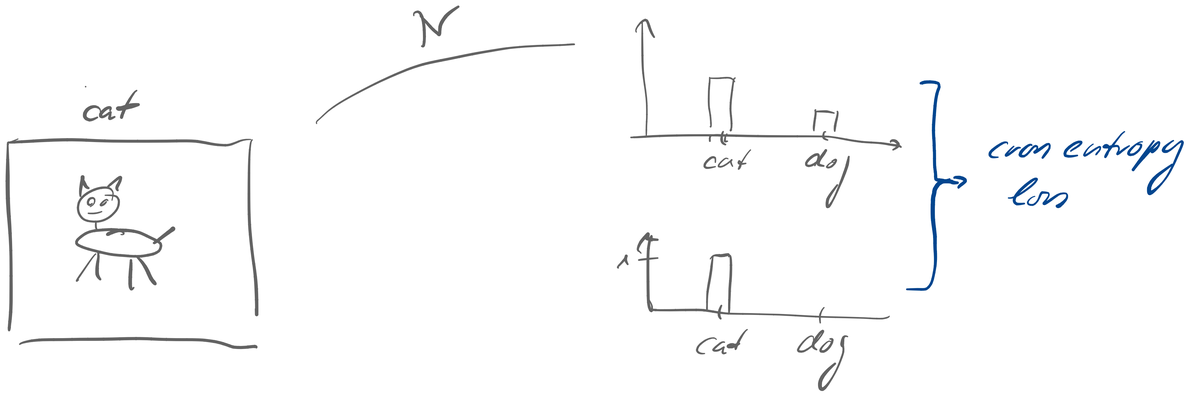
The advantage of this procedure is that the Transformer encoder does not know which words it will be asked to predict or which have been replaced by random words, so it is forced to keep a distributional contextual representation of *every* input token. Additionally, because random replacement only occurs for 1.5% of all tokens (i.e., 10% of 15%), this does not seem to harm the model's language understanding capability. In Section C.2, we evaluate the impact this procedure.

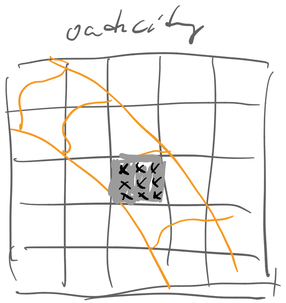
Loss function



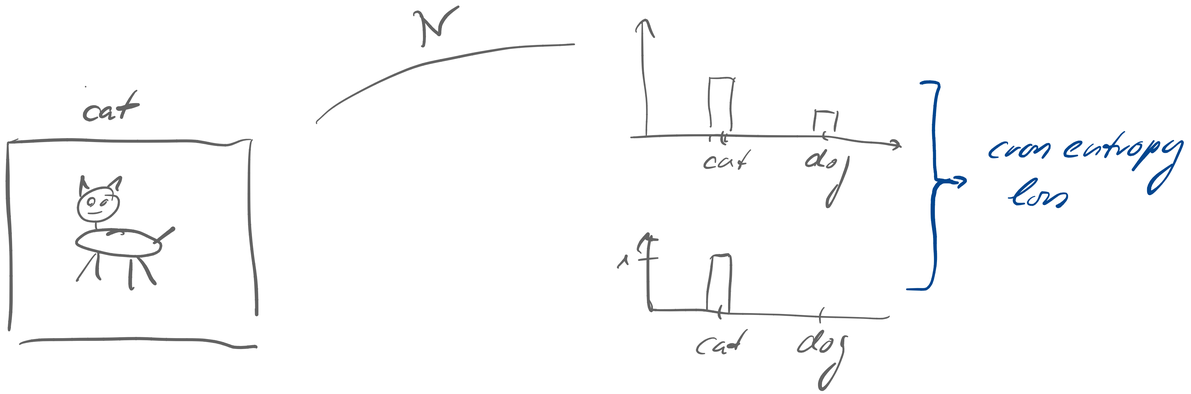


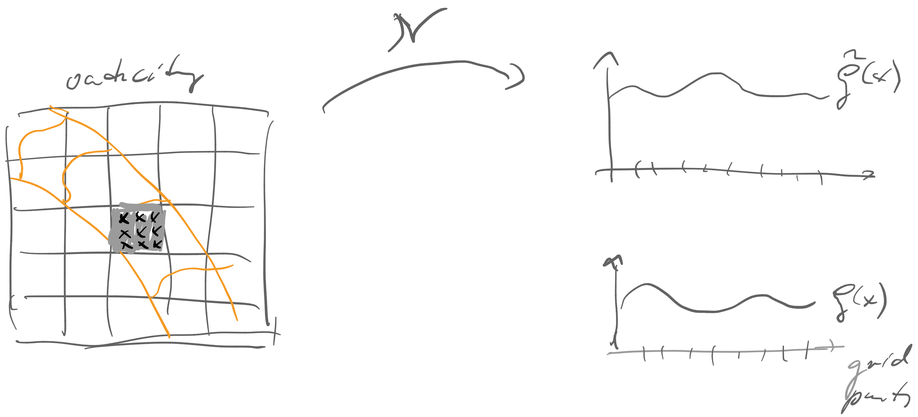
Loss function



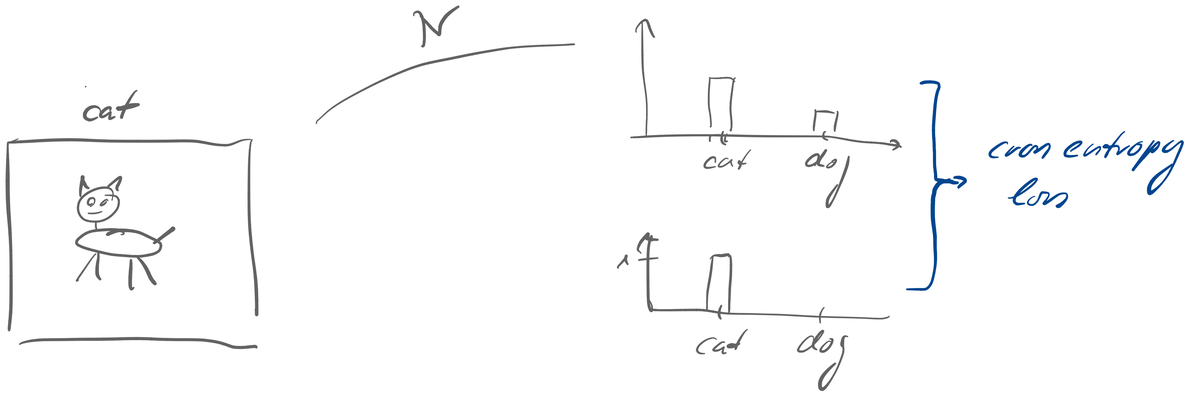


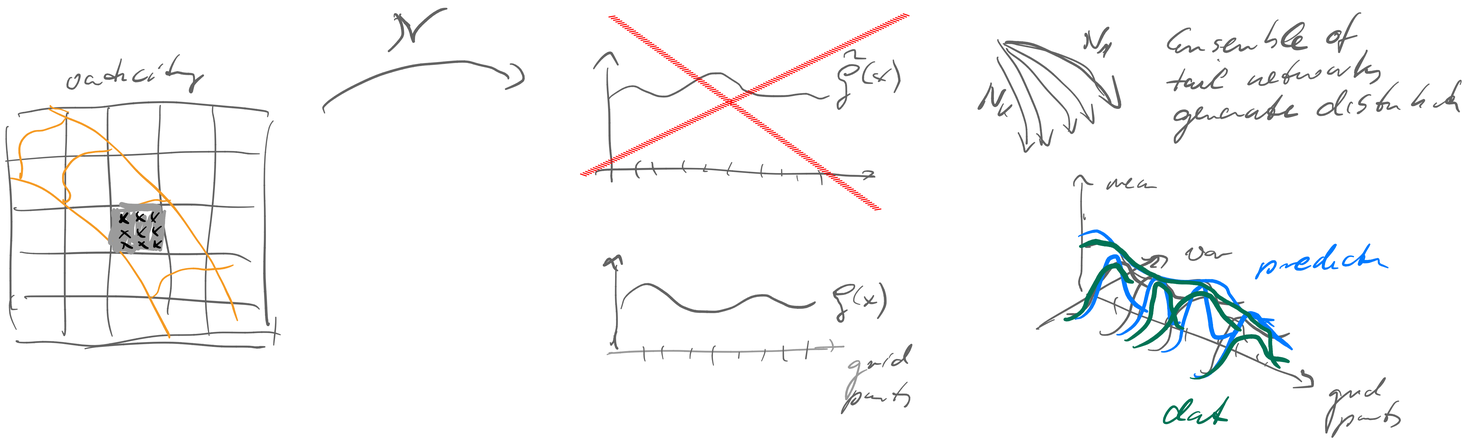
Loss function





Loss function





Next steps

- BERT training for multiformer
  - Integrate forecasting
- Improve performance of data loading
- Probabilistic loss

Next steps

- Embedding network
  - Train for all fields
  - Scale to large neighborhood
- Separable space-time attention
- Larger space-time neighborhoods

Next steps

- Downstream applications
  - Down scaling, hurricane tracking, ...
- Train with absolute or relative error?
- Move towards scientific evaluation
  - Proper statistical analysis (e.g. of spatial distribution of error)
  - How to turn attention maps into a principled tool?

Publication plan

- Workshops
  - AGU, ECMWF, Neurips, Italian meterological society
- Papers
  - ICLR?
  - Attention maps, BERT for space-time, ...
  - Science journal?
  - Downstream applications?