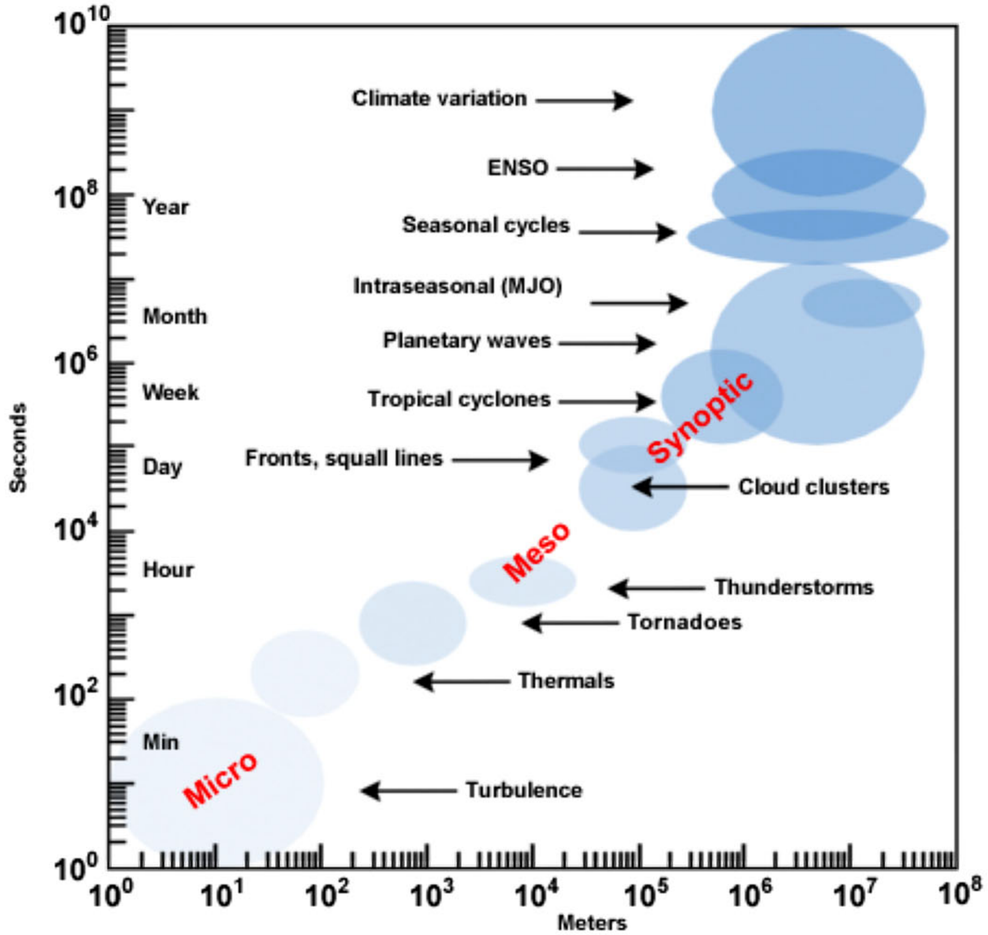
Transformers and representation learning for science and engineering

Christian Lessig, Otto-von-Guericke-Universität Magdeburg

My background

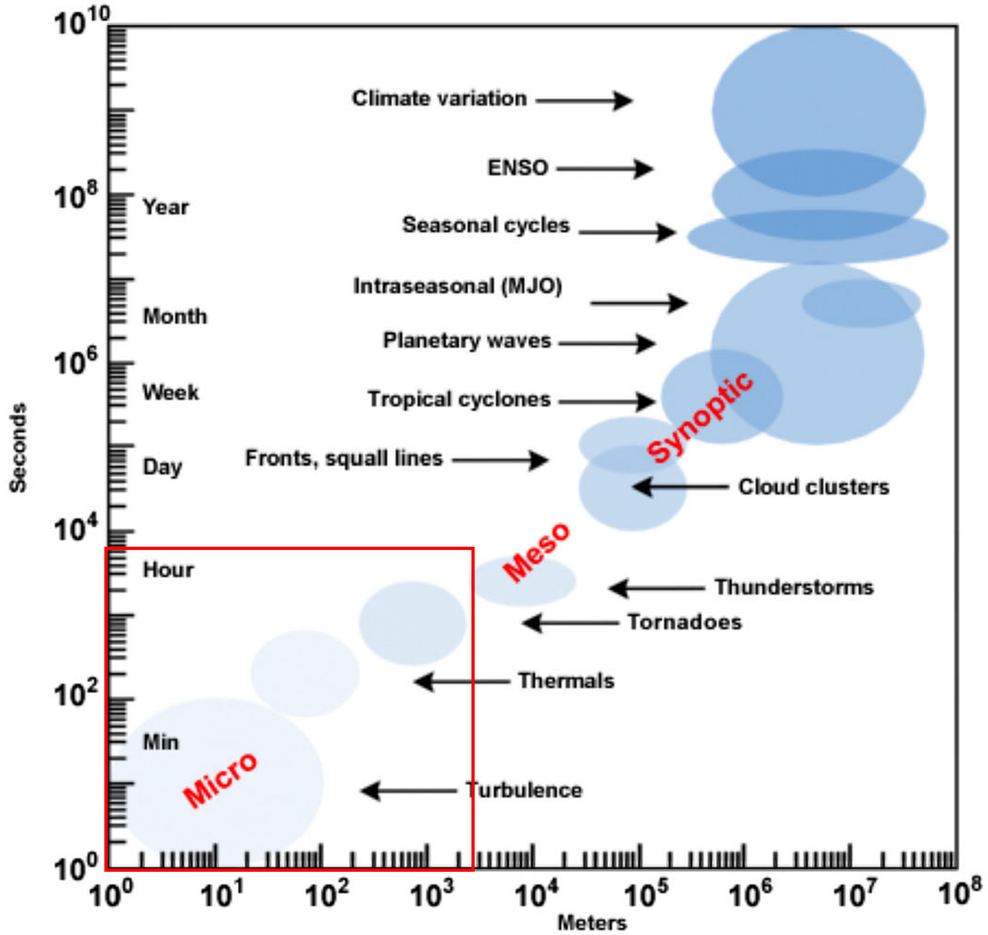
- Ph.D. on simulations in computer graphics
- Classical numerical modeling and simulations of light, fluids, ... for scientific appliations
  - Adaptive schemes
  - Structure-preserving (Lie group) integrators
- Part of DFG-funded CRC-287 “BULK Reaction”
  - Large scale simulation of reacting, gaseous flows in granular assemblies for industrial applications

Motivation: weather and climate



From V. M. Galfi, V. Lucarini, F. Ragone, and J. Wouters. Applications of large deviation theory in geophysical fluid dynamics and climate science. La Rivista del Nuovo Cimento, 44(6):291–363, 2021.

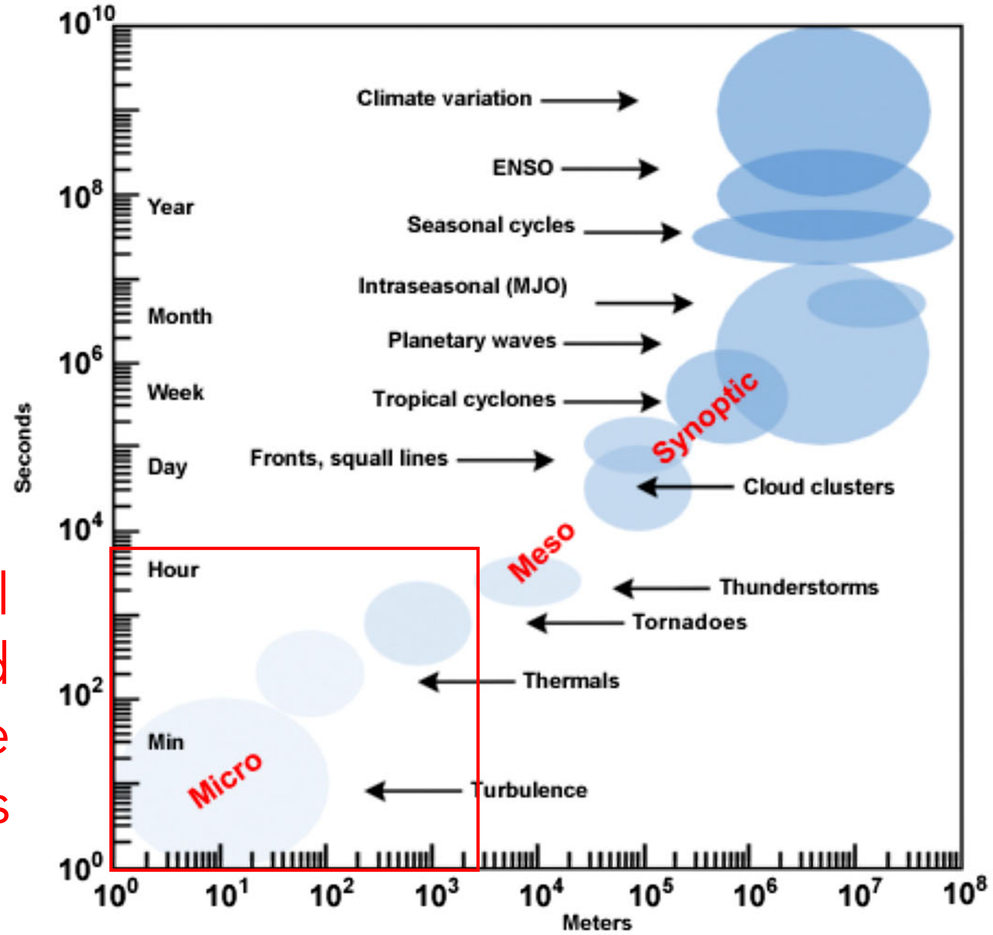
Motivation: weather and climate



From V. M. Galfi, V. Lucarini, F. Ragone, and J. Wouters. Applications of large deviation theory in geophysical fluid dynamics and climate science. La Rivista del Nuovo Cimento, 44(6):291–363, 2021.

Relevant phenomena span 10 orders of magnitude in space and time

Motivation: weather and climate



From V. M. Galfi, V. Lucarini, F. Ragone, and J. Wouters. Applications of large deviation theory in geophysical fluid dynamics and climate science. La Rivista del Nuovo Cimento, 44(6):291–363, 2021.

Relevant phenomena span 10 orders of magnitude in space and time

Often no physical models at all and usually no effective coarse-scale models

Motivation: weather and climate

- Large amounts of data available in the Earth sciences:
  - ERA5: ≈6 PB
  - CMIP6: ≈100 PB
  - MetOp-SG: 8 x 864 GB/day (80 Mbit/s)
  - OCEAN5: ≈4 PB

Motivation: weather and climate

- Large amounts of data available in the Earth sciences:
  - ERA5: ≈6 PB
  - CMIP6: ≈100 PB
  - MetOp-SG: 8 x 864 GB/day (80 Mbit/s)
  - OCEAN5: ≈4 PB

} growing fast

Motivation: weather and climate

- Large amounts of data available in the Earth sciences:
  - ERA5: ≈6 PB
  - CMIP6: ≈100 PB
  - MetOp-SG: 8 x 864 GB/day (80 Mbit/s)
  - OCEAN5: ≈4 PB

  } growing fast

- Observational or quasi-observational data with effects and phenomena not captured in known analytic models

Motivation: weather and climate

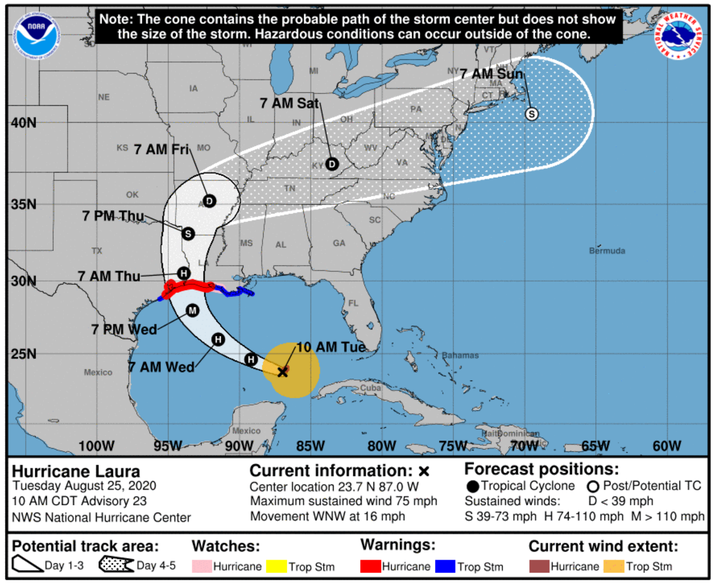
- How to use this data for machine learning?

Motivation: weather and climate

- How to use this data for machine learning?
  - Most data is unlabeled
  - Super-computing infrastructure required for storing and processing
  - Unclear how to ensure that learned models are physically consistent

Motivation: weather and climate

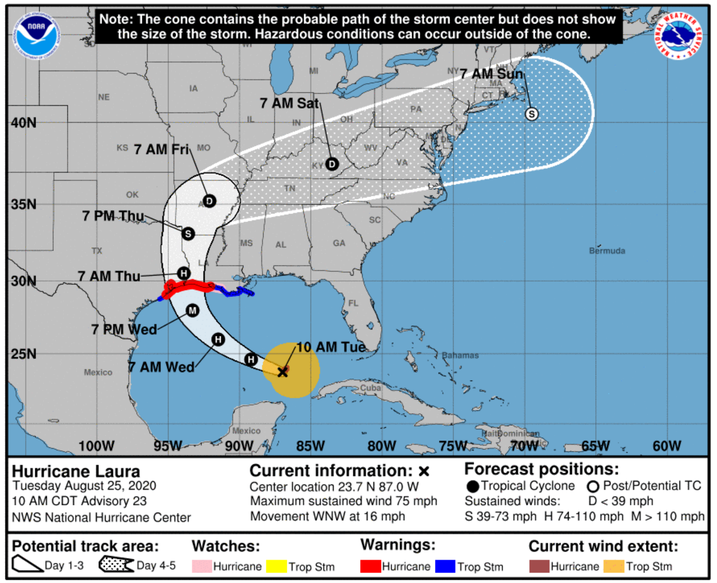
- Example: hurricane tracking
  - Large importance for immediate effects and climate projections



https://www.nhc.noaa.gov/aboutnhcgraphics.shtml

Motivation: weather and climate

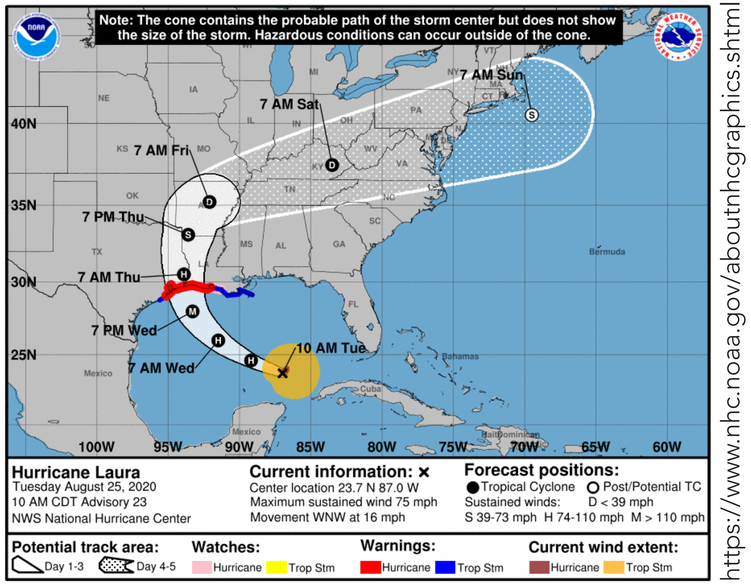
- Example: hurricane tracking
  - Large importance for immediate effects and climate projections
  - NOAA HURDAT2 Atlantic hurricane database: 6.5 MB



https://www.nhc.noaa.gov/aboutnhcgraphics.shtml

Motivation: weather and climate

- Example: hurricane tracking
  - Large importance for immediate effects and climate projections
  - NOAA HURDAT2 Atlantic hurricane database: 6.5 MB



https://www.nhc.noaa.gov/aboutnhcgraphics.shtml

⇒ Can we use large amounts of unlabeled data to augment the very small amounts of labeled hurricane tracking data?

Self-supervised representation learning

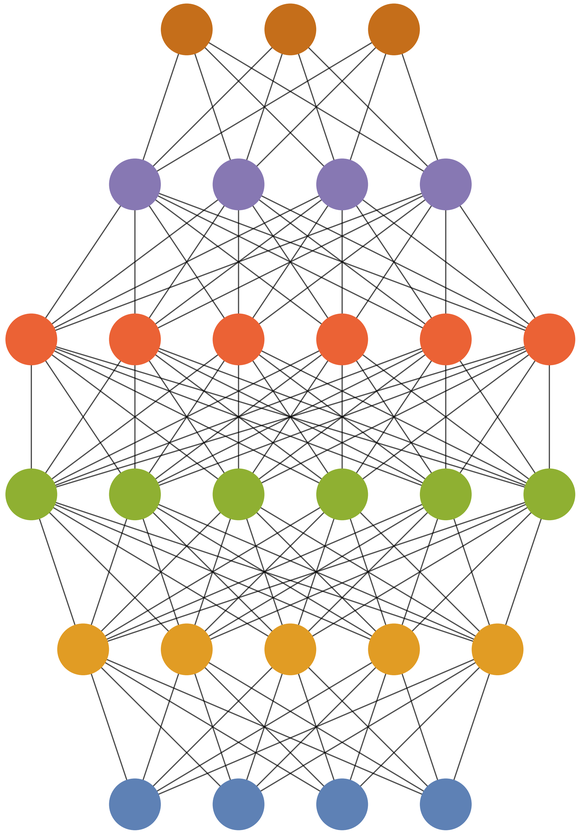
Self-supervised representation learning

- Yoshua Bengio: “Humans develop representations and abstractions to enable problem-solving and reasoning; our computers should do the same.”[1]

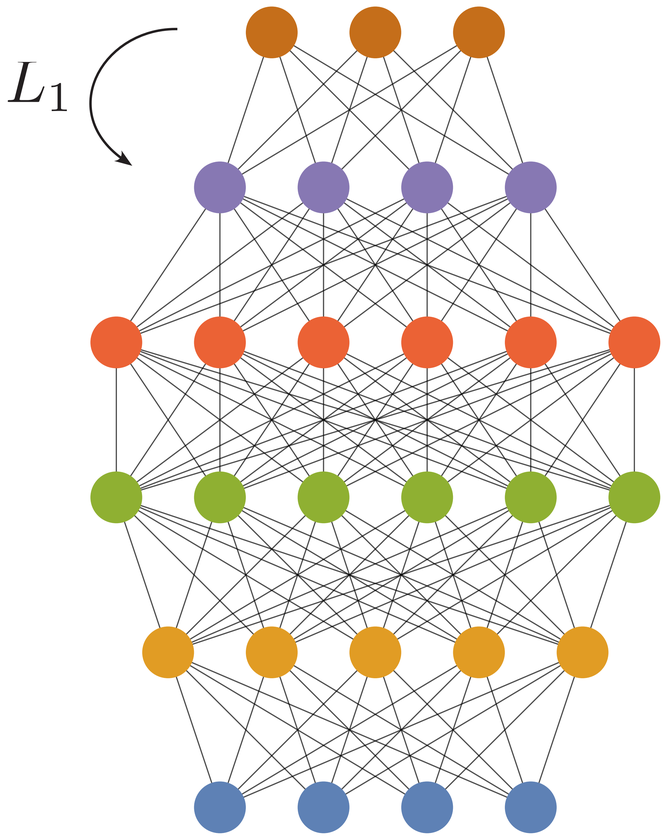
- Yann LeCun: “Self-supervised learning: The dark matter of intelligence“[2]

[1] http://www.iro.umontreal.ca/~bengioy/talks/icml2012-YB-tutorial.pdf
[2] https://ai.facebook.com/blog/self-supervised-learning-the-dark-matter-of-intelligence/

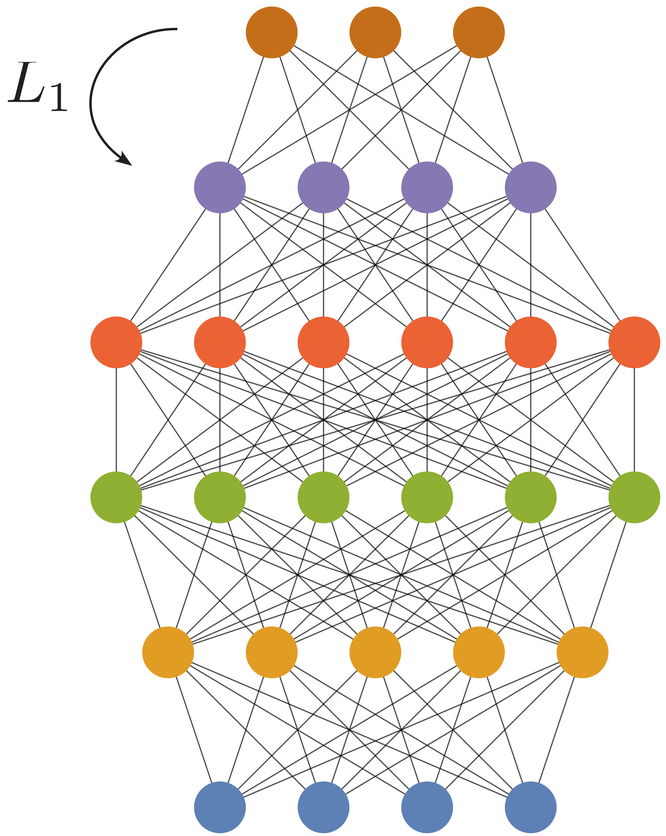
Self-supervised representation learning



Self-supervised representation learning

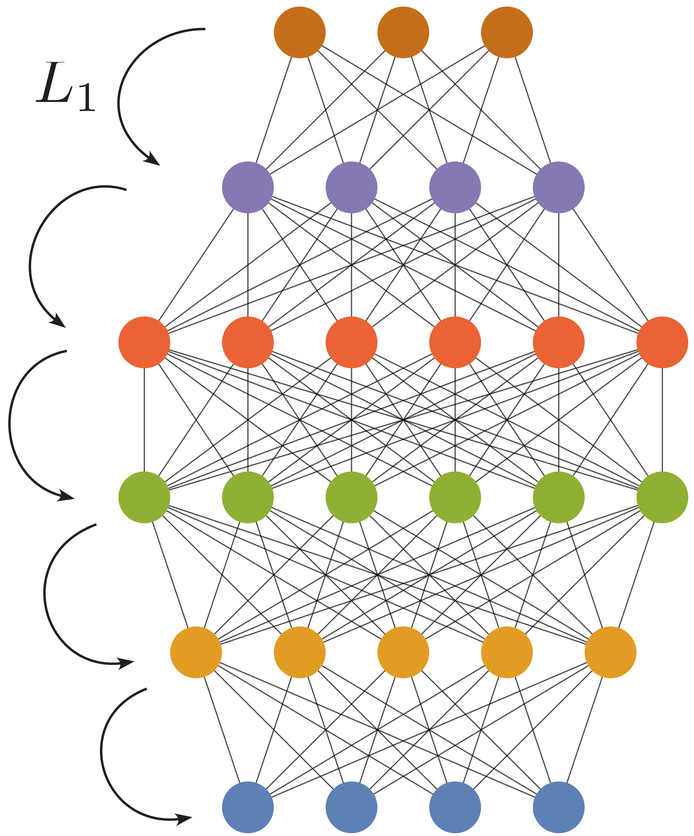


Self-supervised representation learning



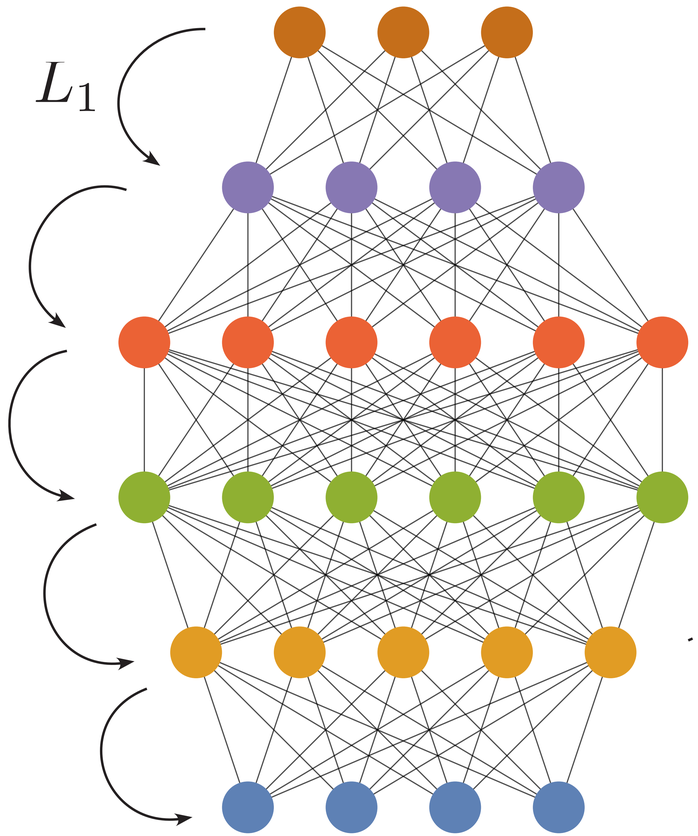
$$L_1 : \mathbb{R}^n \to \mathbb{R}^m$$

Self-supervised representation learning

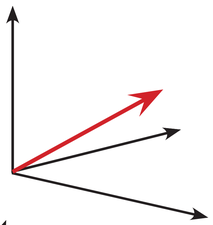


$$L_j : \mathbb{R}^n \to \mathbb{R}^m$$

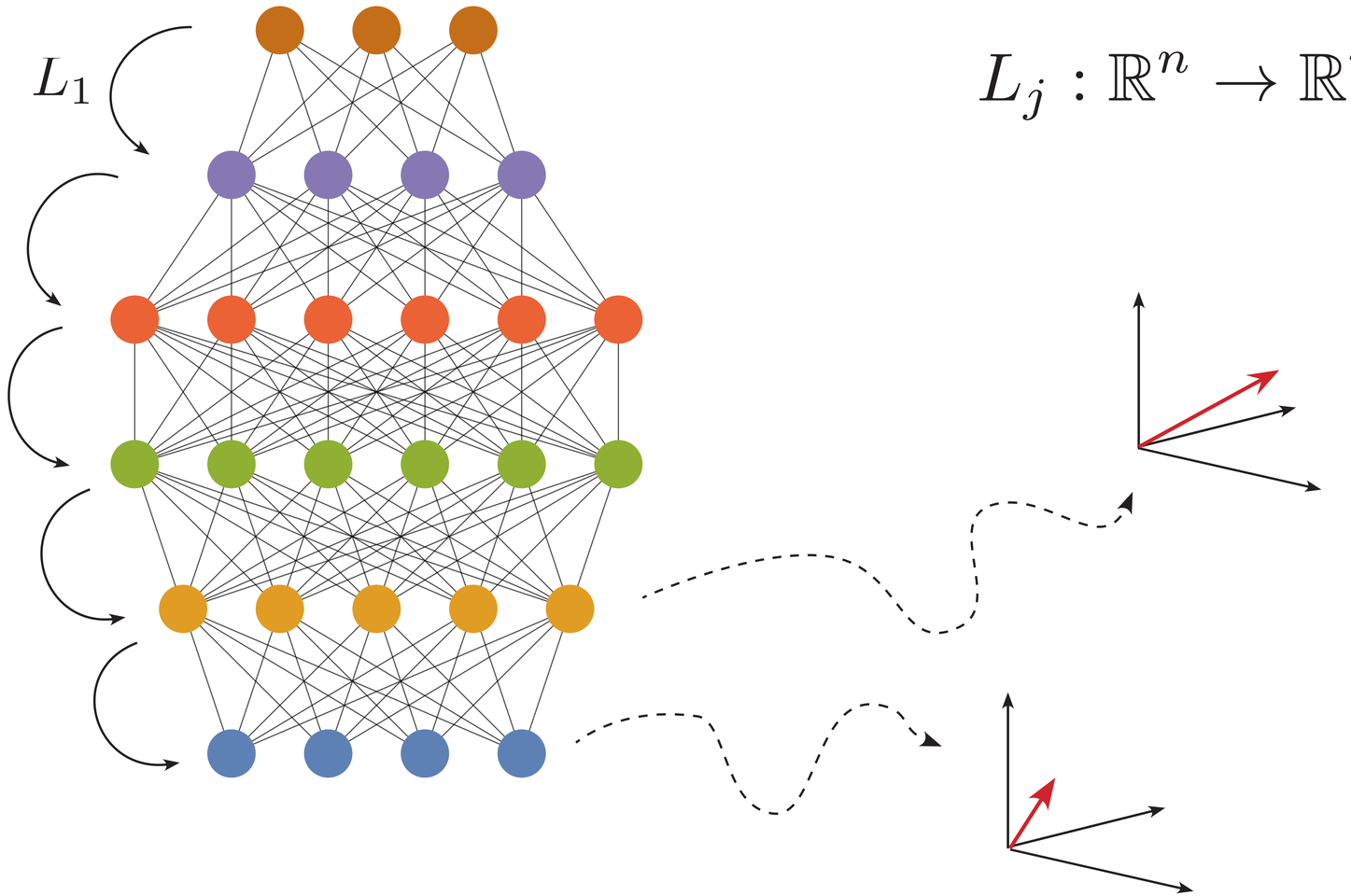
Self-supervised representation learning



$$L_j : \mathbb{R}^n \to \mathbb{R}^m$$



Self-supervised representation learning



Self-supervised representation learning

- Representation learning
  - Learn a task-independent representation of the data in the feature space of the neural network

$$\mathcal{N} : \mathbb{R}^n \to \mathbb{R}^{m_1} \to \cdots \to \mathbb{R}^{m_{j-1}} \to \mathbb{R}^m$$

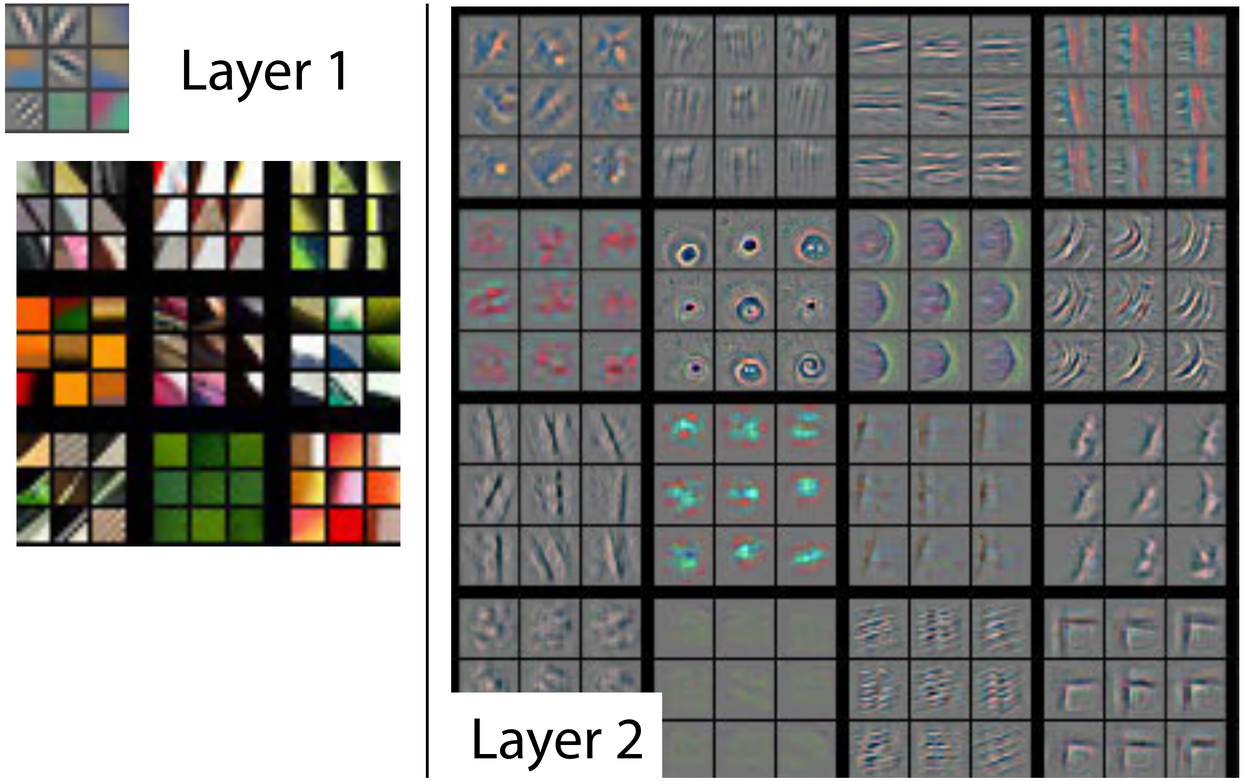
Self-supervised representation learning

- Representation learning
  - Learn a task-independent representation of the data in the feature space of the neural network

$$\mathcal{N} : \mathbb{R}^n \to \mathbb{R}^{m_1} \to \cdots \to \mathbb{R}^{m_{j-1}} \to \mathbb{R}^m$$

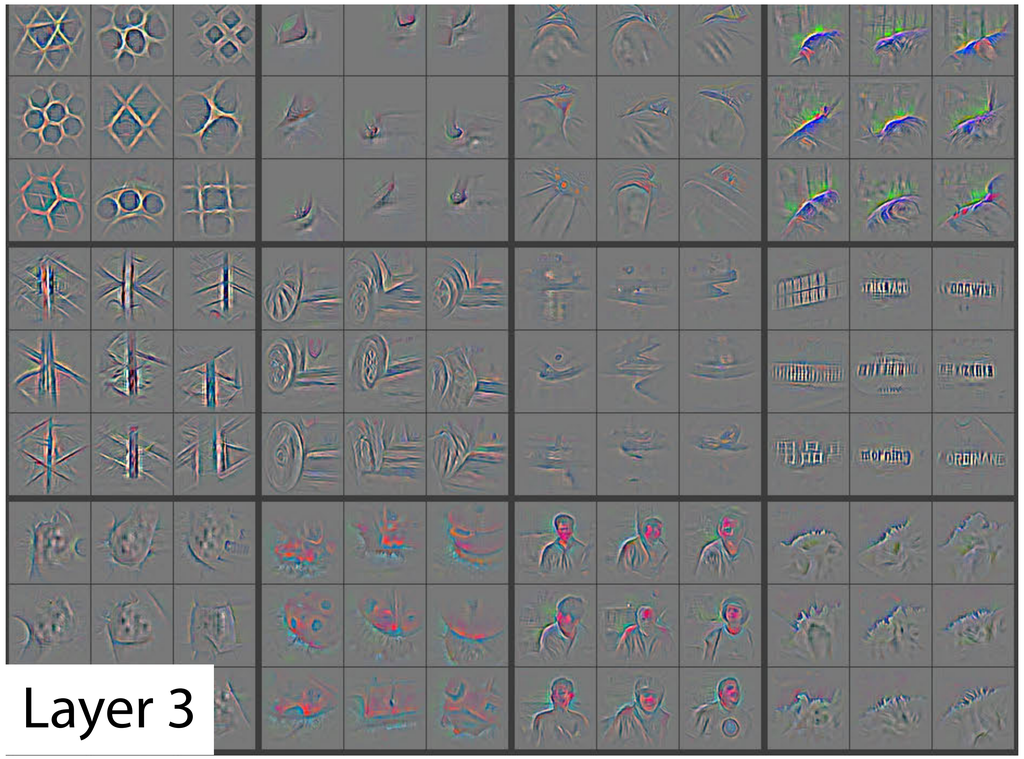
Effective “encoding” of the data useful for many applications

Self-supervised representation learning



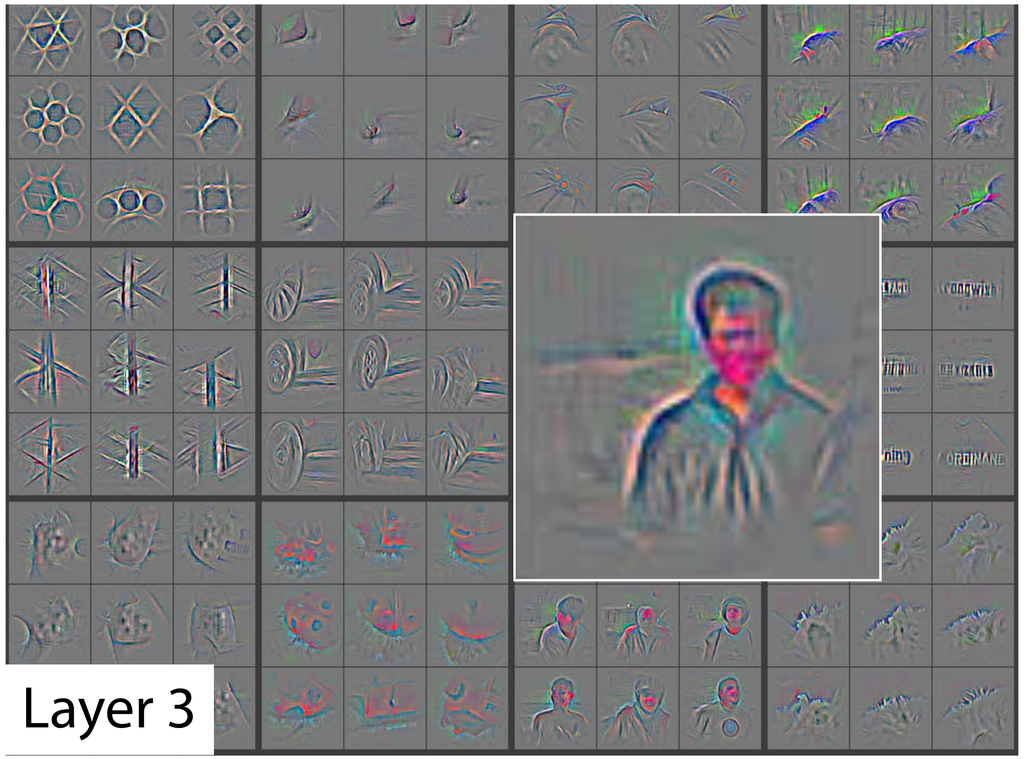
From M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, editors, Computer Vision – ECCV 2014, pages 818–833, Cham, 2014. Springer International Publishing.

Self-supervised representation learning



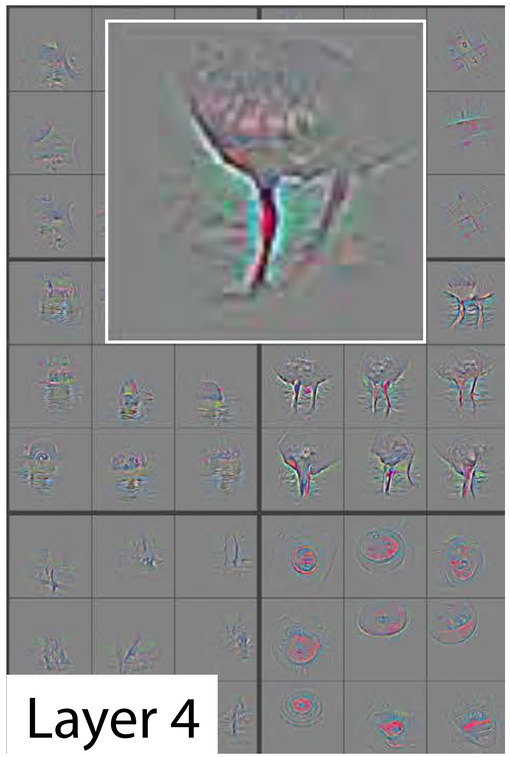
From M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, editors, Computer Vision – ECCV 2014, pages 818–833, Cham, 2014. Springer International Publishing.

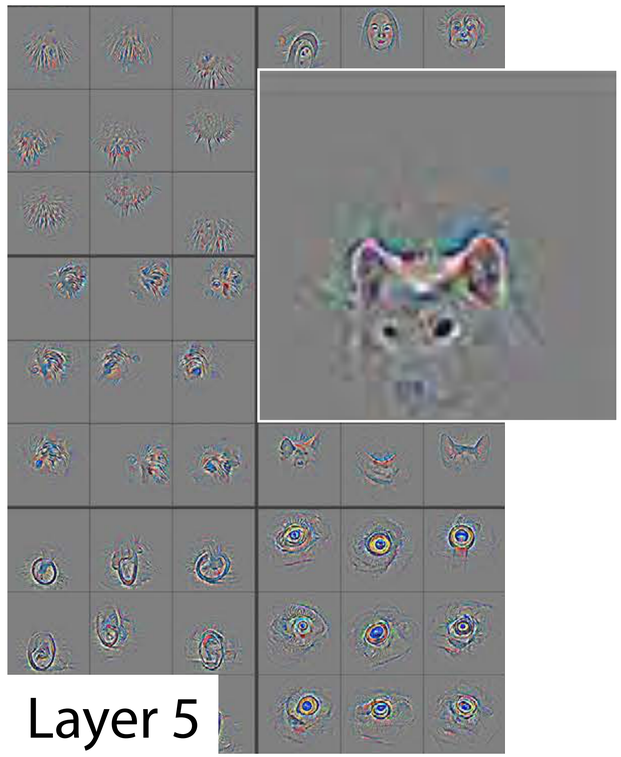
Self-supervised representation learning



From M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, editors, Computer Vision – ECCV 2014, pages 818–833, Cham, 2014. Springer International Publishing.

Self-supervised representation learning





From M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, editors, Computer Vision – ECCV 2014, pages 818–833, Cham, 2014. Springer International Publishing.

Self-supervised representation learning

- Representation learning
  - Learn a task-independent representation of the data in the *feature space* of the neural network

$$\mathcal{N} : \mathbb{R}^n \to \mathbb{R}^{m_1} \to \cdots \to \mathbb{R}^{m_{j-1}} \to \mathbb{R}^m$$

Effective “encoding” of the data useful for many applications

Self-supervised representation learning

- Representation learning
  - Learn a task-independent representation of the data in the *feature space* of the neural network

$$\mathcal{N} : \mathbb{R}^n \to \mathbb{R}^{m_1} \to \cdots \to \mathbb{R}^{m_{j-1}} \to \mathbb{R}^m$$

Effective “encoding” of the data useful for many applications

- Self-supervised training
  - Train with “labels” intrinsic to the data

Self-supervised representation learning

- Self-supervised pretext tasks:

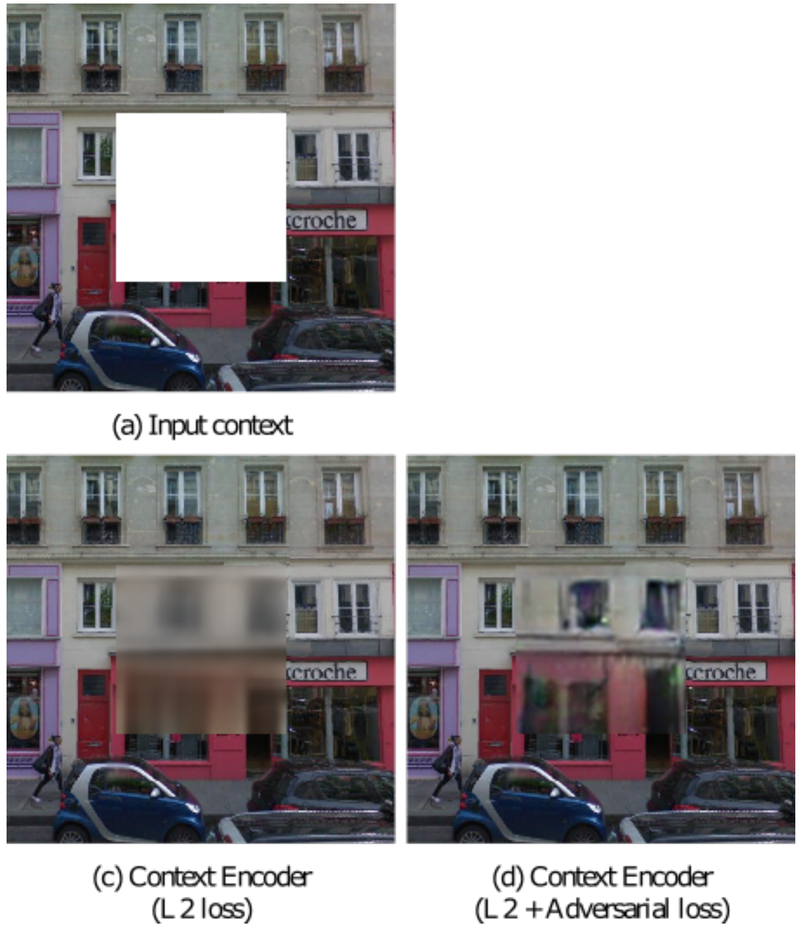
Self-supervised representation learning

- Self-supervised pretext tasks: inpainting of randomly deleted image parts[1]

[1] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros. Context encoders: Feature learning by inpainting. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2016.

Self-supervised representation learning

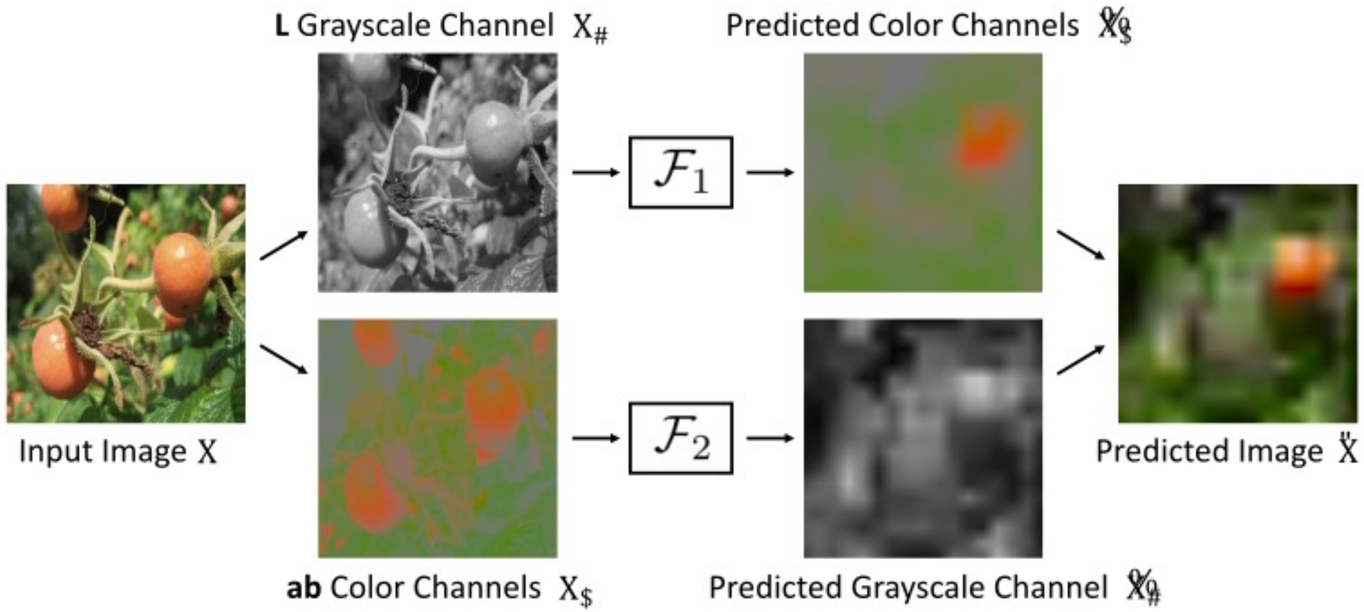
- Self-supervised pretext tasks: inpainting of randomly deleted image parts[1]



[1] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros. Context encoders: Feature learning by inpainting. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2016.

Self-supervised representation learning

- Self-supervised pretext tasks: predicting deleted color and gray scale channels[1]



[1] R. Zhang, P. Isola, and A. A. Efros. Split-brain autoencoders: Unsupervised learning by cross-channel prediction. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 2017.

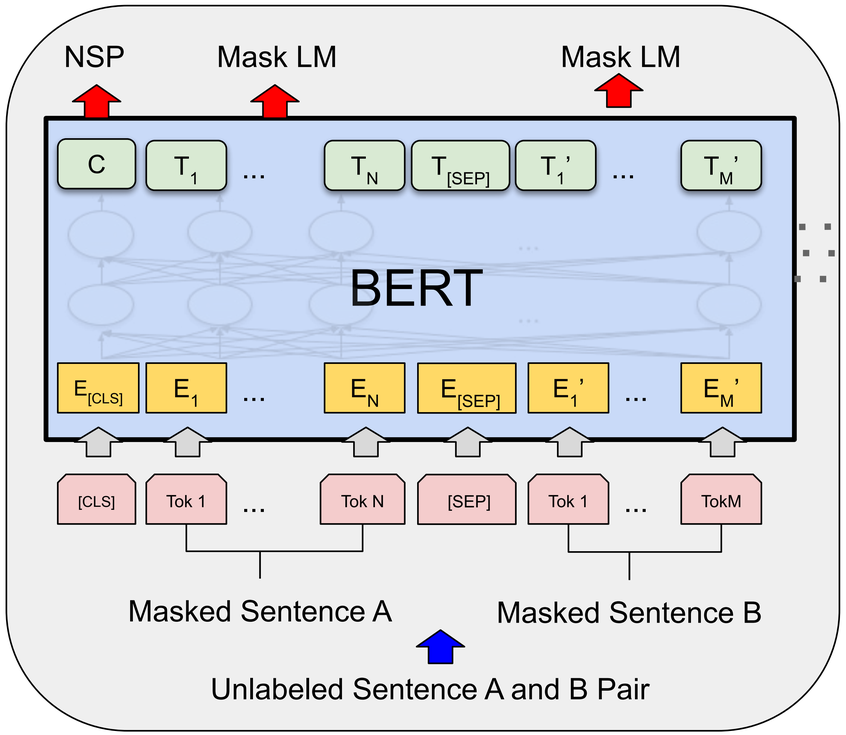
Self-supervised representation learning

- BERT masked language model[1]
  - Self-supervised representation learning for natural language processing (NLP)
  - Very large transformer neural network with billions of parameters
  - Self-supervised training essentially only feasible option at this scale

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

Self-supervised representation learning

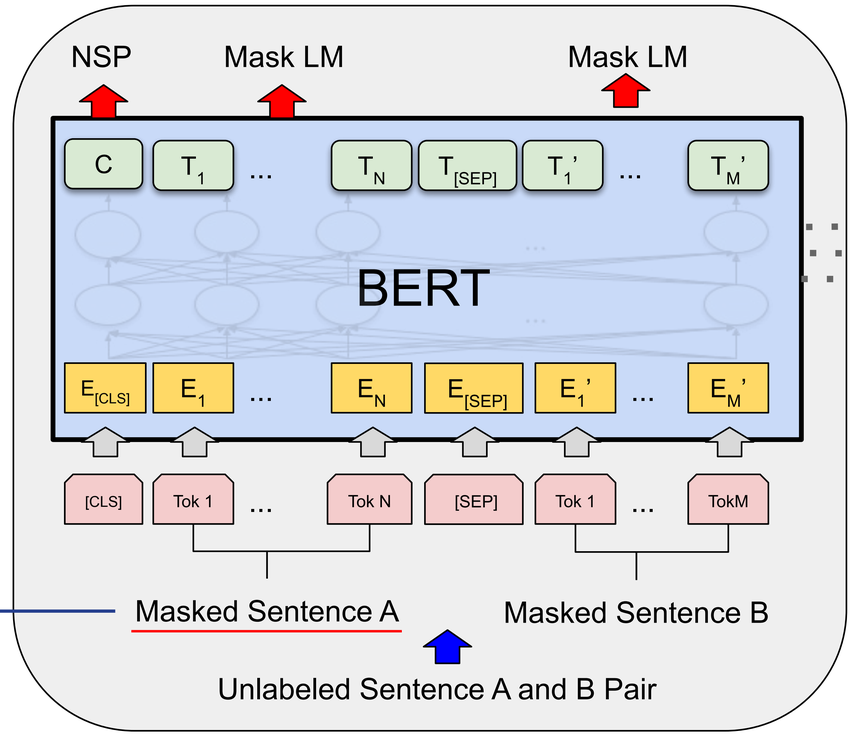
- BERT[1]



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

Self-supervised representation learning

- BERT[1]



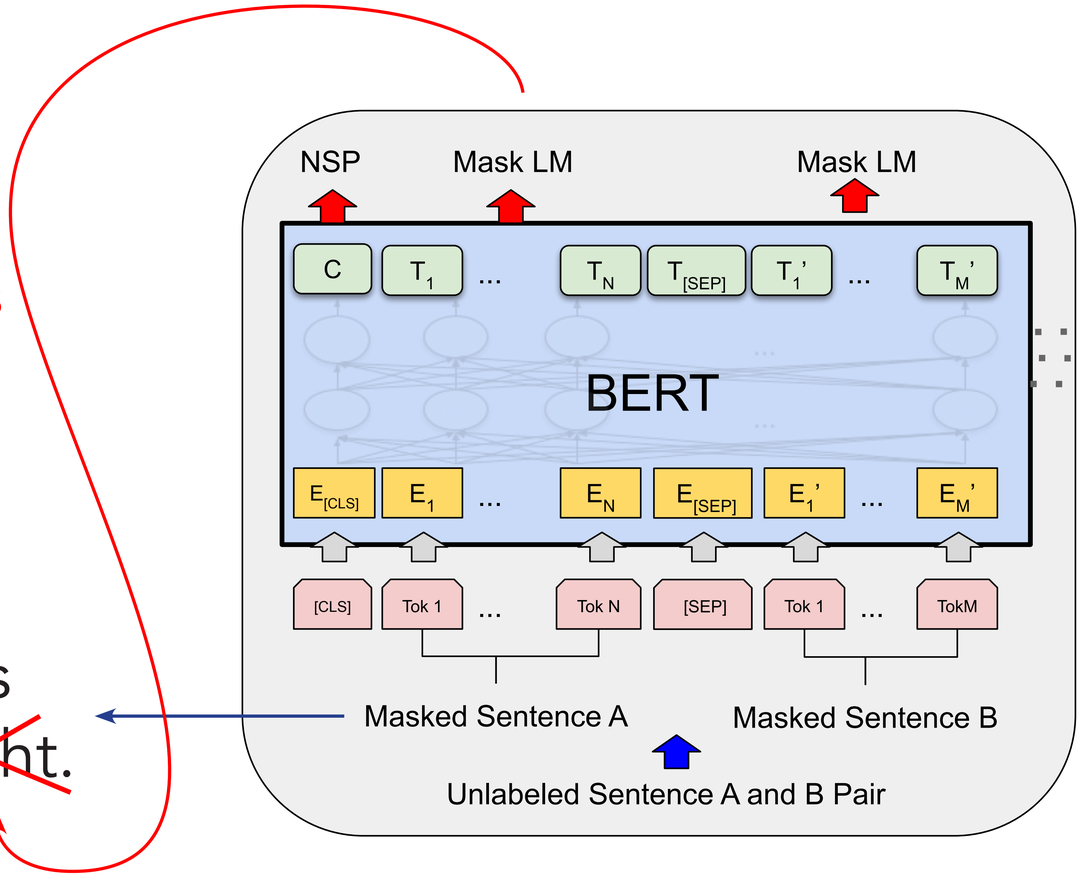
The sun was shining ~~bright~~.

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

Self-supervised representation learning

- BERT[1]

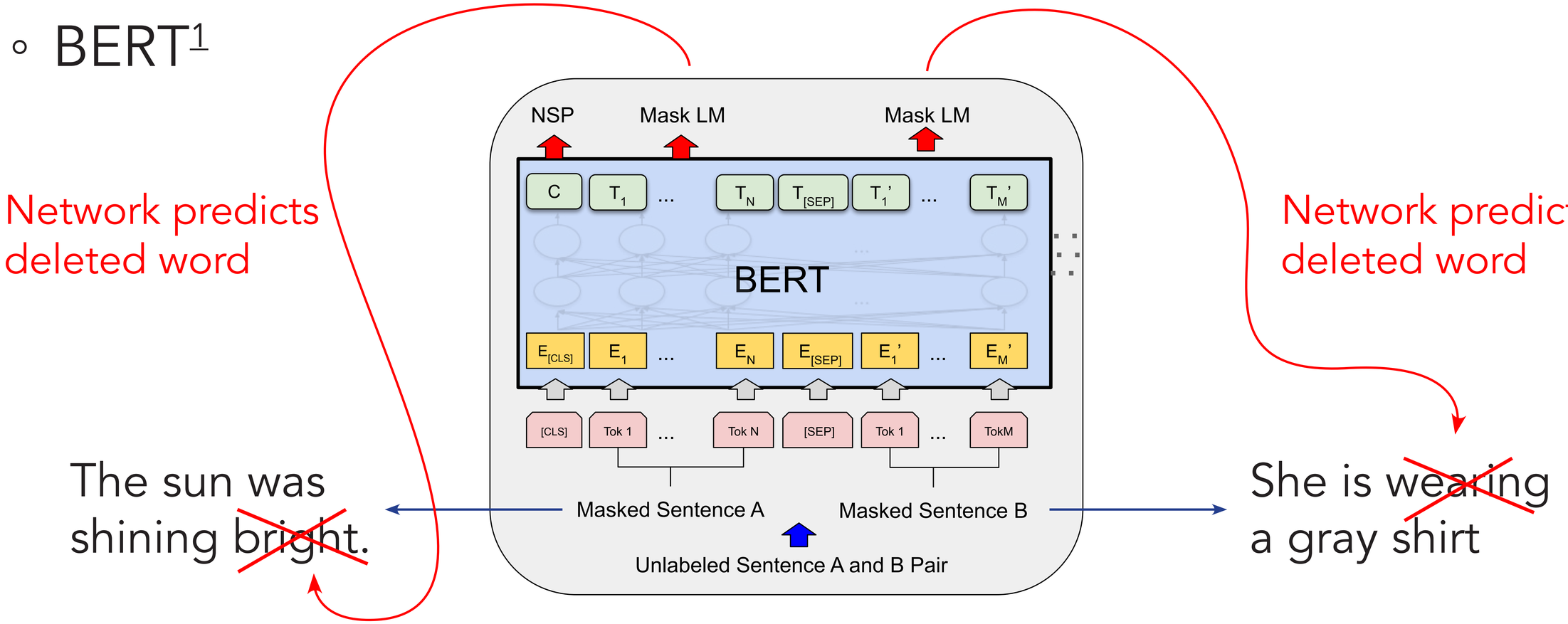
Network predicts deleted word



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

Self-supervised representation learning

- BERT[1]

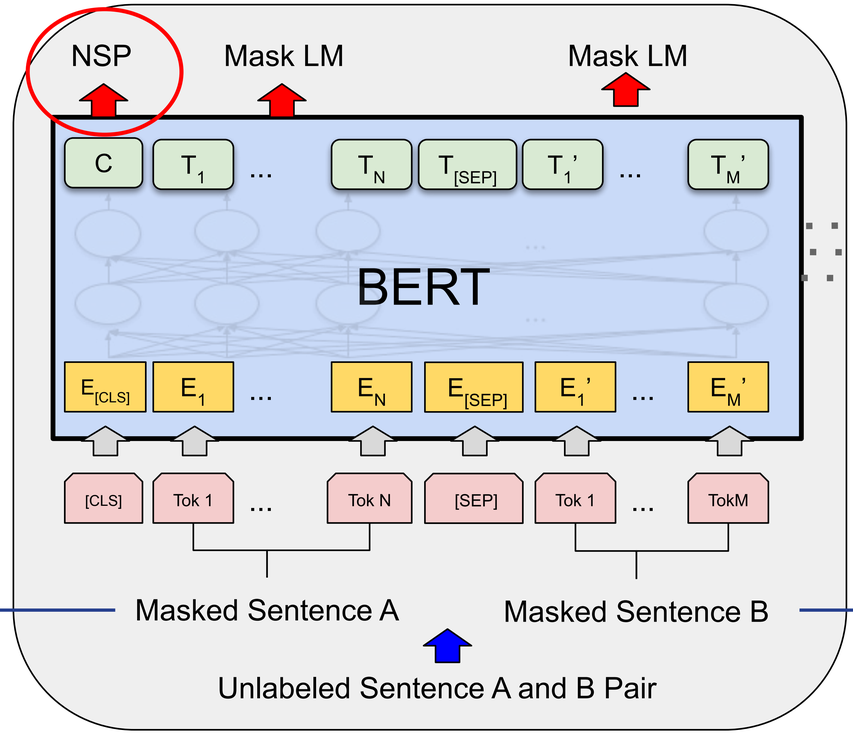


[1] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding, 2019.

Self-supervised representation learning

- BERT[1]

binary next sentence prediction



The sun was shining ~~bright.~~

She is ~~wearing~~ a gray shirt

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

Self-supervised representation learning

- BERT[1]

System	Dev		Test	
	EM	F1	EM	F1
Top Leaderboard Systems (Dec 10th, 2018)				
Human	-	-	82.3	91.2
#1 Ensemble - nlnet	-	-	86.0	91.7
#2 Ensemble - QANet	-	-	84.5	90.5
Published				
BiDAF+ELMo (Single)	-	85.6	-	85.8
R.M. Reader (Ensemble)	81.2	87.9	82.3	88.5
Ours				
$\text{BERT}_{\text{BASE}}$ (Single)	80.8	88.5	-	-
$\text{BERT}_{\text{LARGE}}$ (Single)	84.1	90.9	-	-
$\text{BERT}_{\text{LARGE}}$ (Ensemble)	85.8	91.8	-	-
$\text{BERT}_{\text{LARGE}}$ (Sgl.+TriviaQA)	**84.2**	**91.1**	**85.1**	**91.8**
$\text{BERT}_{\text{LARGE}}$ (Ens.+TriviaQA)	**86.2**	**92.2**	**87.4**	**93.2**

Table 2: SQuAD 1.1 results. The BERT ensemble is 7x systems which use different pre-training checkpoints and fine-tuning seeds.

Performance of fine-tuned model on question-answer benchmark

Self-supervised representation learning

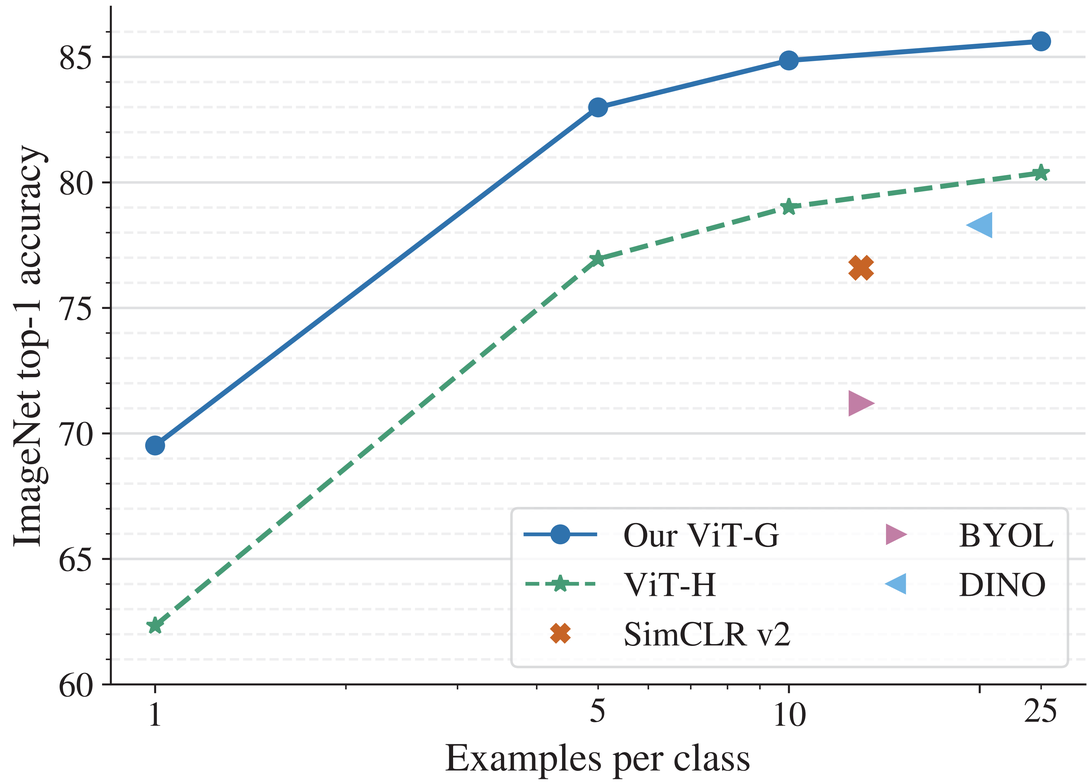
- Many other recent approaches:
  - Contrastive loss functions[1]
  - DINO[2]
  - Extensions of BERT-style completion tasks[3]
  - ...

[1] P. H. Le-Khac, G. Healy, and A. F. Smeaton. Contrastive representation learning: A framework and review. IEEE Access, 8:193907–193934, 2020.

[2] 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.

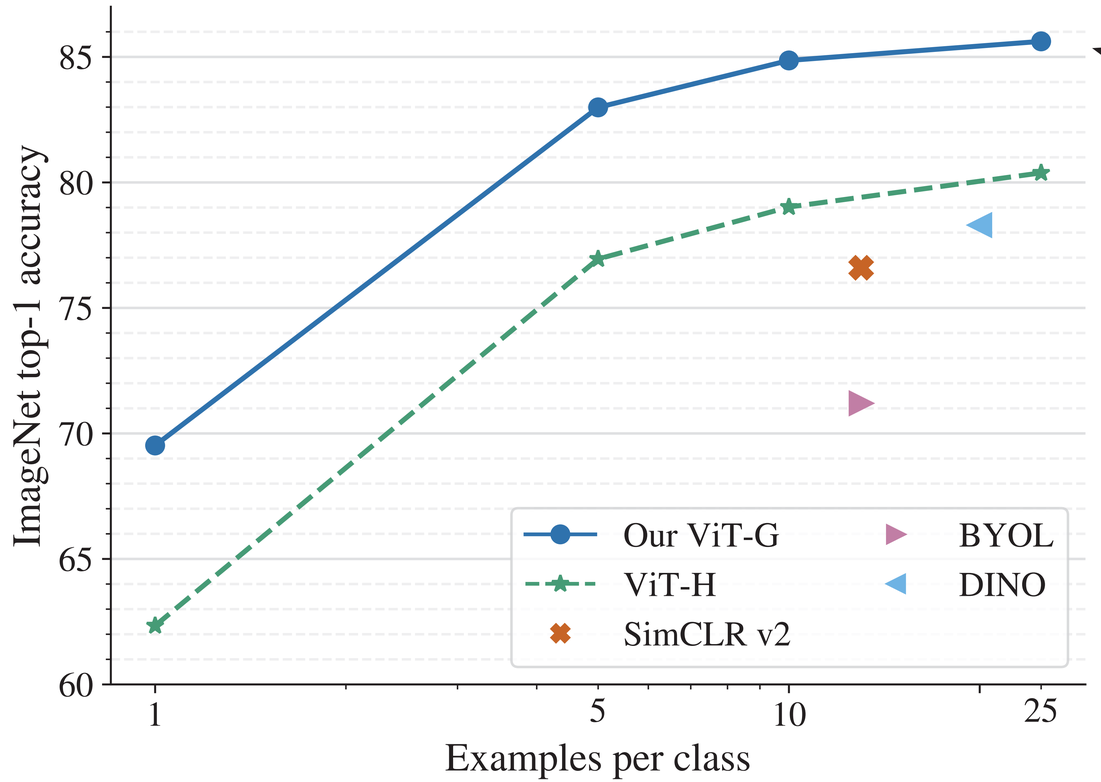
[3] H. Bao, L. Dong, S. Piao, and F. Wei. BEit: BERT pre-training of image transformers. In International Conference on Learning Representations, 2022.

Self-supervised representation learning



From X. Zhai, A. Kolesnikov, N. Houlsby, and L. Beyer. Scaling vision transformers, 2021.

Self-supervised representation learning



Scaling data and model size leads to continuously improving results

From X. Zhai, A. Kolesnikov, N. Houlsby, and L. Beyer. Scaling vision transformers, 2021.

Transformers

Transformer neural networks

- Standard architecture in natural language processing but largely also in computer vision

Transformer neural networks

- Standard architecture in natural language processing but largely also in computer vision
- Scale well to highly parallel training and billions of parameters, especially for sequential data

Transformer neural networks

	Ours-JFT (ViT-H/14)	Ours-JFT (ViT-L/16)	Ours-I21k (ViT-L/16)	BiT-L (ResNet152x4)	Noisy Student (EfficientNet-L2)
ImageNet	**88.55** ± 0.04	87.76 ± 0.03	85.30 ± 0.02	87.54 ± 0.02	88.4/88.5*
ImageNet ReaL	**90.72** ± 0.05	90.54 ± 0.03	88.62 ± 0.05	90.54	90.55
CIFAR-10	**99.50** ± 0.06	99.42 ± 0.03	99.15 ± 0.03	99.37 ± 0.06	–
CIFAR-100	**94.55** ± 0.04	93.90 ± 0.05	93.25 ± 0.05	93.51 ± 0.08	–
Oxford-IIIT Pets	**97.56** ± 0.03	97.32 ± 0.11	94.67 ± 0.15	96.62 ± 0.23	–
Oxford Flowers-102	99.68 ± 0.02	**99.74** ± 0.00	99.61 ± 0.02	99.63 ± 0.03	–
VTAB (19 tasks)	**77.63** ± 0.23	76.28 ± 0.46	72.72 ± 0.21	76.29 ± 1.70	–
TPUv3-core-days	2.5k	0.68k	0.23k	9.9k	12.3k

very large transformer

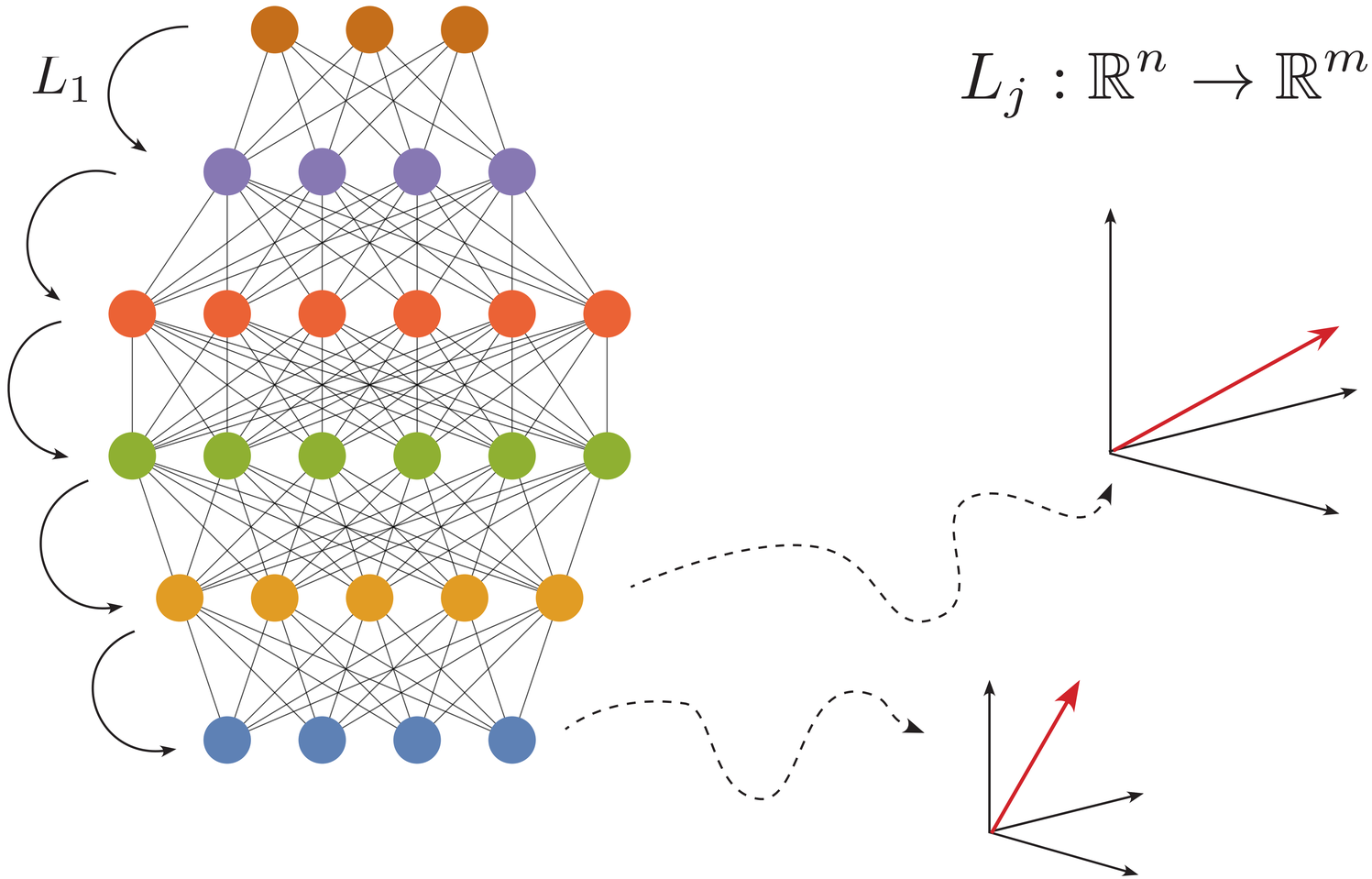
very large convolutional net

From A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In International Conference on Learning Representations, 2021.

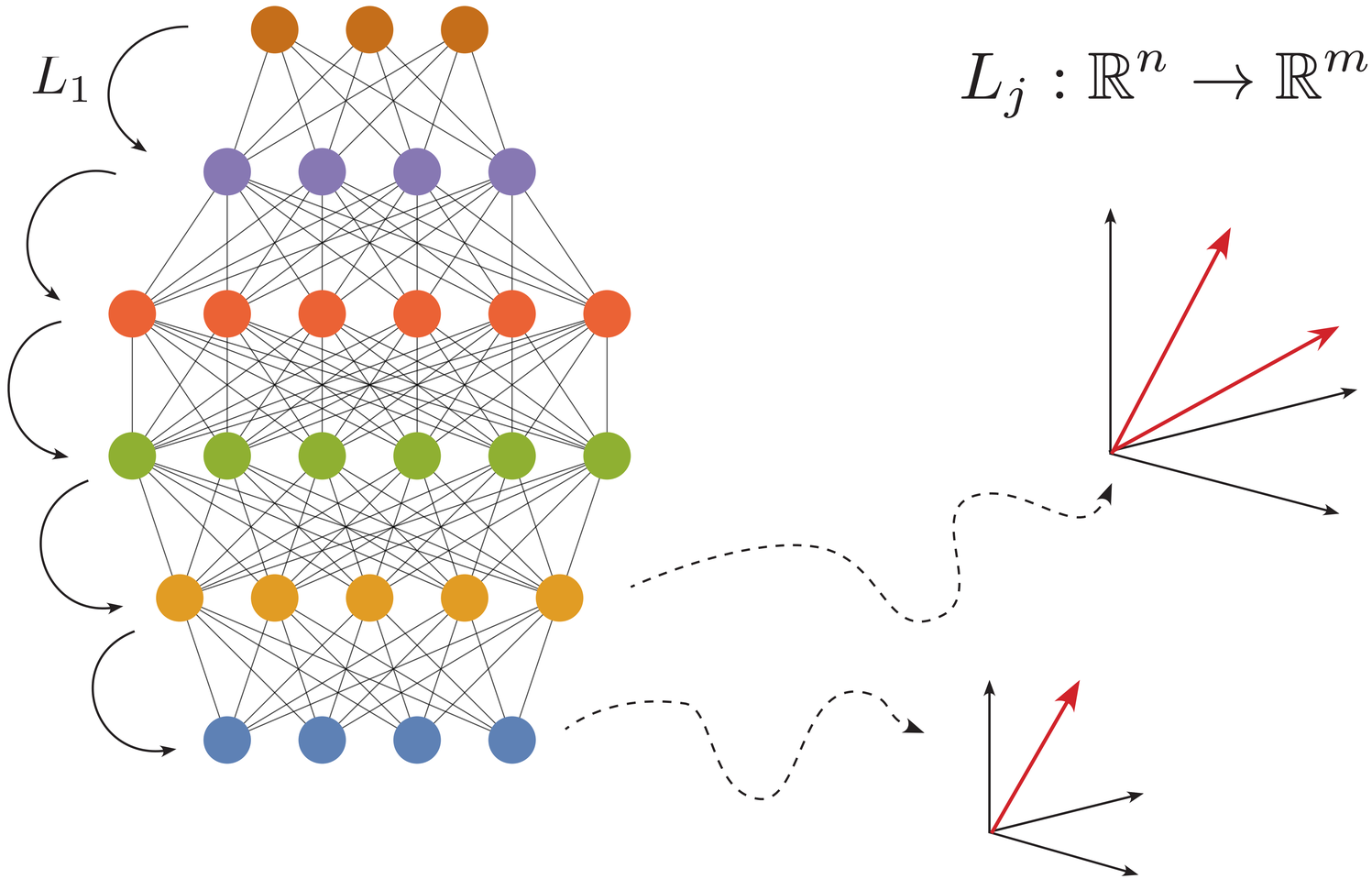
Transformer neural networks

- Standard architecture in natural language processing but largely also in computer vision
- Scale well to highly parallel training and billions of parameters, especially for sequential data

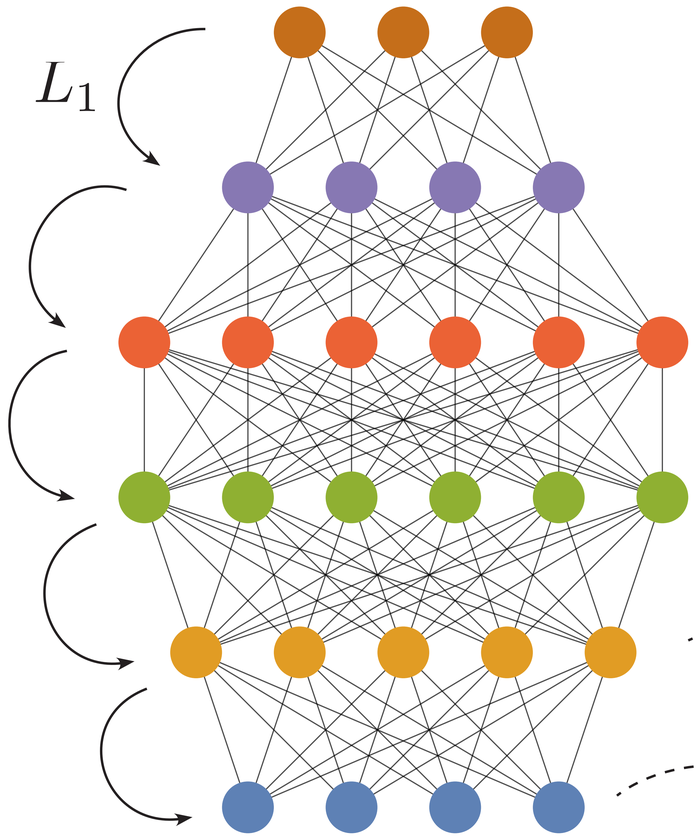
Transformer neural networks



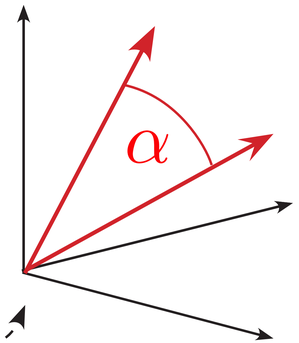
Transformer neural networks

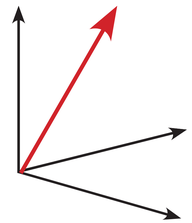


Transformer neural networks



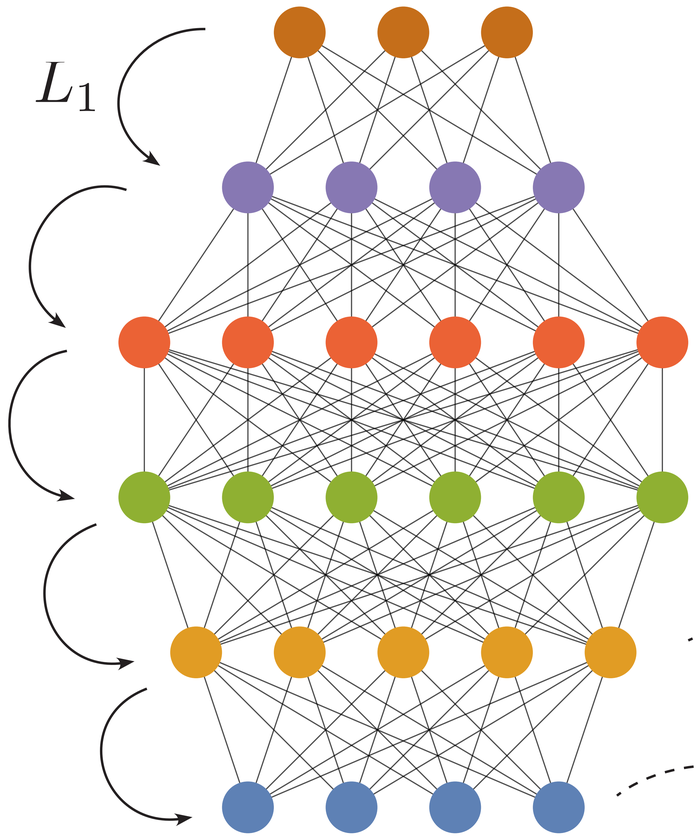
$$L_j : \mathbb{R}^n \to \mathbb{R}^m$$



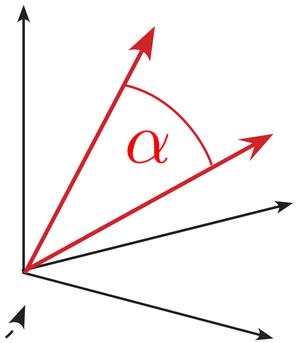


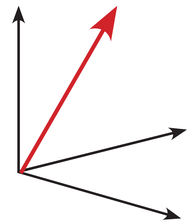
correlation between inputs in feature space

Transformer neural networks



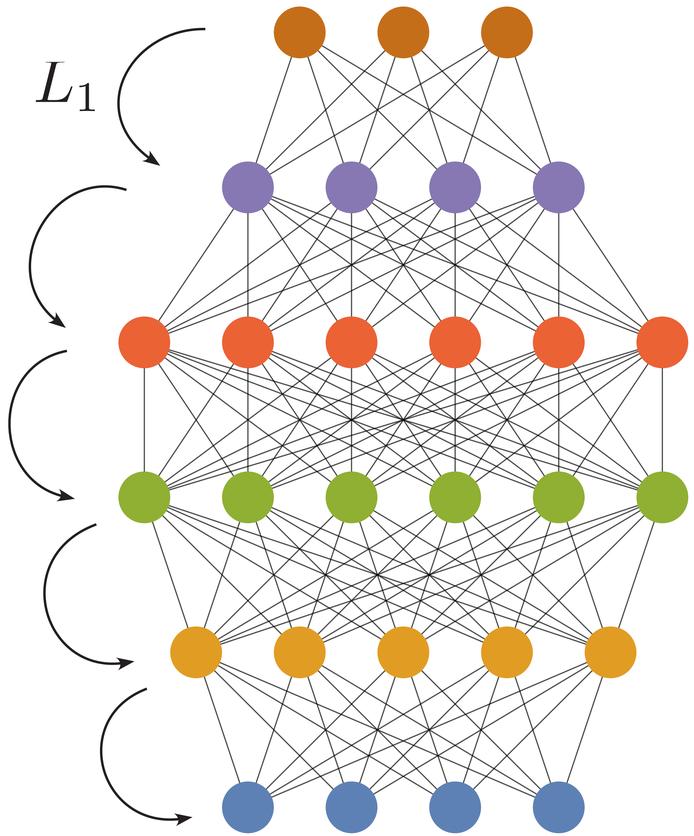
$$L_j : \mathbb{R}^n \to \mathbb{R}^m$$





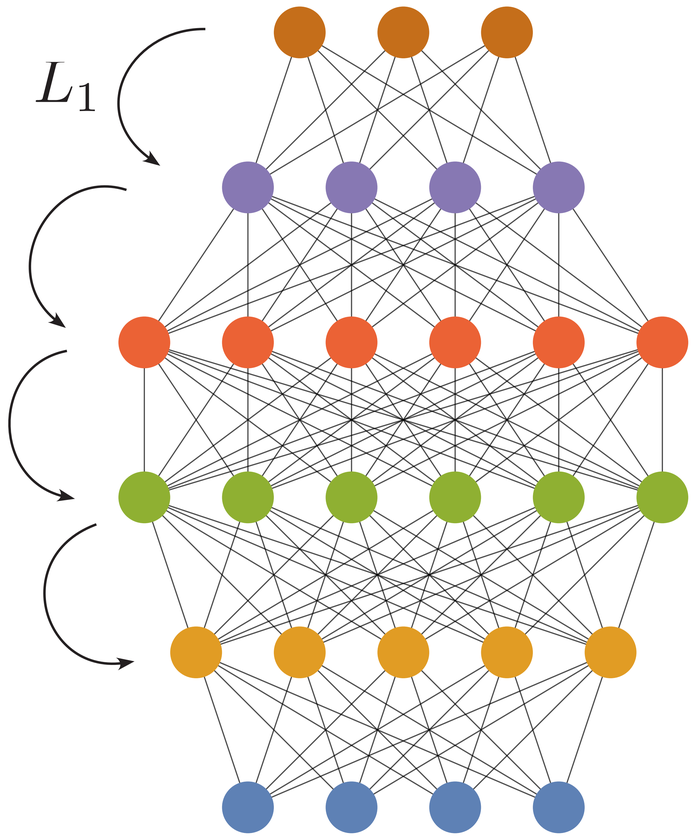
correlation between inputs in feature space: attention

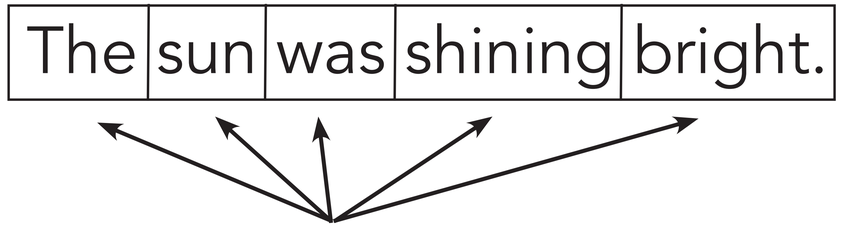
Transformer neural networks



The sun was shining bright.

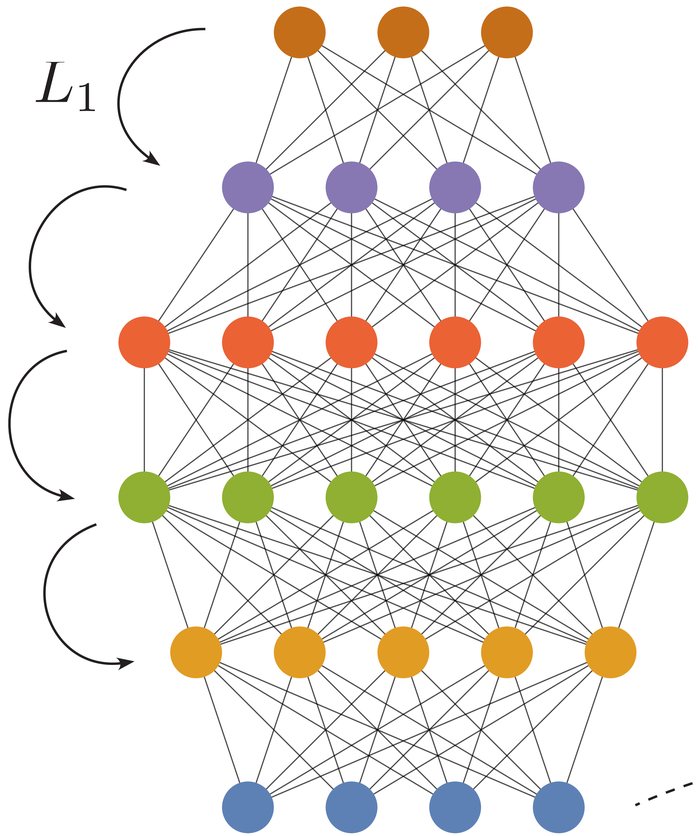
Transformer neural networks



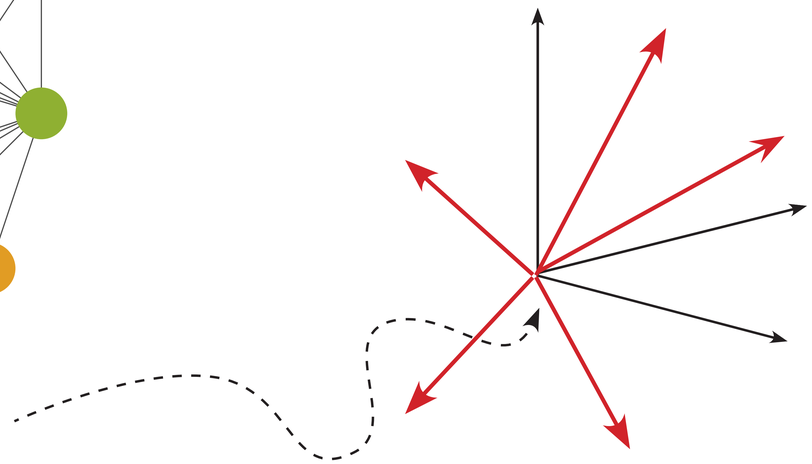


independent inputs to the network: token

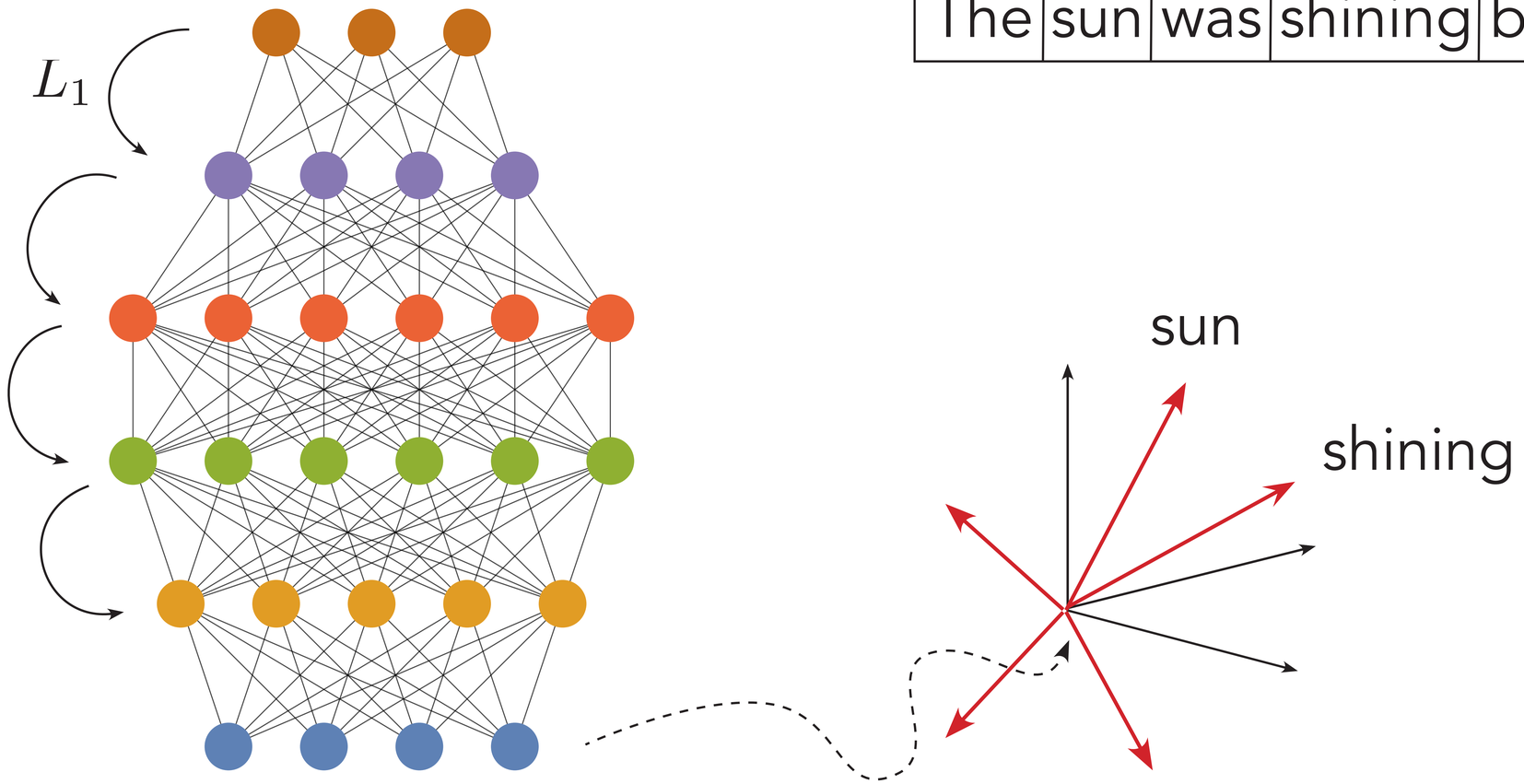
Transformer neural networks



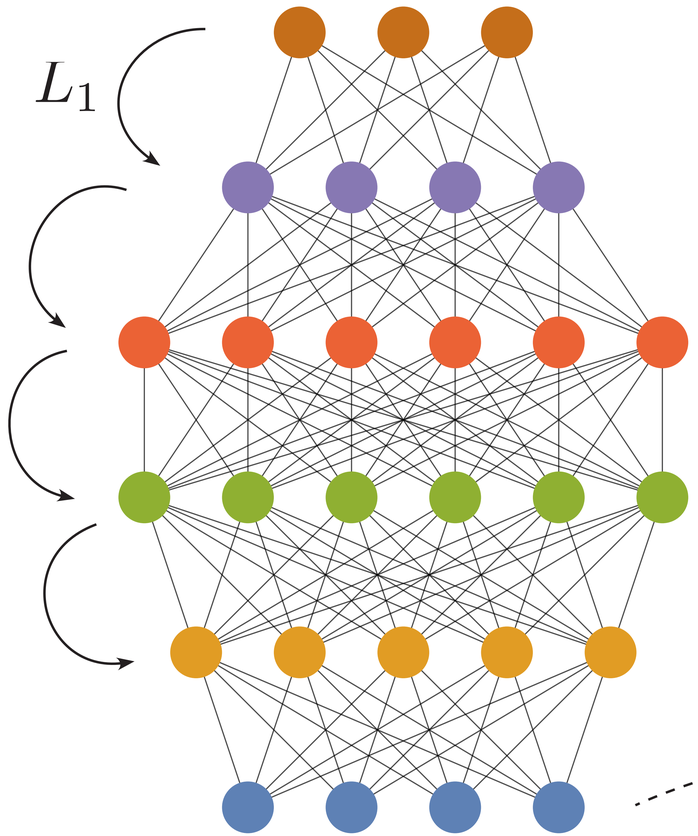
The	sun	was	shining	bright.



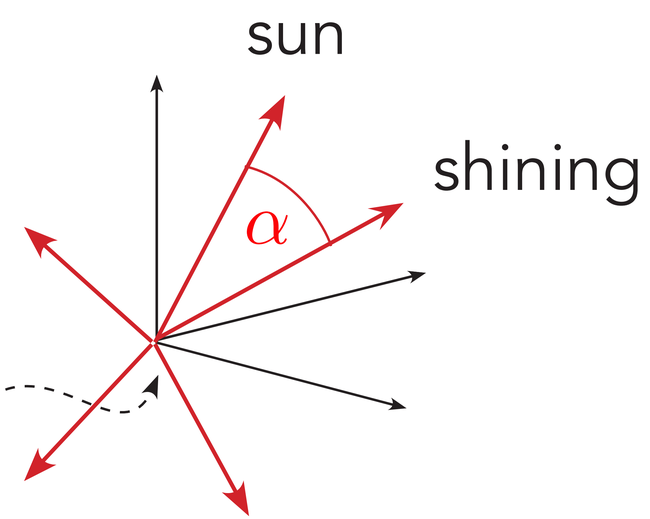
Transformer neural networks



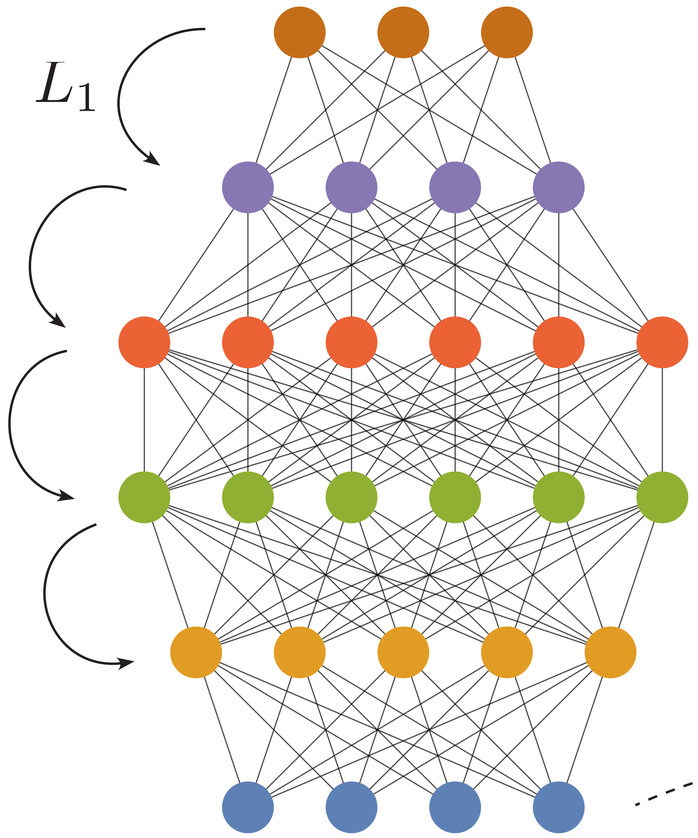
Transformer neural networks



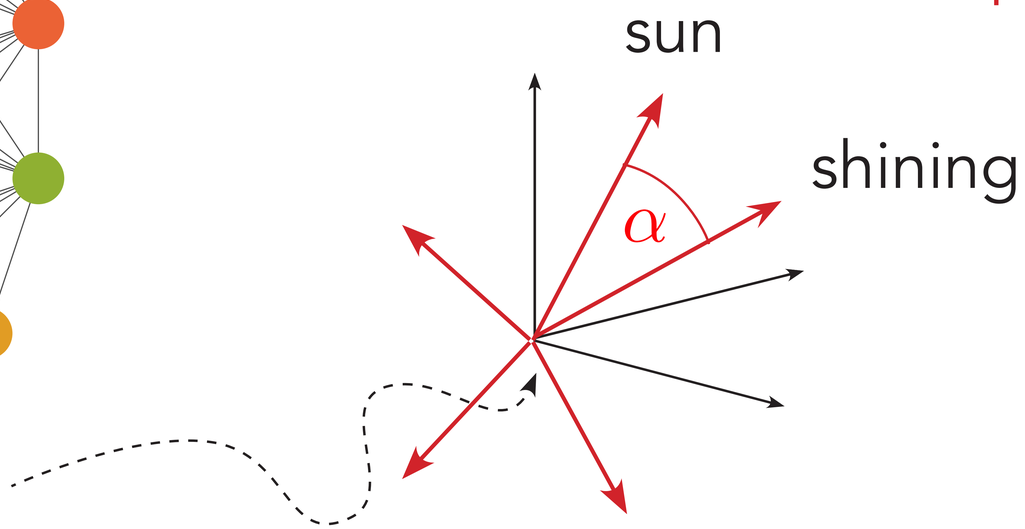
The	sun	was	shining	bright.



Transformer neural networks



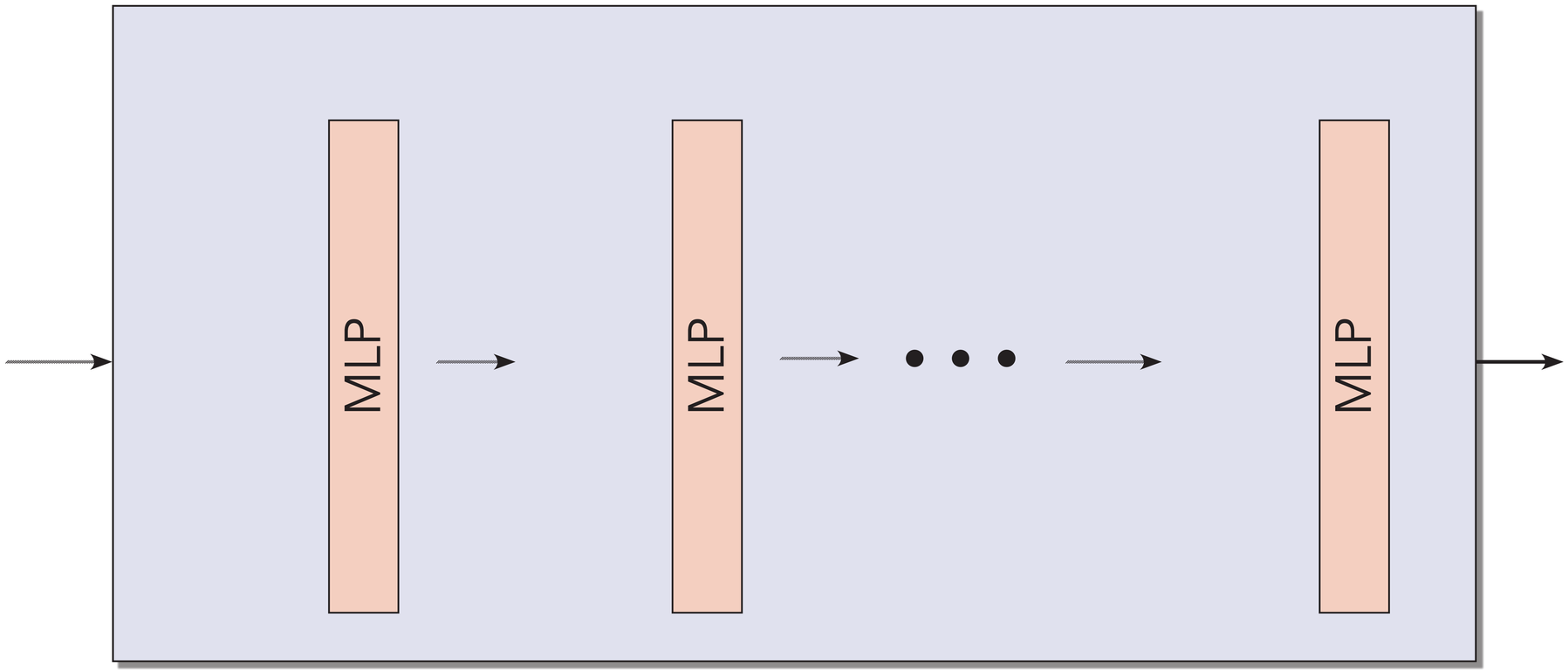
The	sun	was	shining	bright.



correlation between inputs in feature space: attention

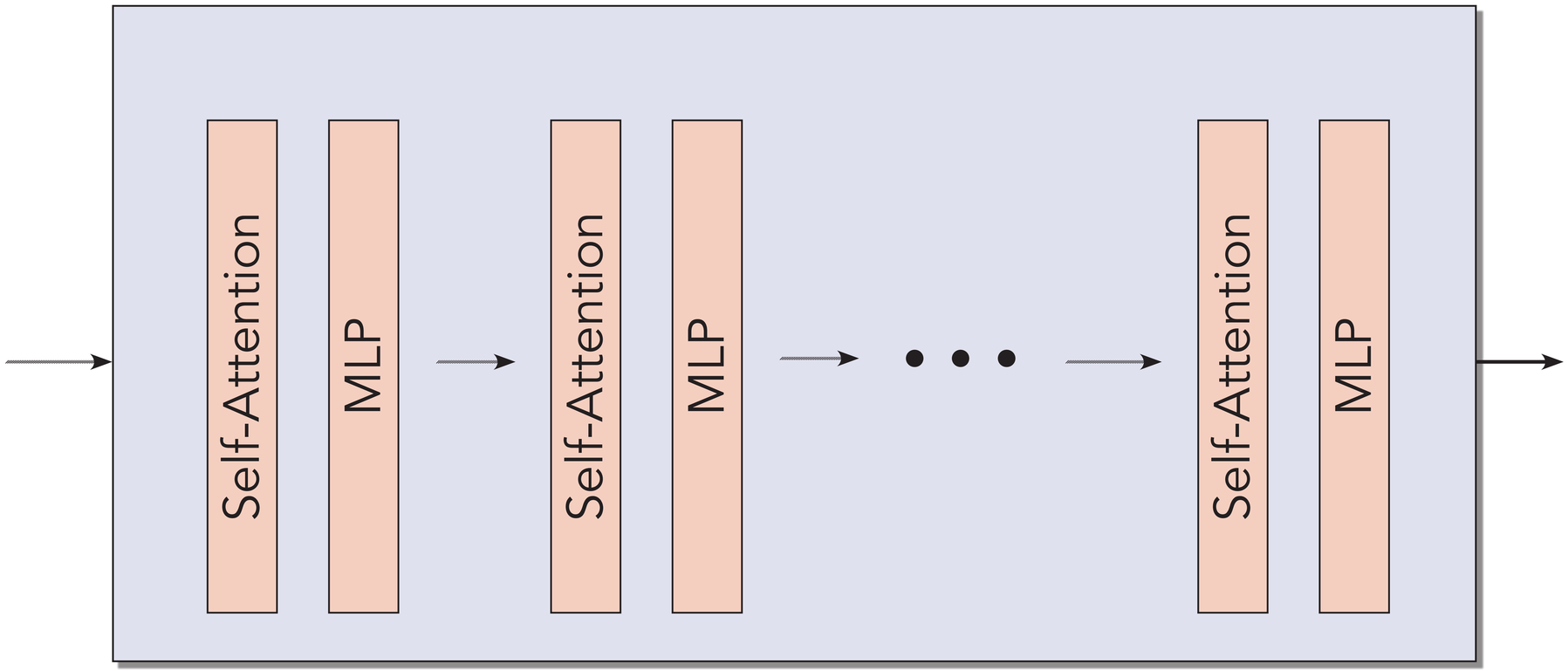
Transformer neural networks

Transformer neural networks



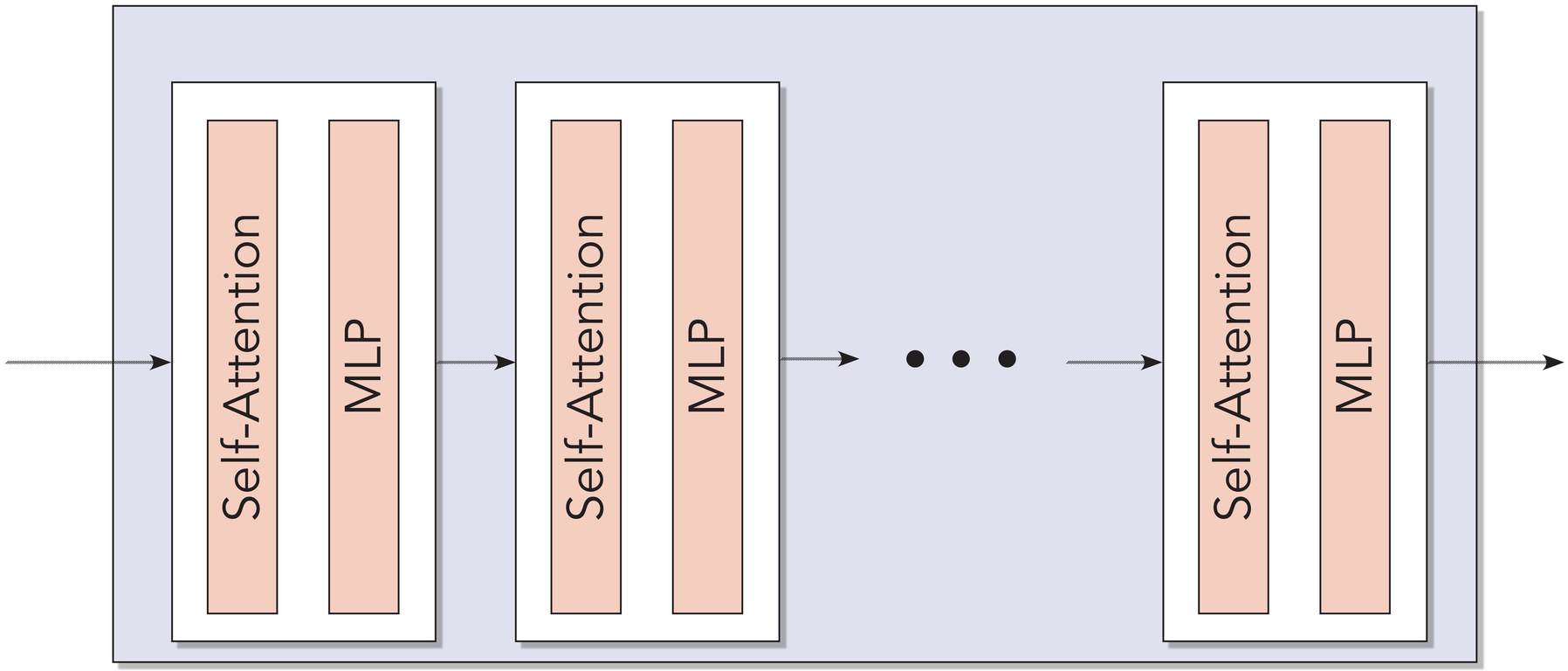
MLP: map between feature spaces

Transformer neural networks



Self attention: compute correlation between

Transformer neural networks



Transformer neural networks: self attention

- Project all tokens in k-th feature space onto i-th token

$$t_i \cdot t_j \quad \forall j$$

Transformer neural networks: self attention

- Project all tokens in k-th feature space onto i-th token

$$t_i \cdot t_j \quad \forall j$$

- Use correlation/projection for weighted average (analogous to basis representation)

$$\tilde{t}_i = \sum_j \sigma(t_i \cdot t_j)\, t_j$$

Transformer neural networks: self attention

- Project all tokens in k-th feature space onto i-th token

$$t_i \cdot t_j \quad \forall j$$

- Use correlation/projection for weighted average (analogous to basis representation)

$$\tilde{t}_i = \sum_j \sigma(t_i \cdot t_j)\, t_j$$

softmax, i.e. soft argmax

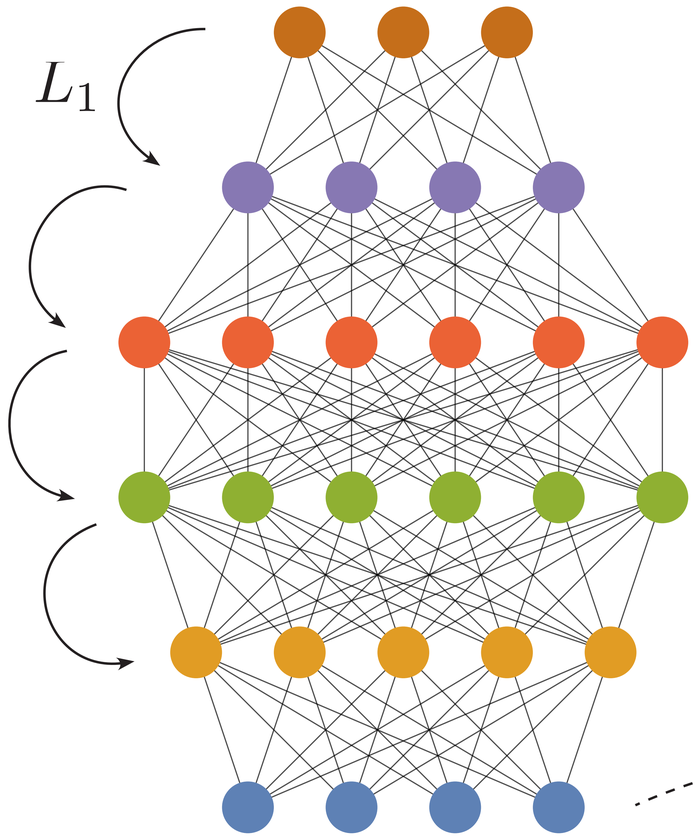
Transformer neural networks: self attention

- Use correlation/projection for weighted average (analogous to basis representation)

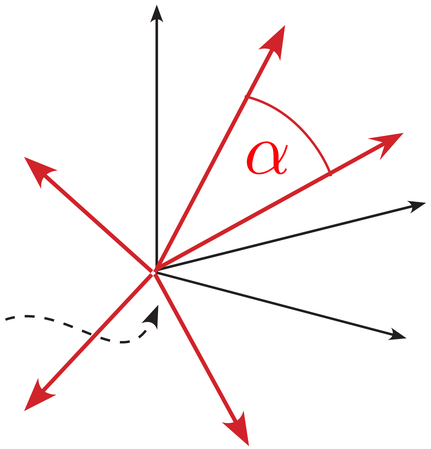
$$\tilde{t}_i = \sum_j \sigma(t_i \cdot t_j)\, t_j$$

- Different heads for different feature spaces at each level

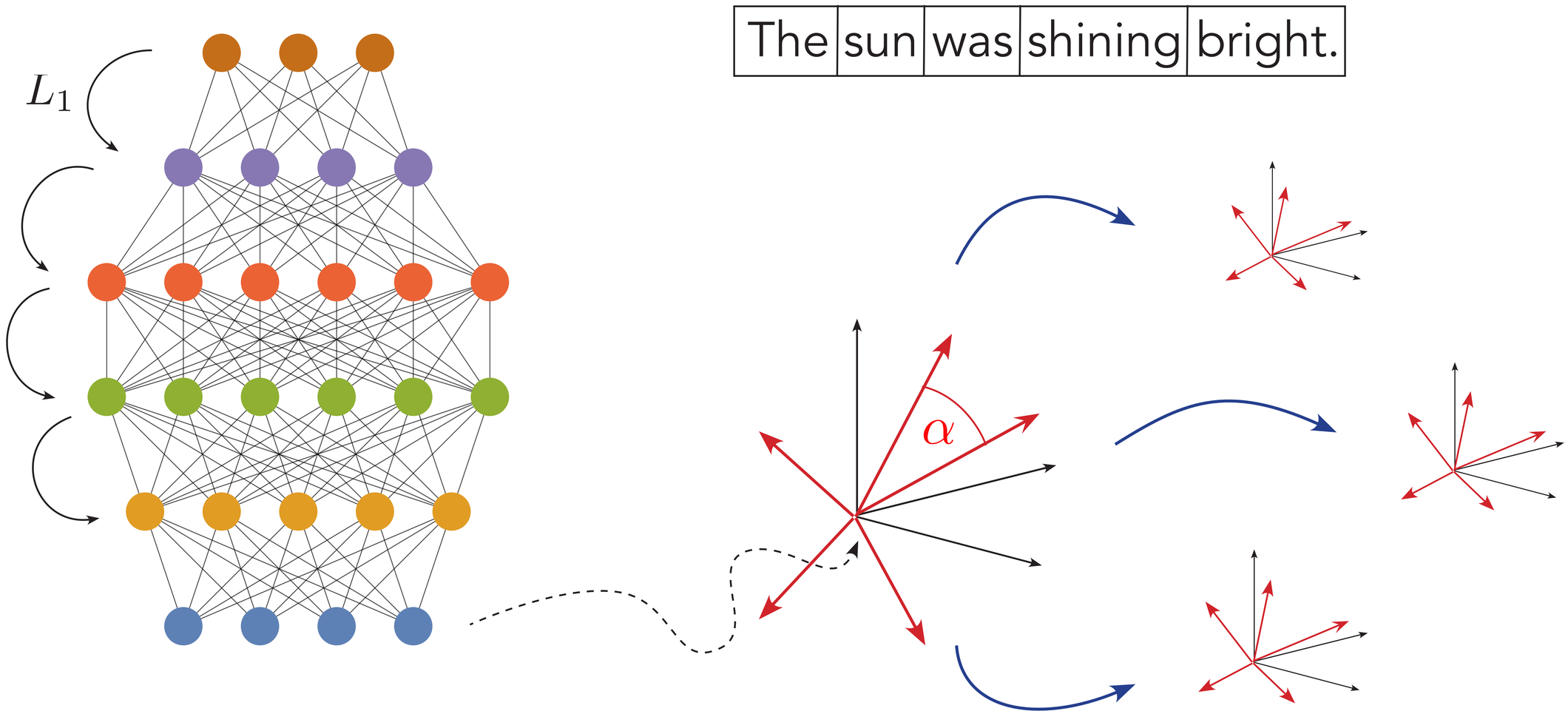
Transformer neural networks



The	sun	was	shining	bright.



Transformer neural networks



Transformer neural networks: self attention

- Use correlation/projection for weighted average (analogous to basis representation)

$$\tilde{t}_i = \sum_j \sigma(t_i \cdot t_j)\, t_j$$

- Different heads for different feature spaces at each level

$$t_i^{h,*} = W_{\{q,k,v\}}^h t_i$$

Transformer neural networks: self attention

- Different heads for different feature spaces at each level

$$t_i^{h,*} = W_{\{q,k,v\}}^h t_i$$

- Learnable inner product

$$t_i^{h,q} \cdot t_j^{h,k} = t_i^T W_{h,q}^T W_{h,k} t_j$$

Transformer neural networks: self attention

- Different heads for different feature spaces at each level

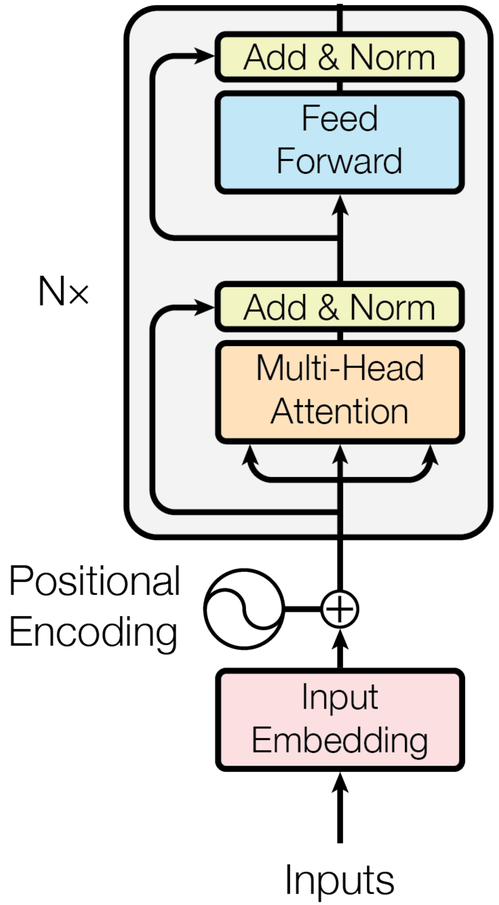
$$t_i^{h,*} = W_{\{q,k,v\}}^h t_i$$

- Learnable inner product

$$t_i^{h,q} \cdot t_j^{h,k} = t_i^T W_{h,q}^T W_{h,k} t_j$$

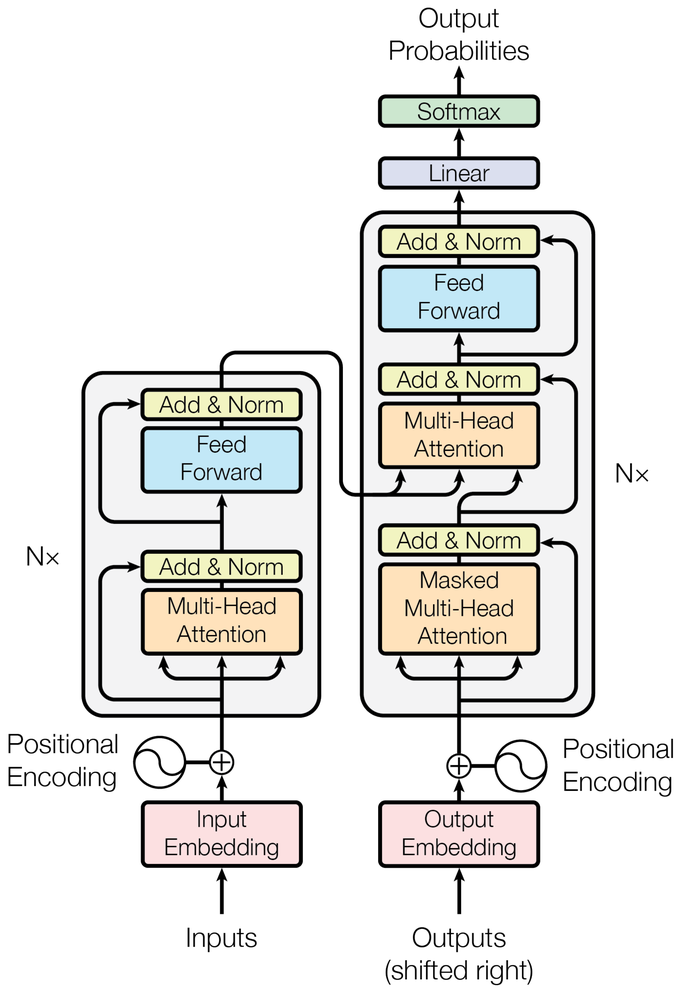
$$= t_i^T G_h t_j$$

Transformer neural networks



A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. *Attention is all you need*. In NEURIPS 2017, 2017.

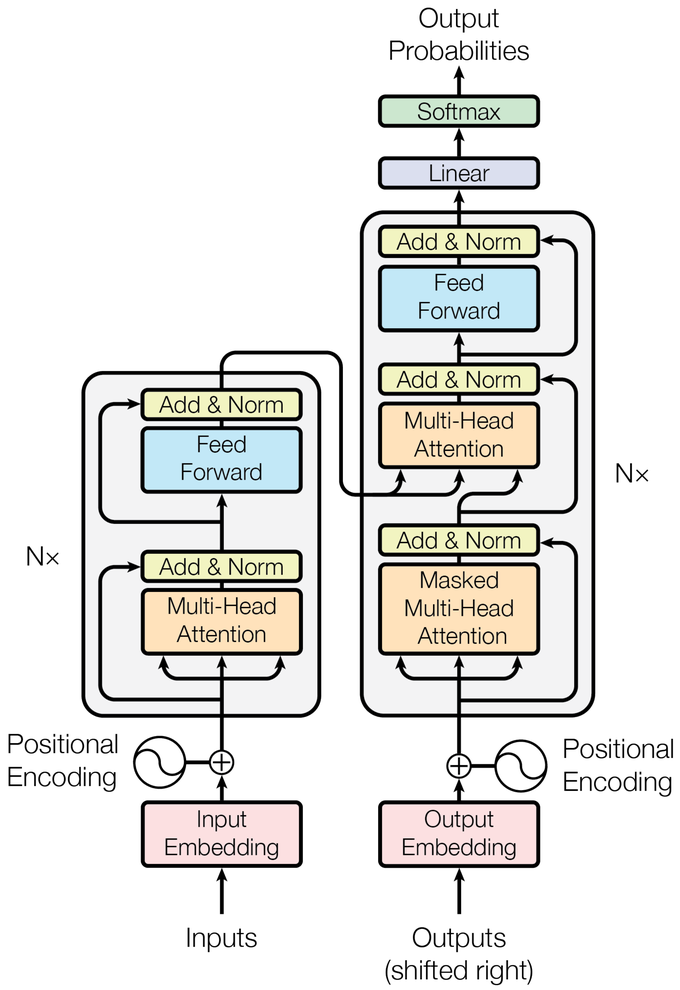
Transformer neural networks



A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. *Attention is all you need*. In NEURIPS 2017, 2017.

Transformer neural networks

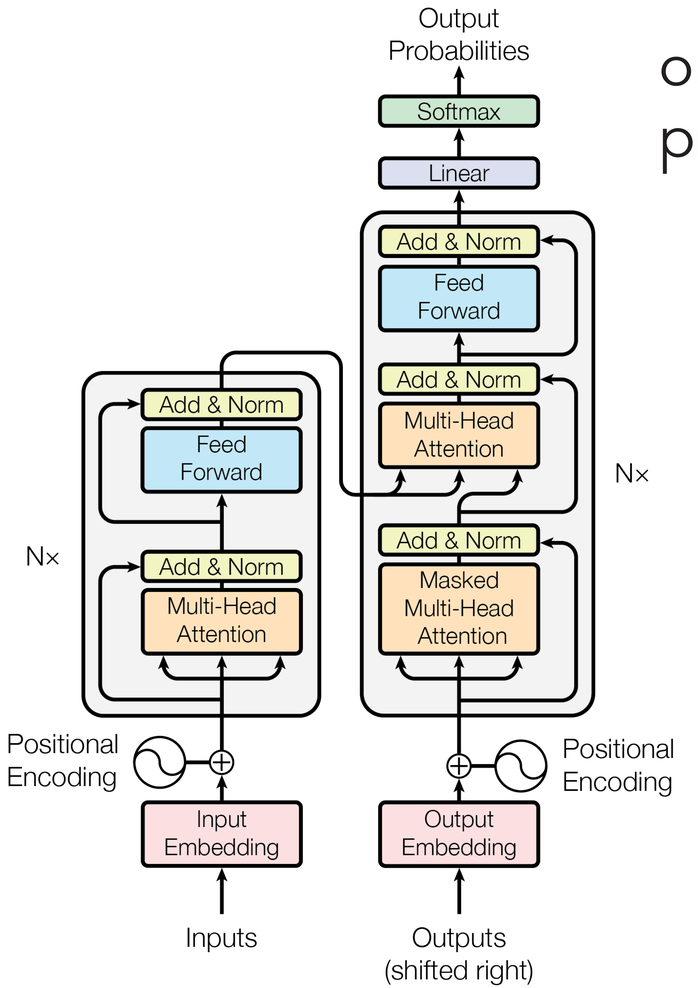
encoder



decoder

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. *Attention is all you need*. In NEURIPS 2017, 2017.

Transformer neural networks



output stream of predicted words

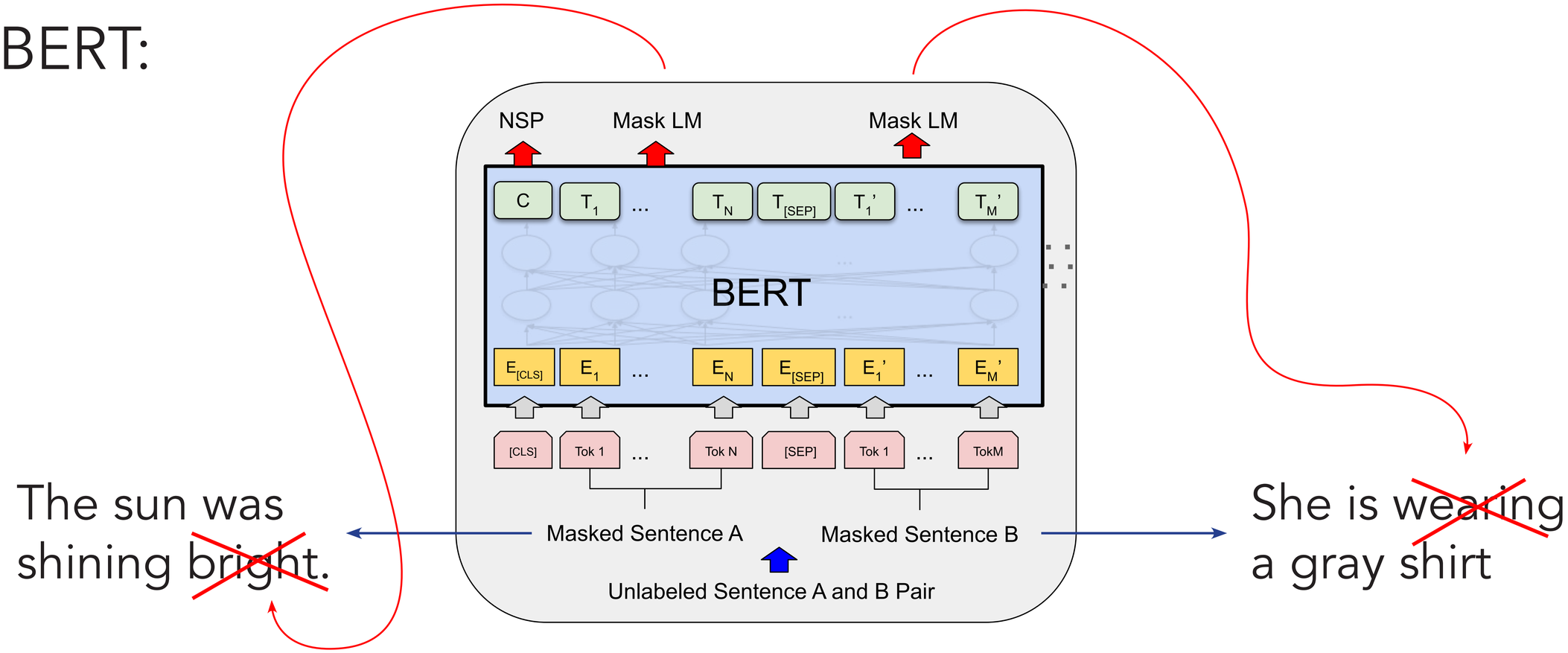
encoder

decoder

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. *Attention is all you need*. In NEURIPS 2017, 2017.

Transformer neural networks

- BERT:



Transformer neural networks

- BERT:

"The law will never be perfect, but its application should be just, this is what we are missing, in my opinion."

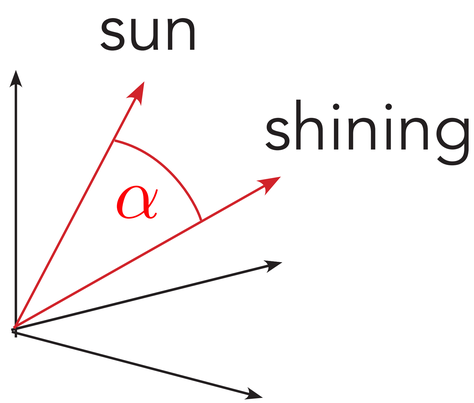
Transformer neural networks

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.

Transformer neural networks

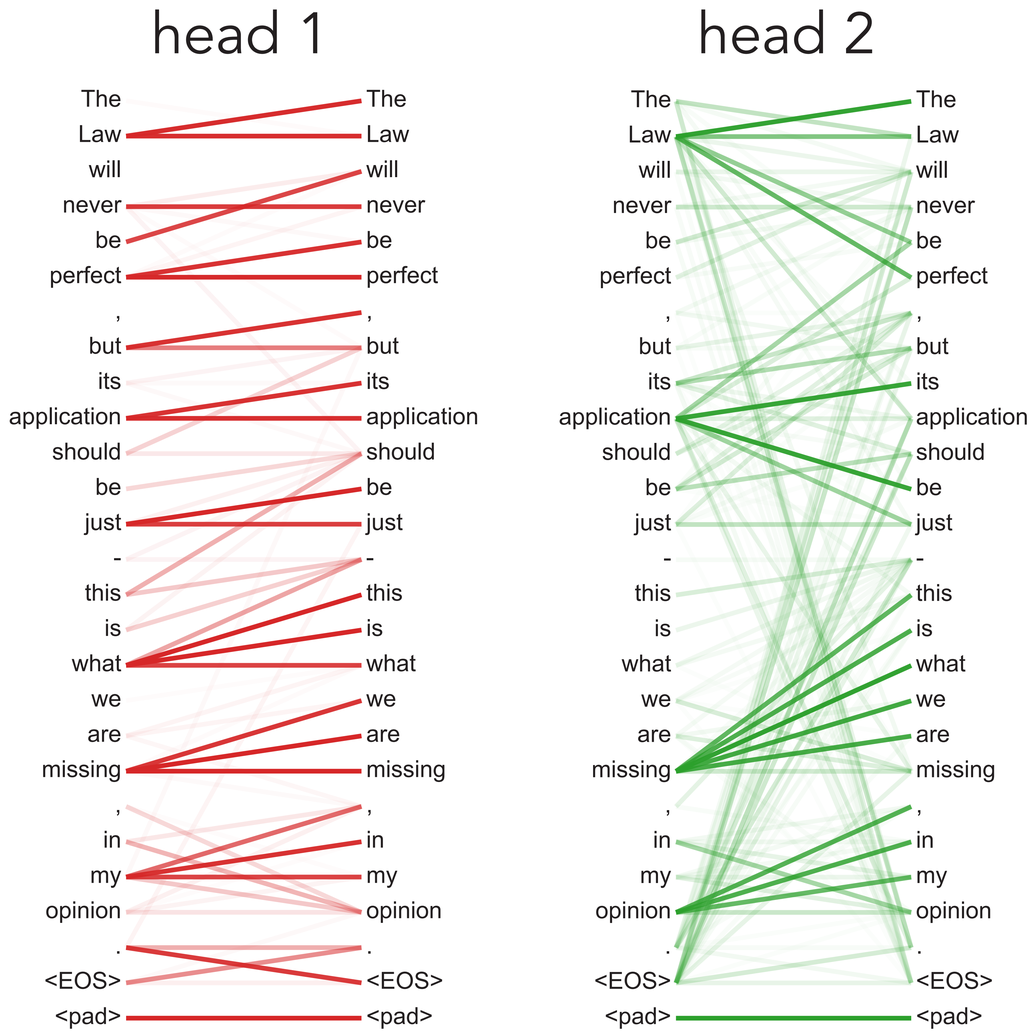
A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, Advances in Neural Information Processing Systems, volume 30. Curran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

The	sun	was	shining	bright.



Transformer neural networks

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.



Transformer neural networks for vision

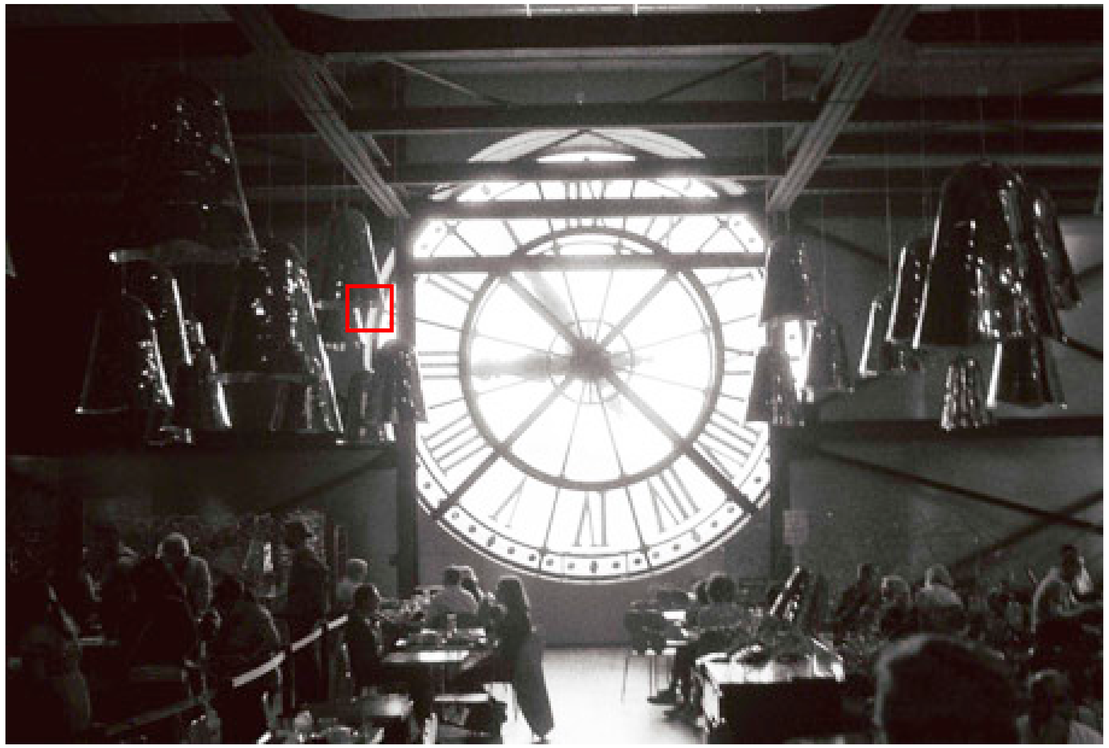
- Wide adoption of transformers for vision took 4 years

Transformer neural networks for vision

- Wide adoption of transformers for vision took 4 years

Transformer neural networks for vision

- Wide adoption of transformers for vision took 4 years



Transformer neural networks for vision

- Wide adoption of transformers for vision took 4 years
  - Token initially often defined as a pixel: too small and too many
  - Training with insufficient amounts of data leads to worse performance than CNN
  - Token as 16x16 pixels now widely adopted and substantially improved performance with sufficient data

Transformer neural networks

- DINO (computer vision):

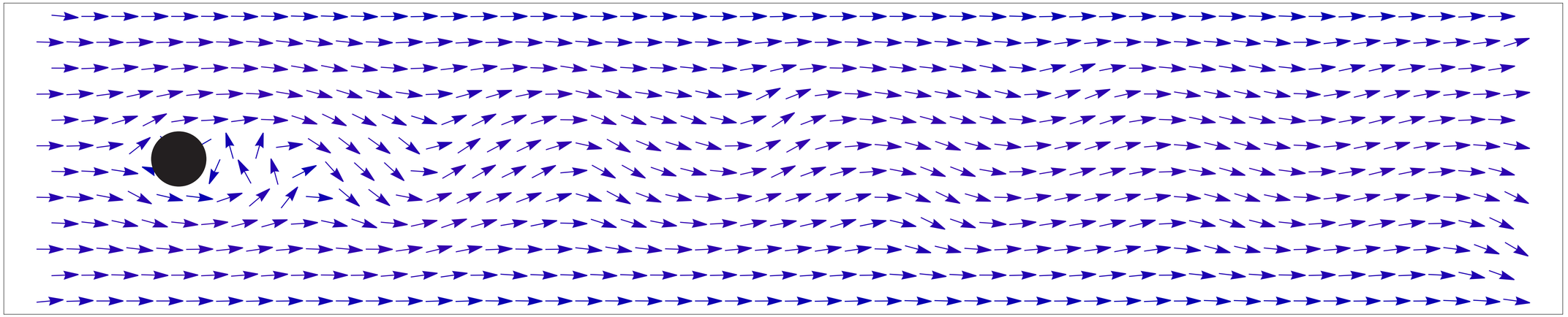
M. Caron, H. Touvron, I. Misra, H. Jegou, J. Mairal, P. Bojanowski, and A. Joulin. Emerging properties in self-supervised vision transformers. CoRR, abs/2104.14294, 2021.

Transformer neural networks

- DINO (computer vision):

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 for fluid flow



Transformers for fluid flow

vorticity

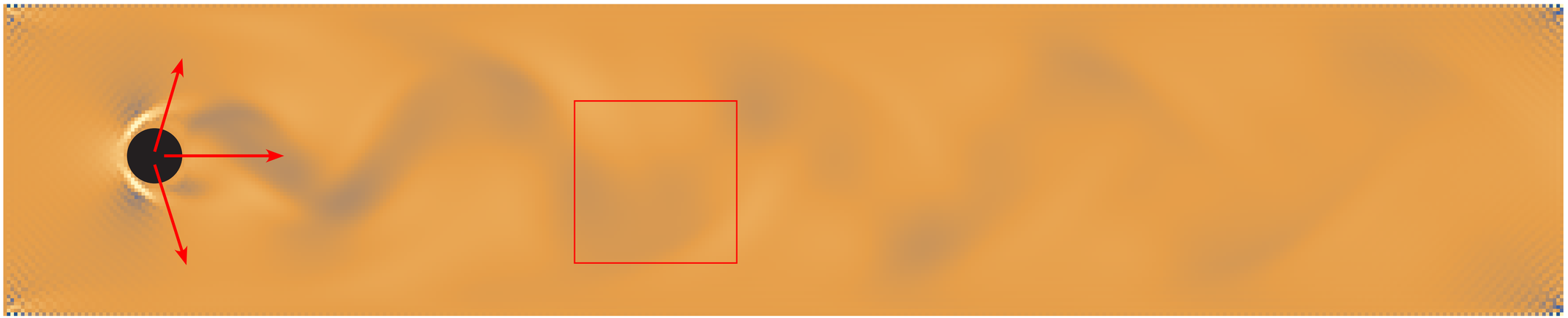
Transformers for fluid flow

vorticity

training with varying position
and spherical eccentrity

Transformers for fluid flow

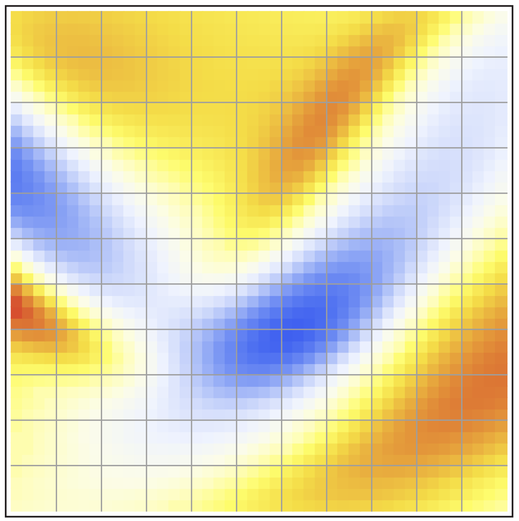
vorticity



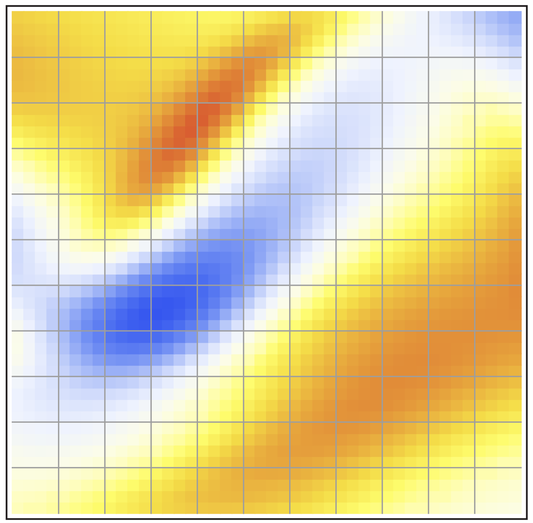
training with varying position
and spherical eccentrity

Transformers for fluid flow

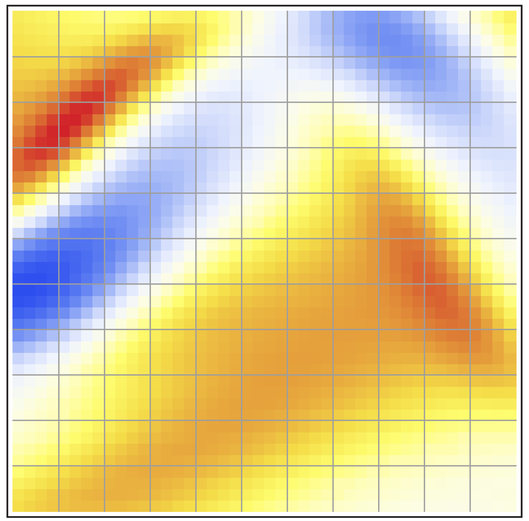
t-2



t-1

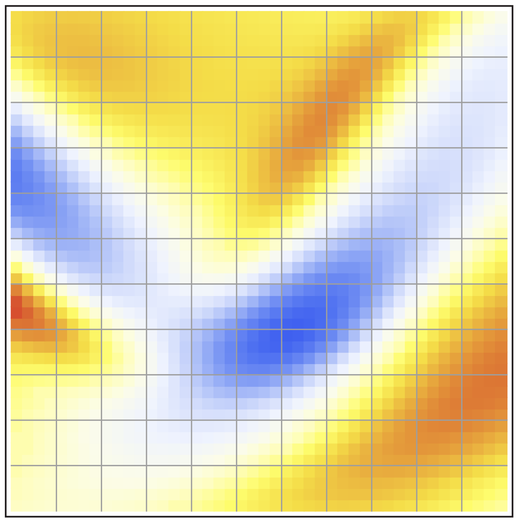


t

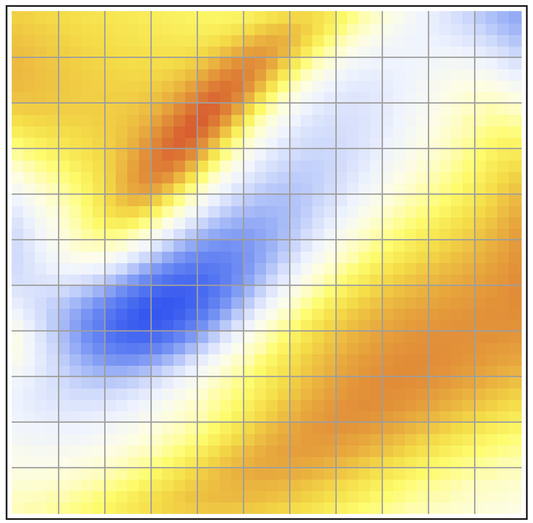


Transformers for fluid flow

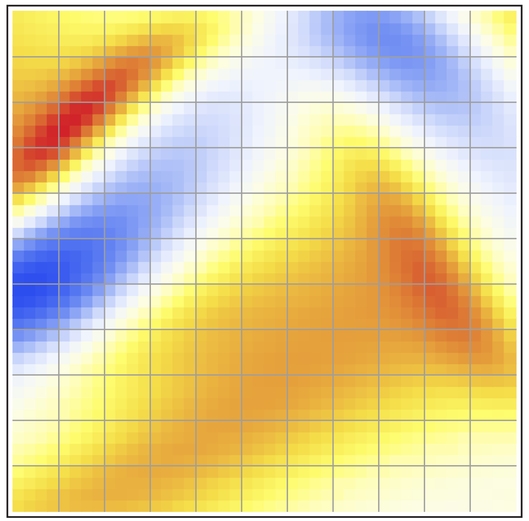
t-2



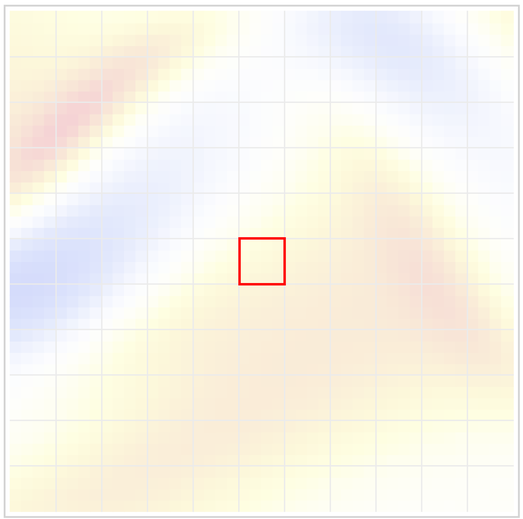
t-1



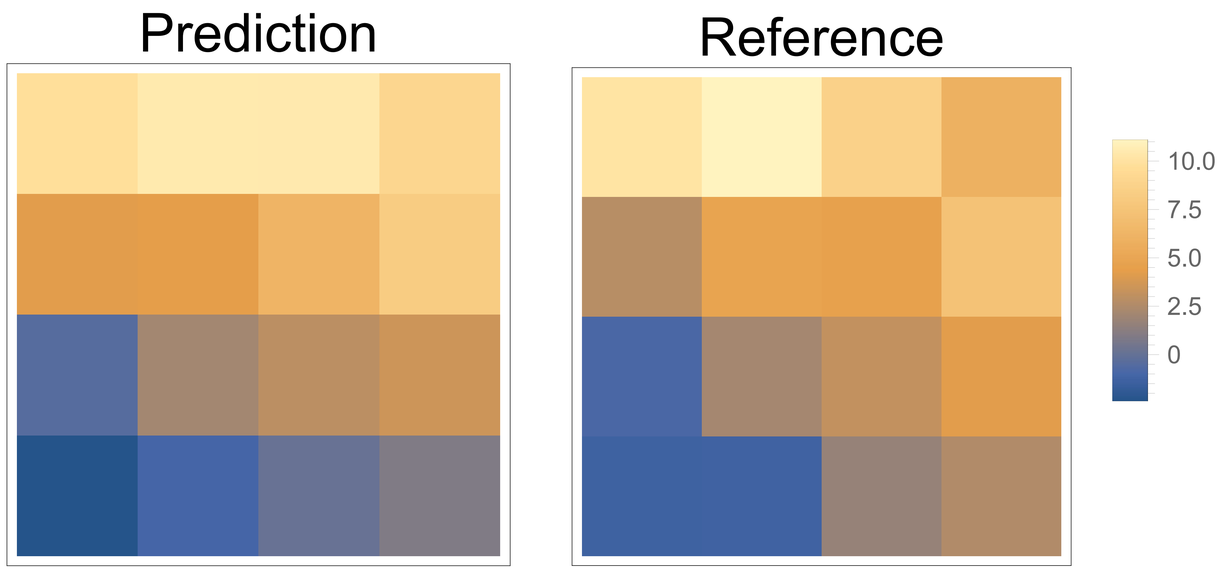
t



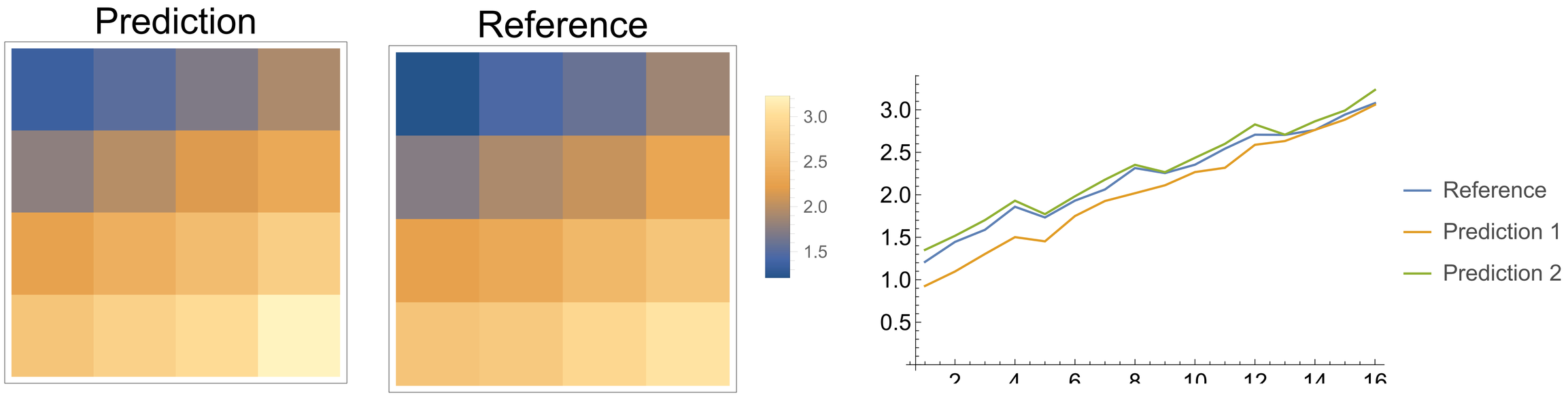
t+1



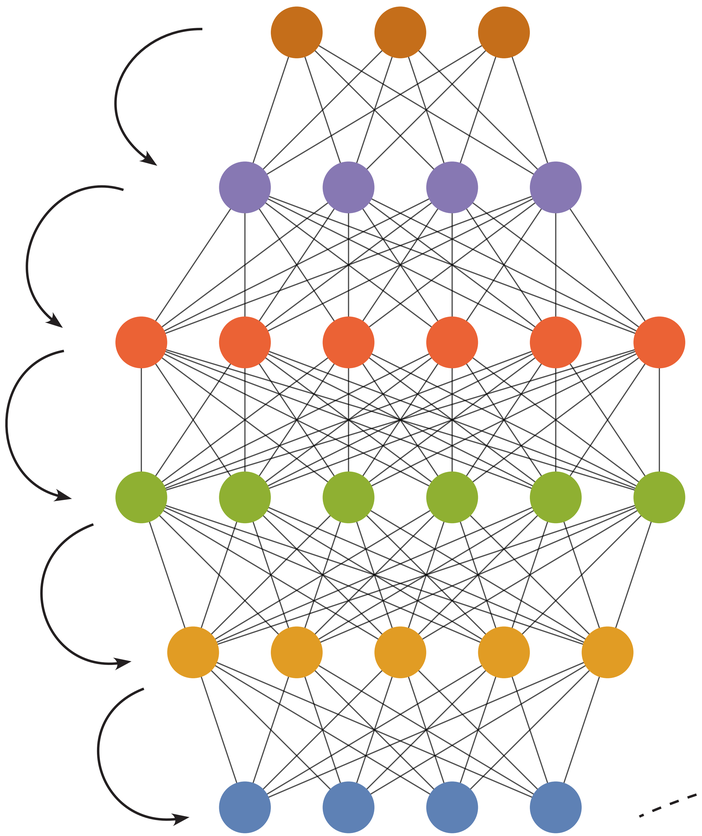
Transformers for fluid flow



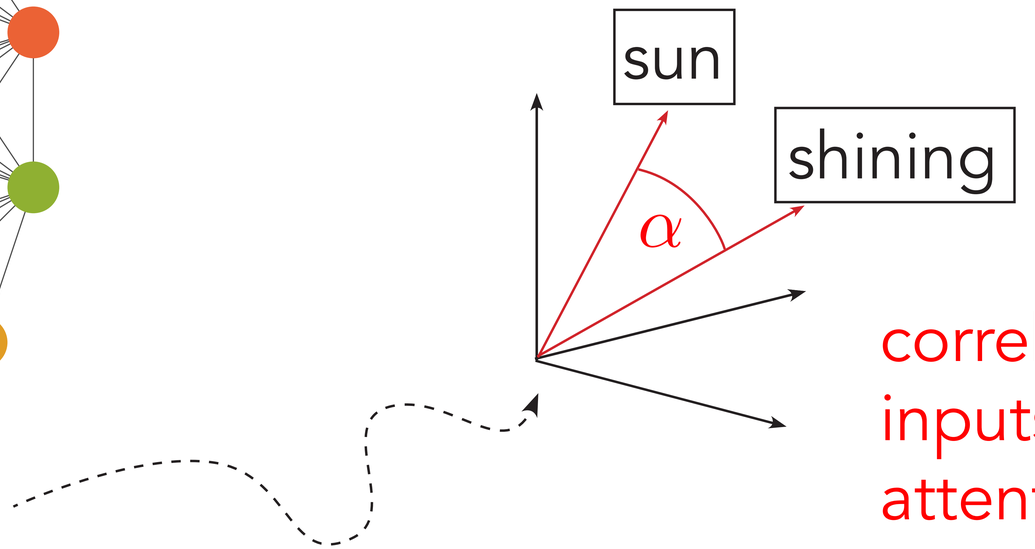
Transformers for fluid flow



Transformers for fluid flow

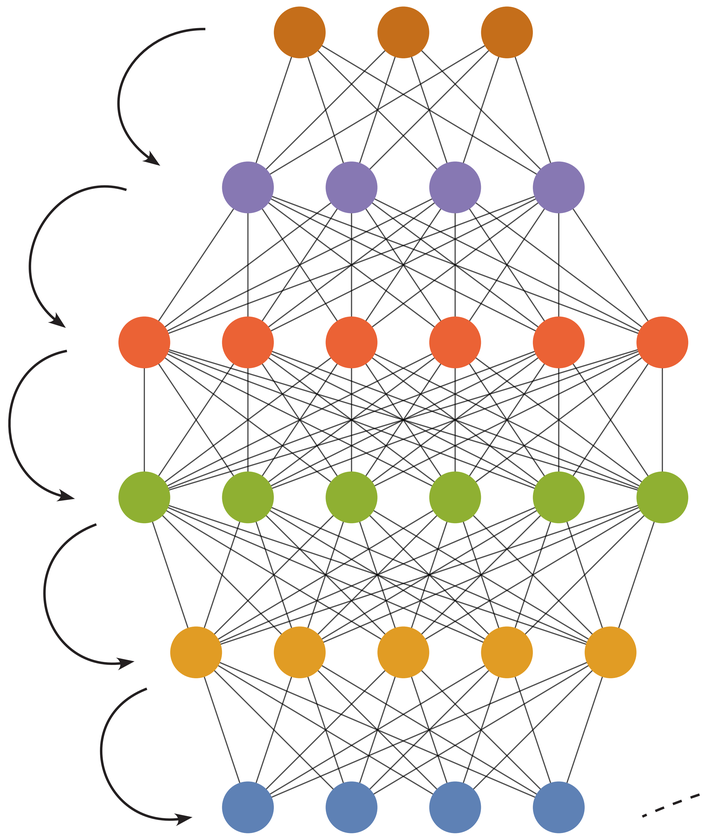


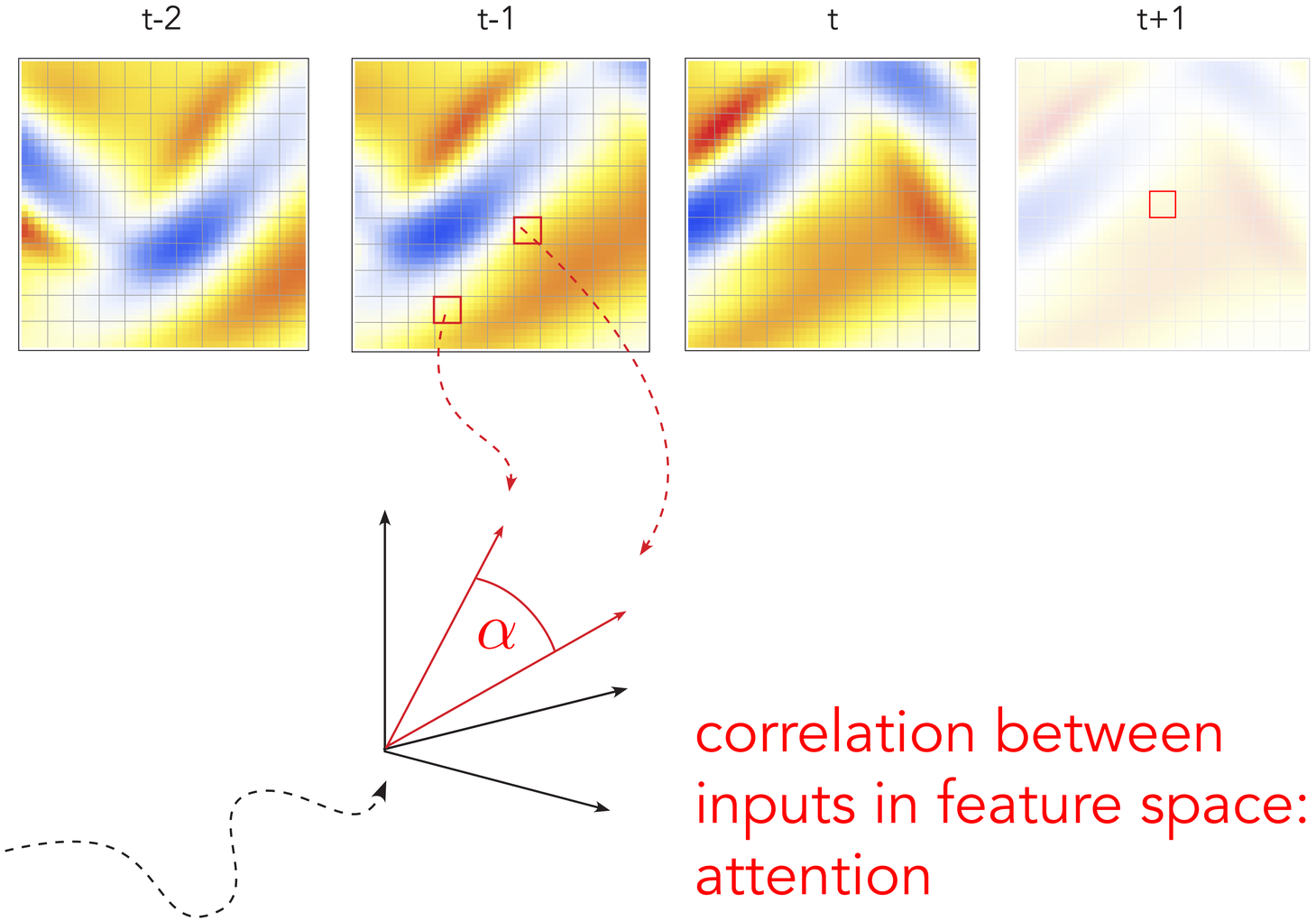
The	sun	was	shining	bright.



correlation between inputs in feature space: attention

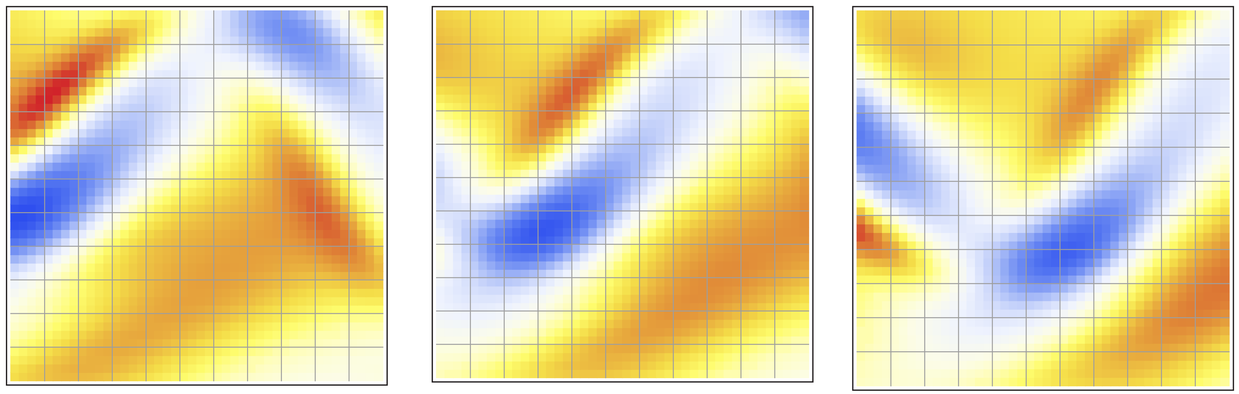
Fluid flow





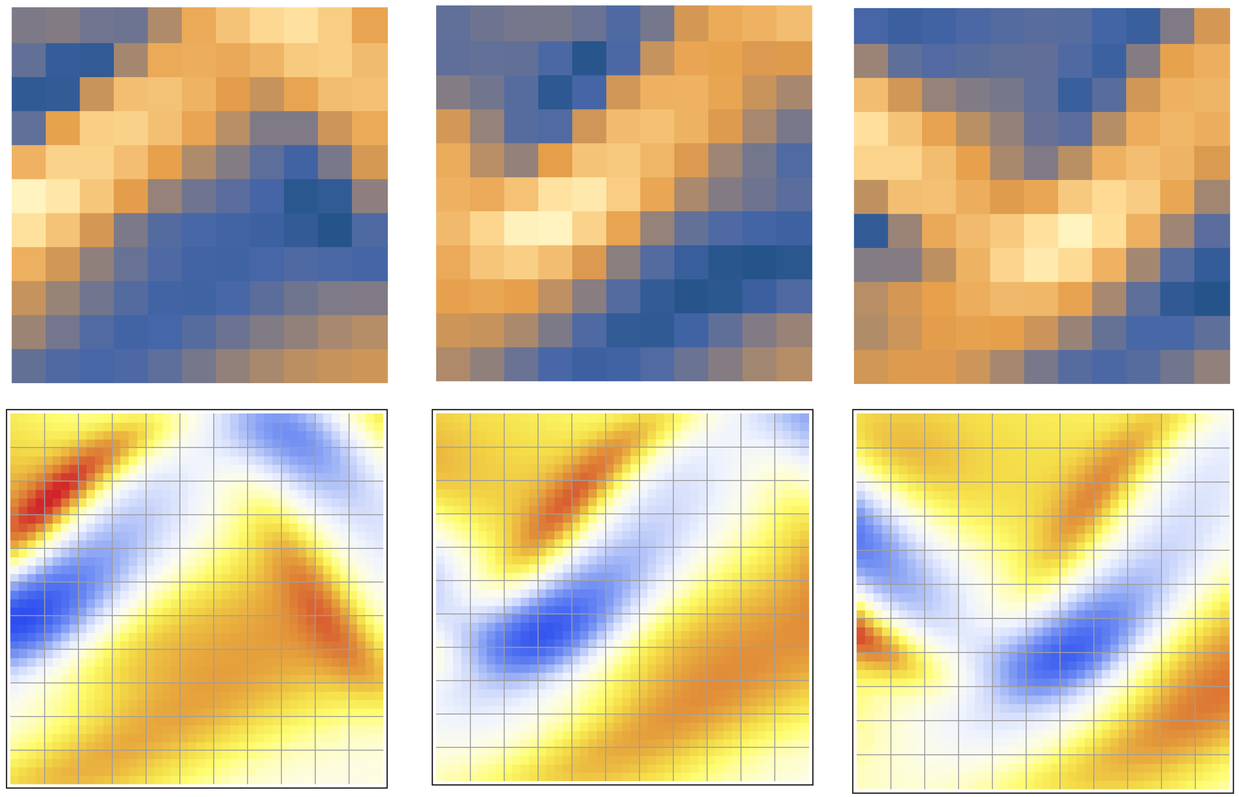
correlation between inputs in feature space: attention

Transformers for fluid flow

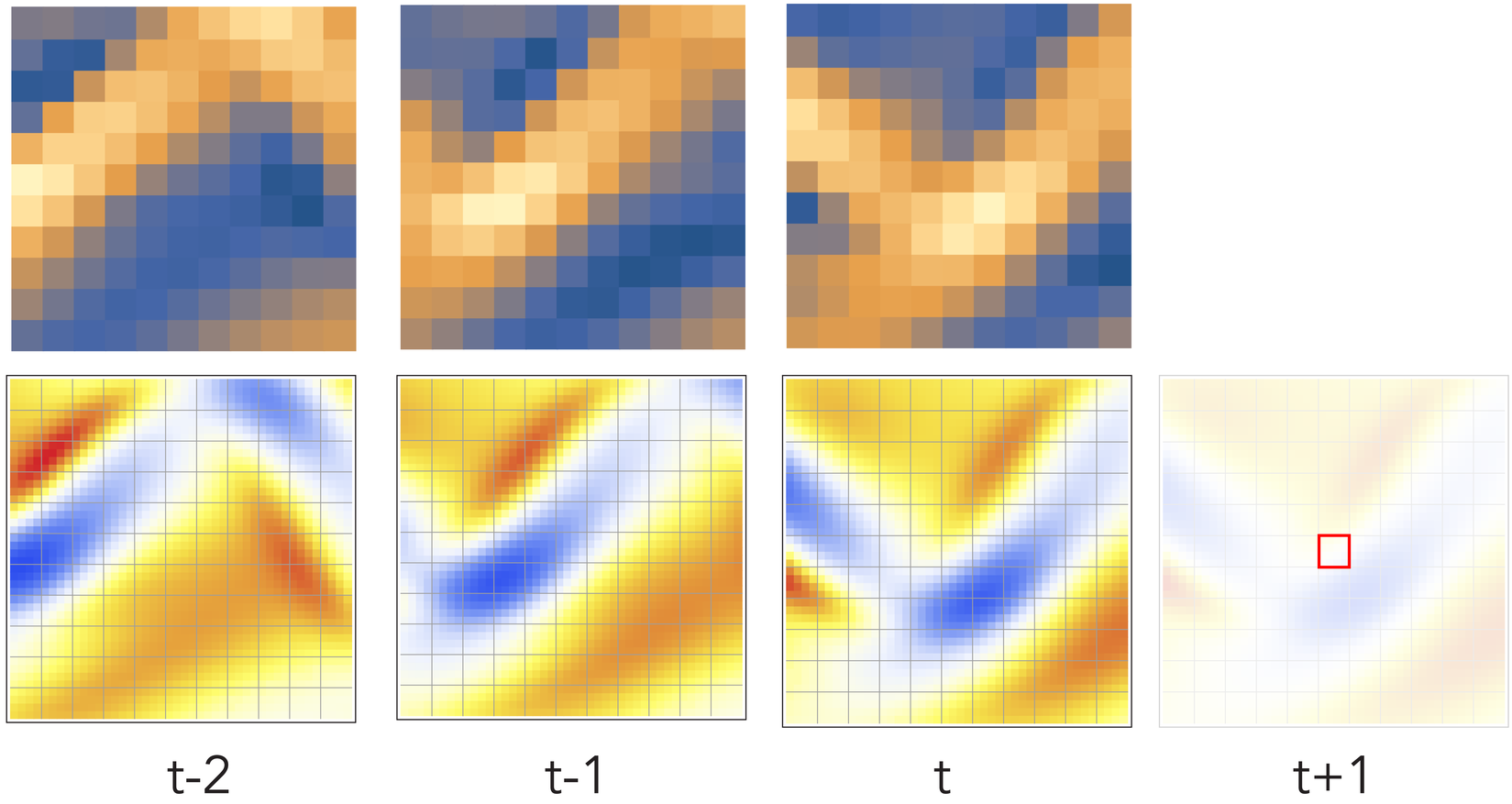


t-2 t-1 t

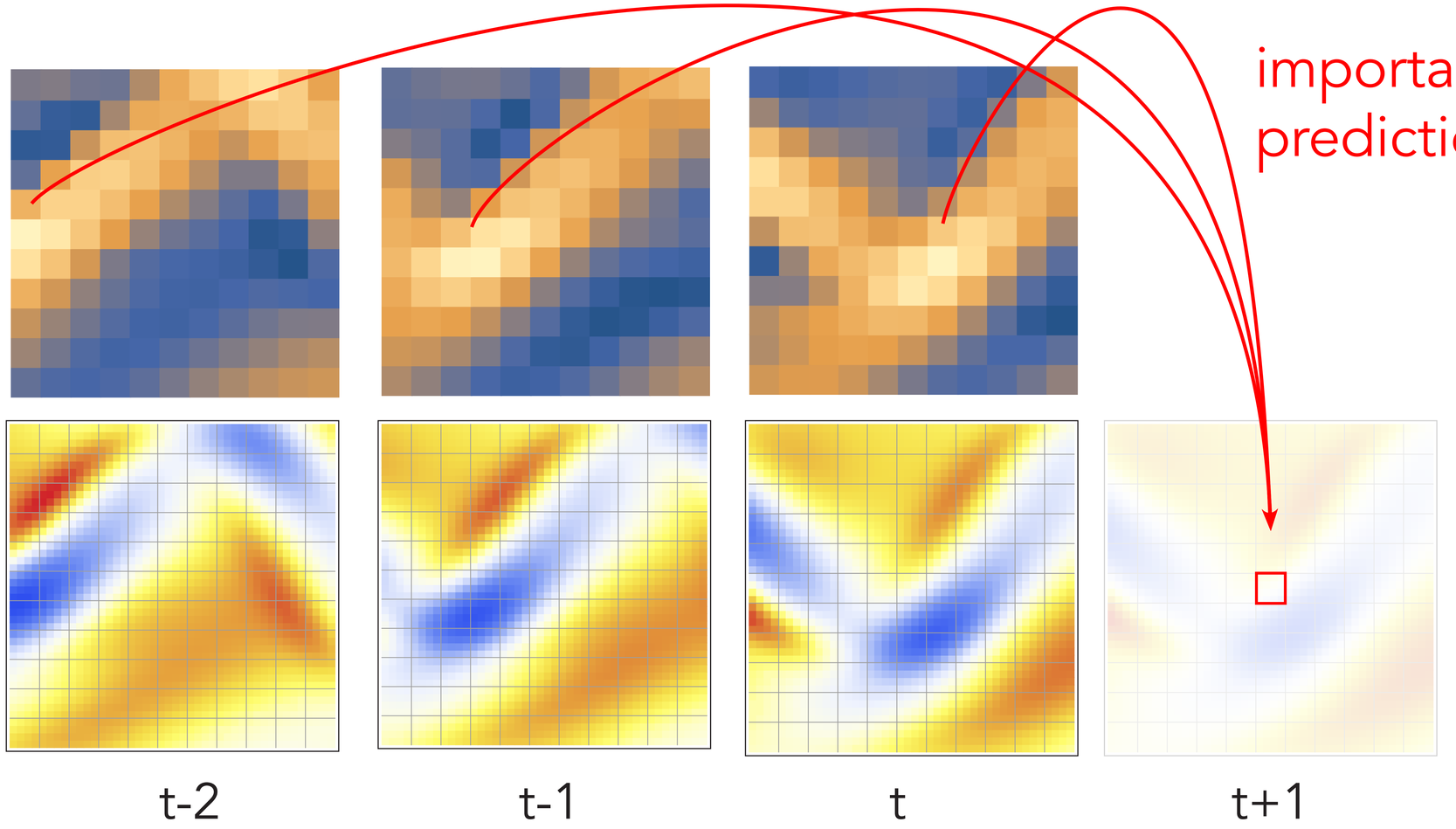
Transformers for fluid flow



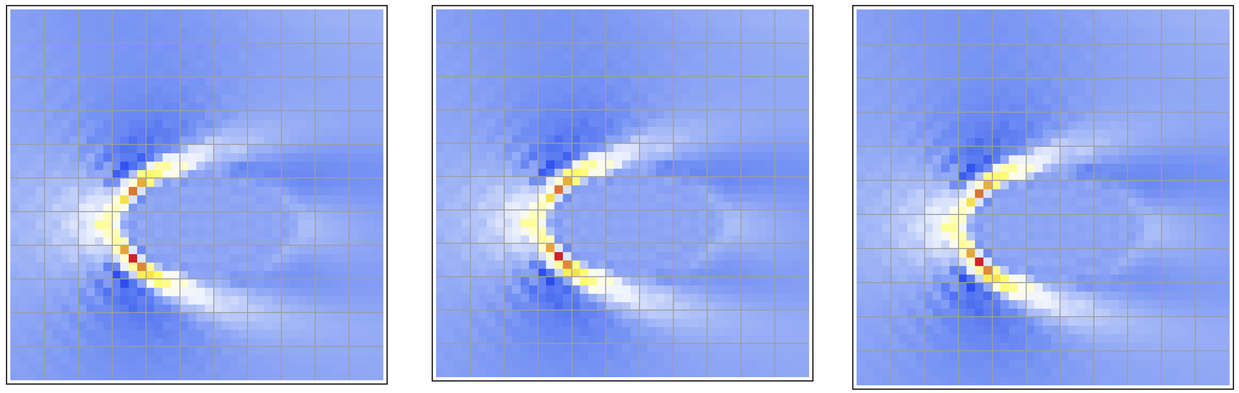
Transformers for fluid flow



Transformers for fluid flow

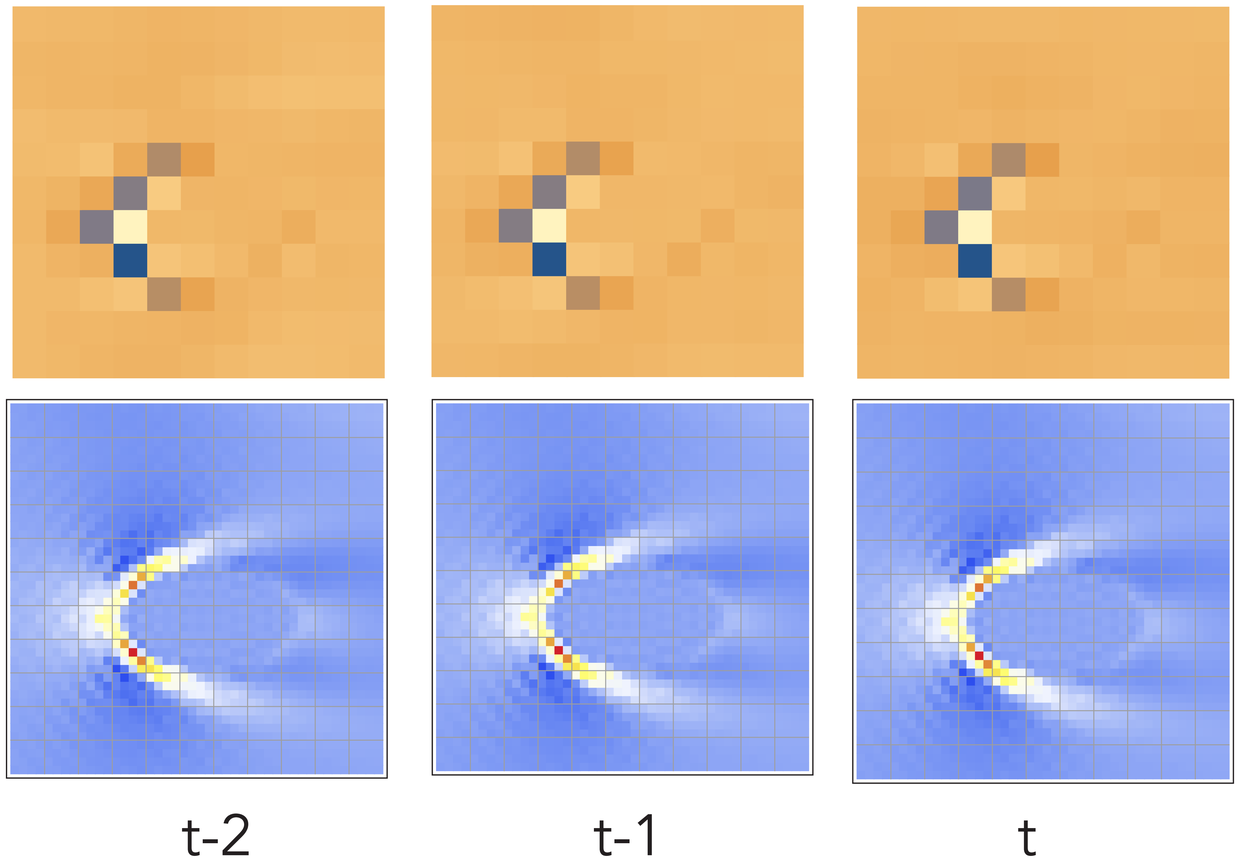


Transformers for fluid flow

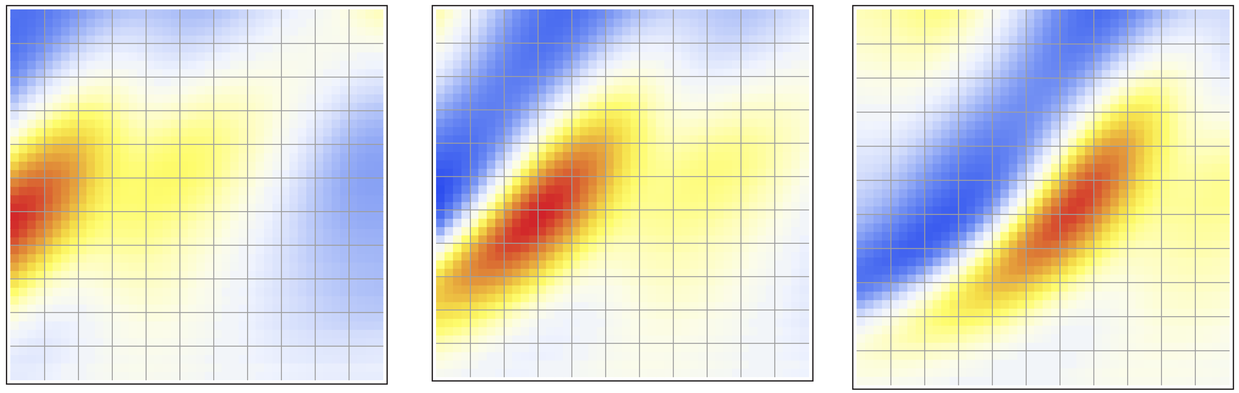


t-2 t-1 t

Transformers for fluid flow

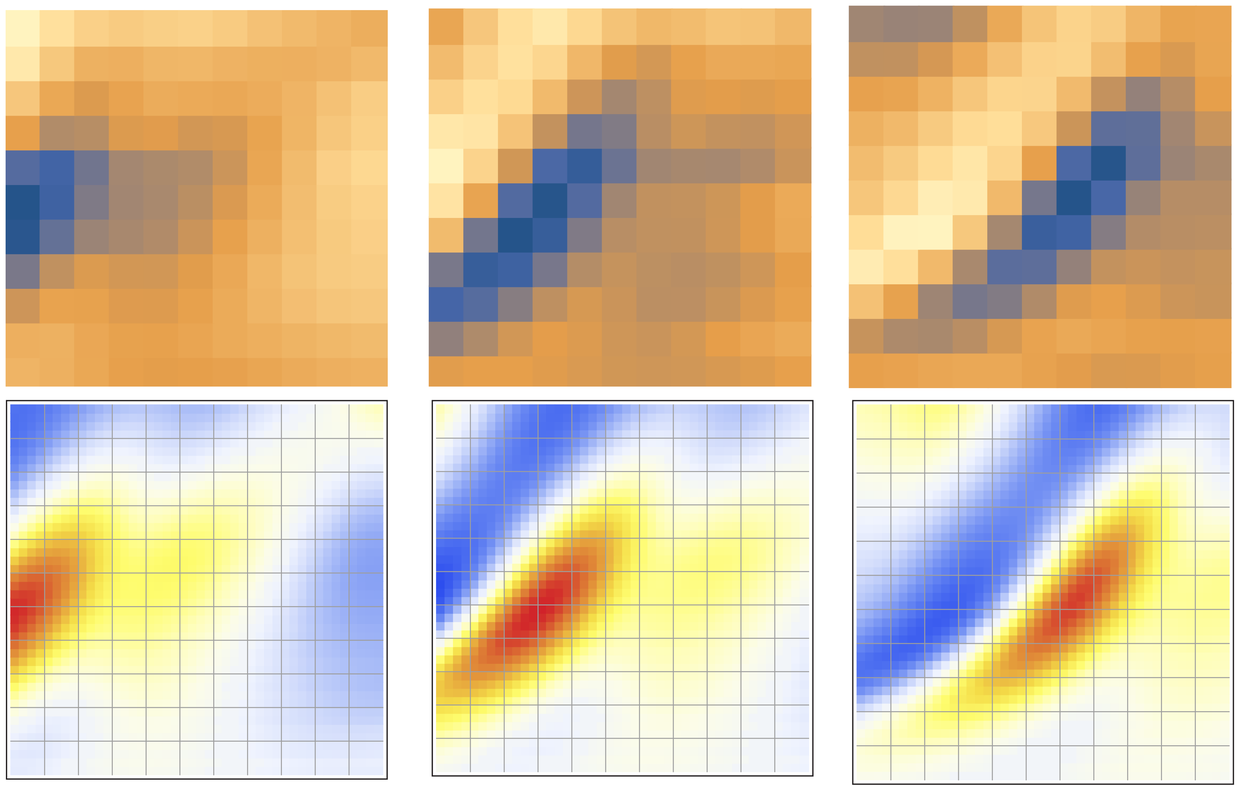


Transformers for fluid flow



t-2 t-1 t

Transformers for fluid flow



Transformers for fluid flow

- The classical perspective
  - Token: finite volume/discontinuous FEM cell
  - Attention: learnable inner product
  - Feature/embedding space: analogous to PCA (or complex bases such as wavelets)

Transformer neural networks

- Scale well to highly parallel training and billions of parameters, especially for sequential data
- Attention mechanism allows to model complex dependencies in data
  - Learned inner product in feature space with “legs” in input domain
  - Direct interpretation of learned representations through attention maps

Transformer neural networks

- Standard model for language models and vision
- Attention mechanism allows to model complex dependencies in data
- Attention can also be used creatively wherever interaction between inputs / part of the input need to be modelled
- Generative prediction model

How can we adapt representation learning and transformers to science and engineering?

Rep. learning for science and engineering?

- Often very large amounts of data
  - CMIP6: 100+ PB
  - E.g. MetOp-SG: 8 x 864 GB/day

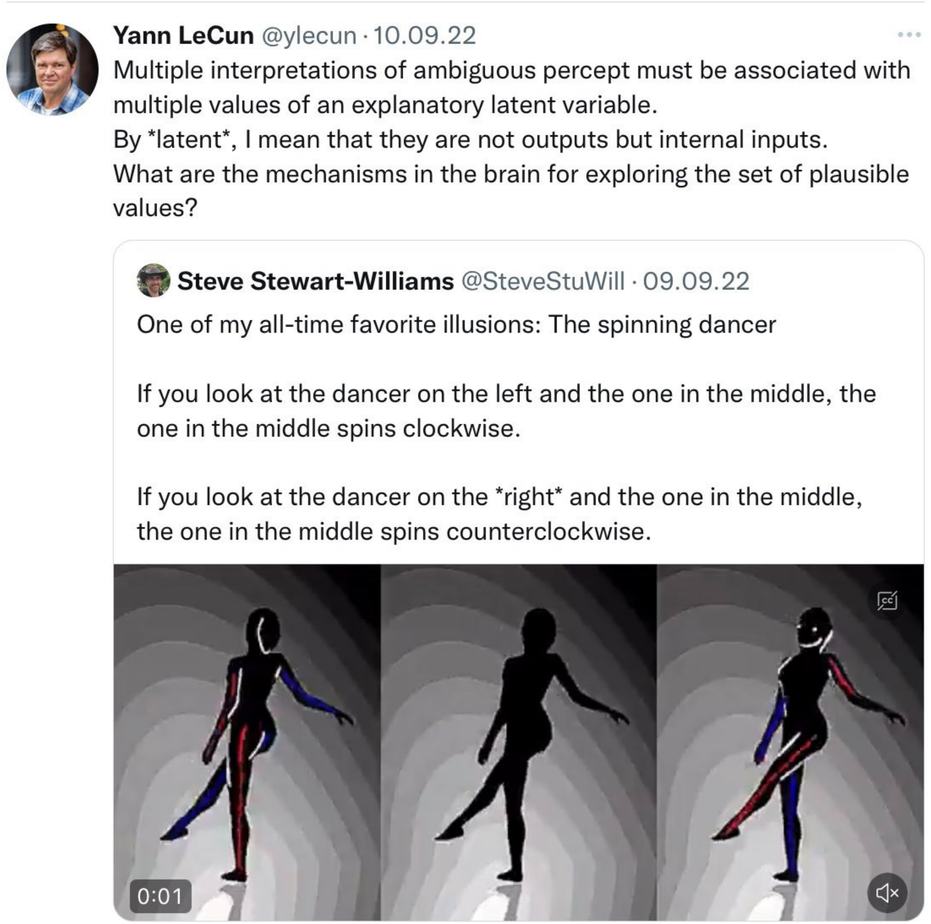
Rep. learning for science and engineering?

- Often very large amounts of data
- Often no complete classical model for system and dynamics in complex interacting systems
  - Central issue for weather and climate projections

Rep. learning for science and engineering?

- Often very large amounts of data
- Often no complete classical model for system and dynamics in complex interacting systems
- Large networks learn statistical representations

Rep. learning for science and engineering?



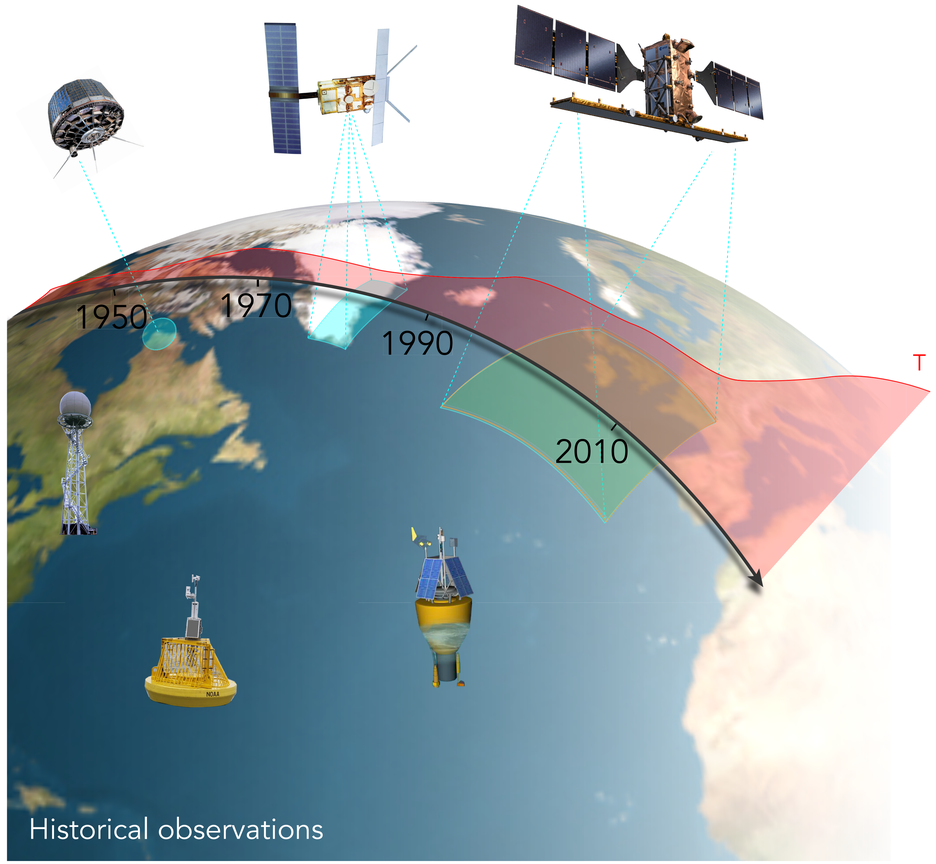
Rep. learning for science and engineering?

- Often very large amounts of data
- Often no complete classical model for system and dynamics in complex interacting systems
- Large networks learn statistical representations
  - Point predictions usually not meaningful in large complex systems

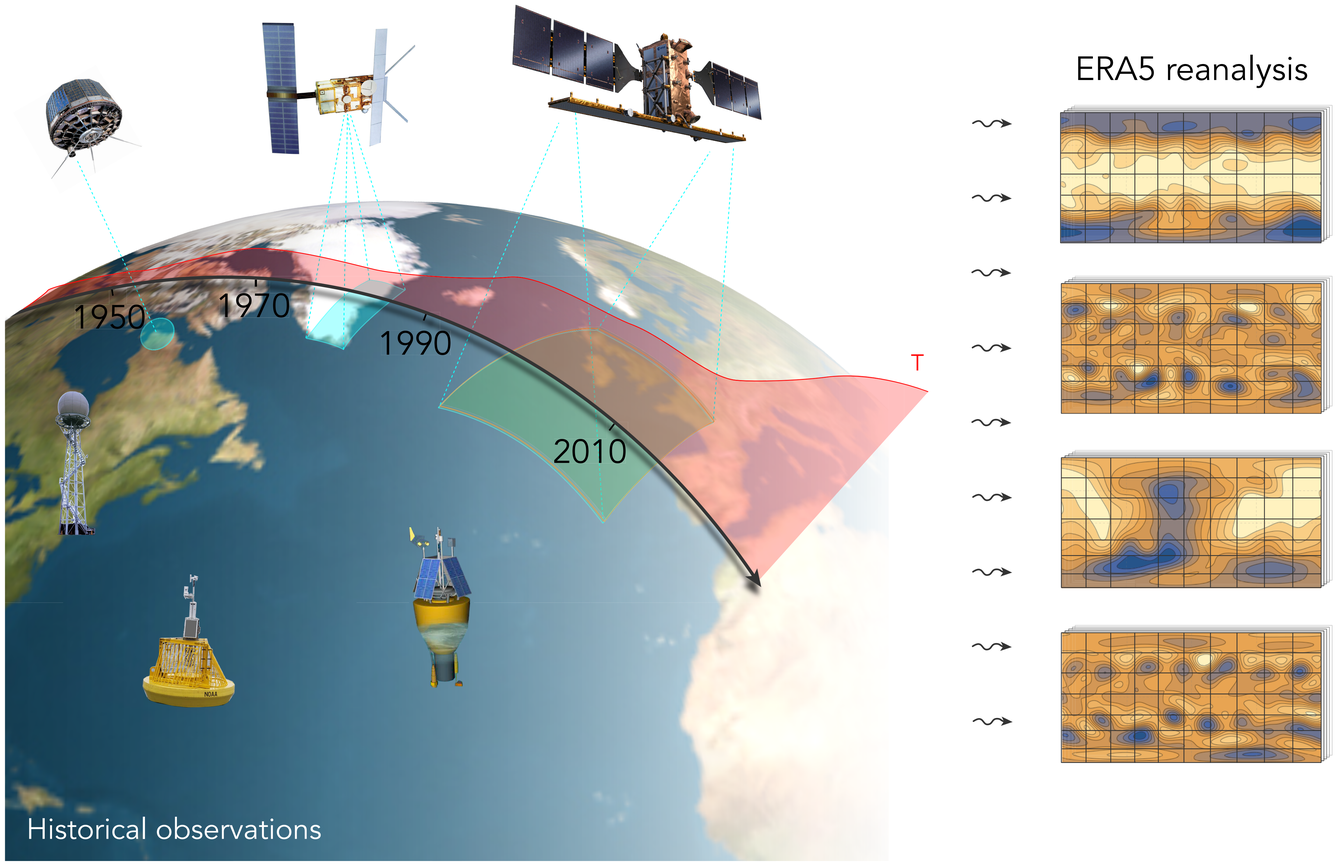
AtmoRep

Large scale representation learning of atmospheric dynamics

AtmoRep

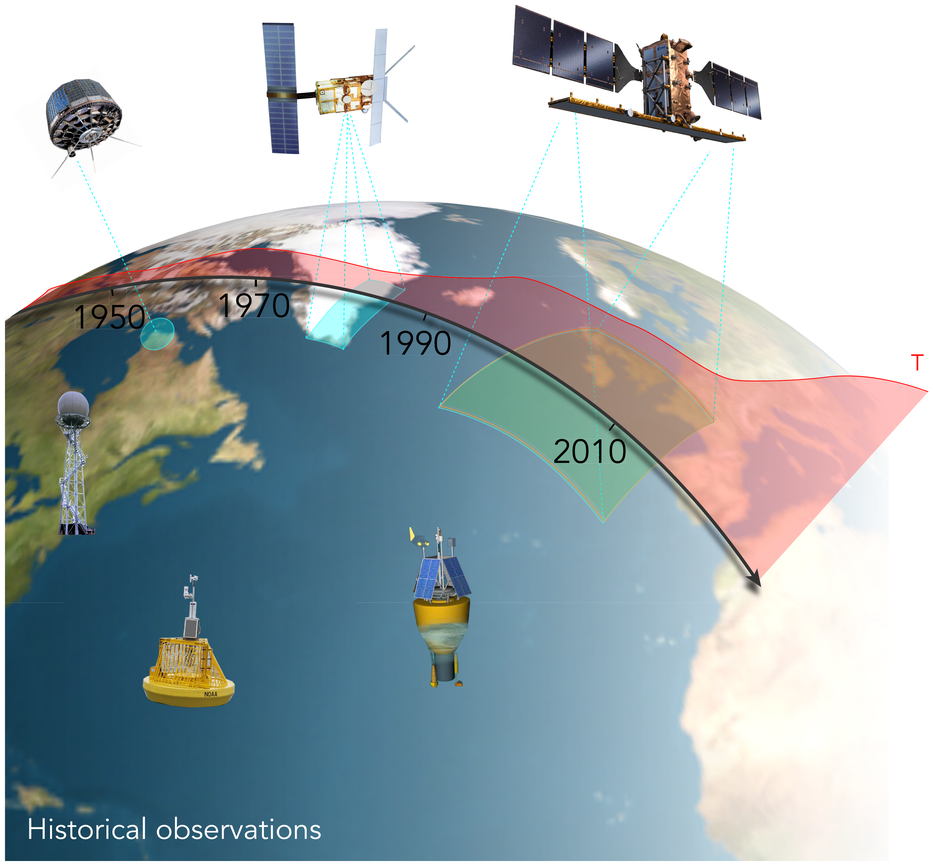


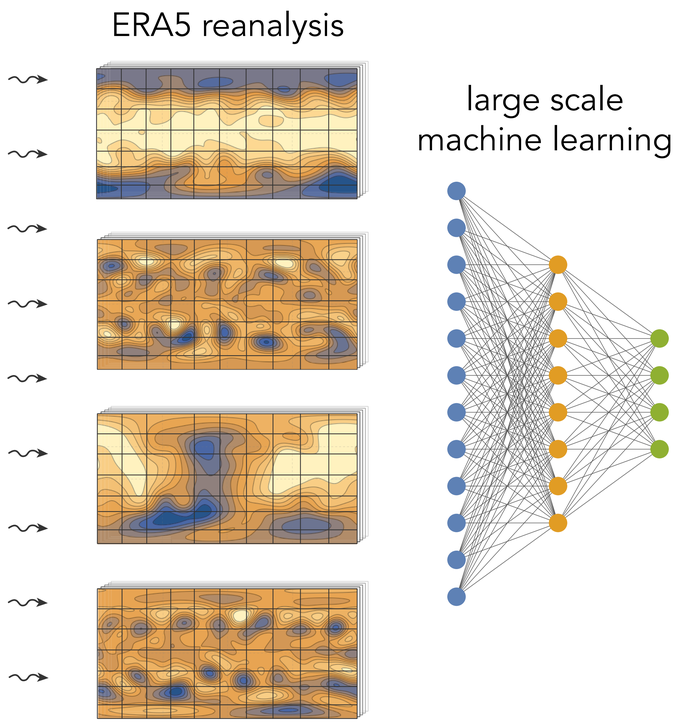
AtmoRep



ERA5 reanalysis

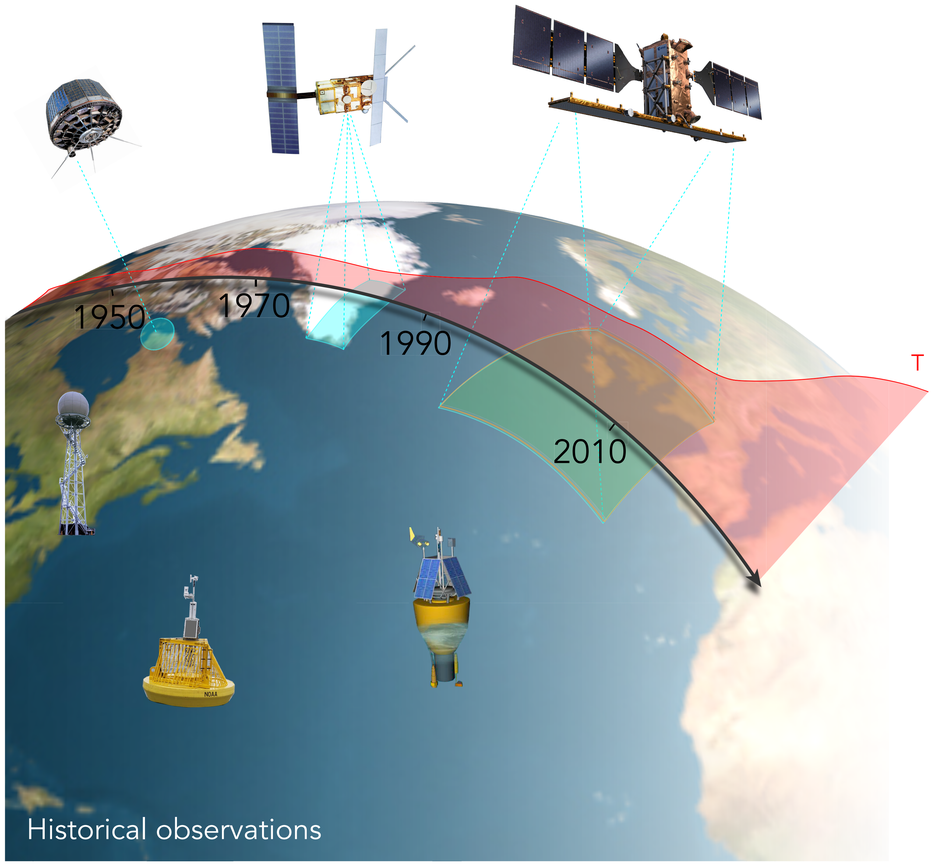
AtmoRep

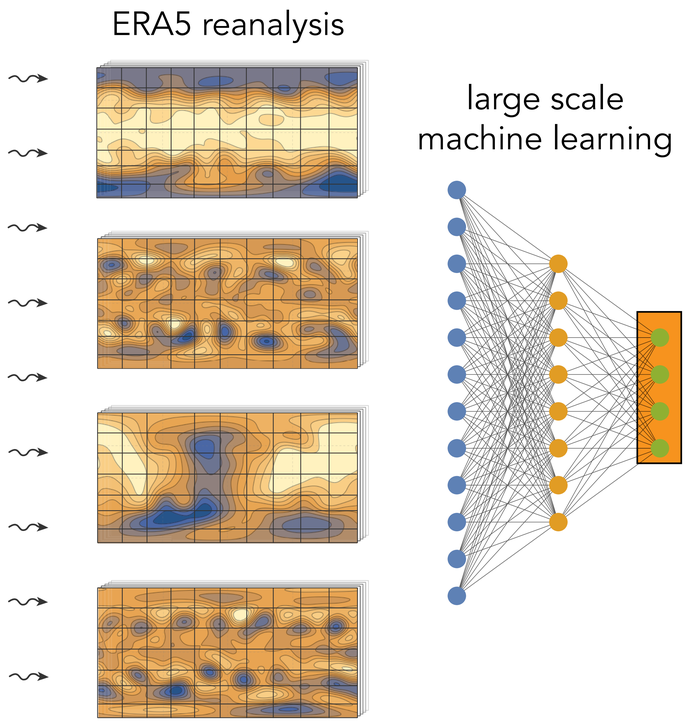




large scale
machine learning

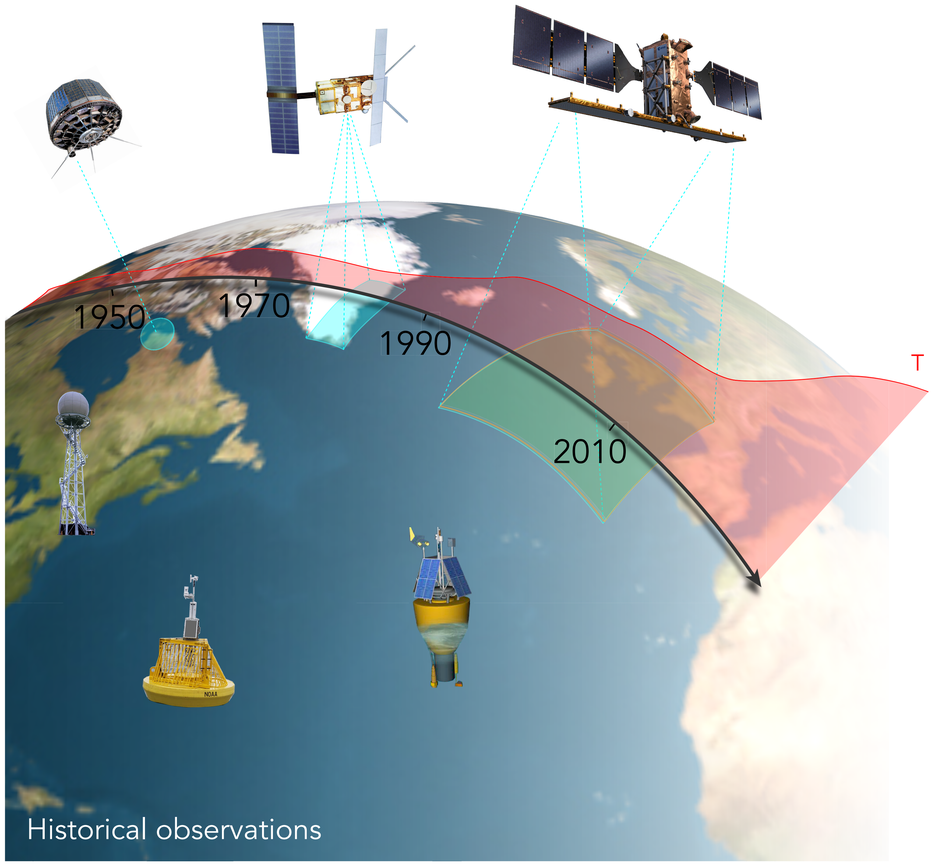
AtmoRep

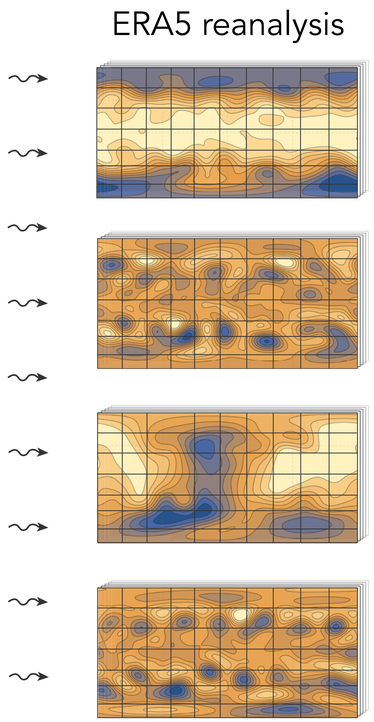


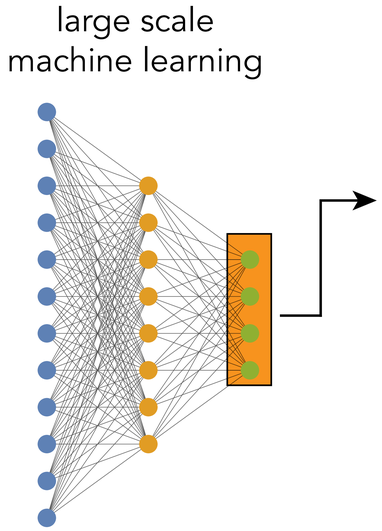


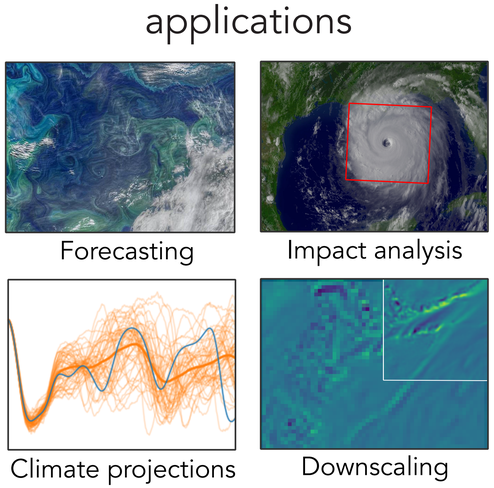
large scale
machine learning

AtmoRep

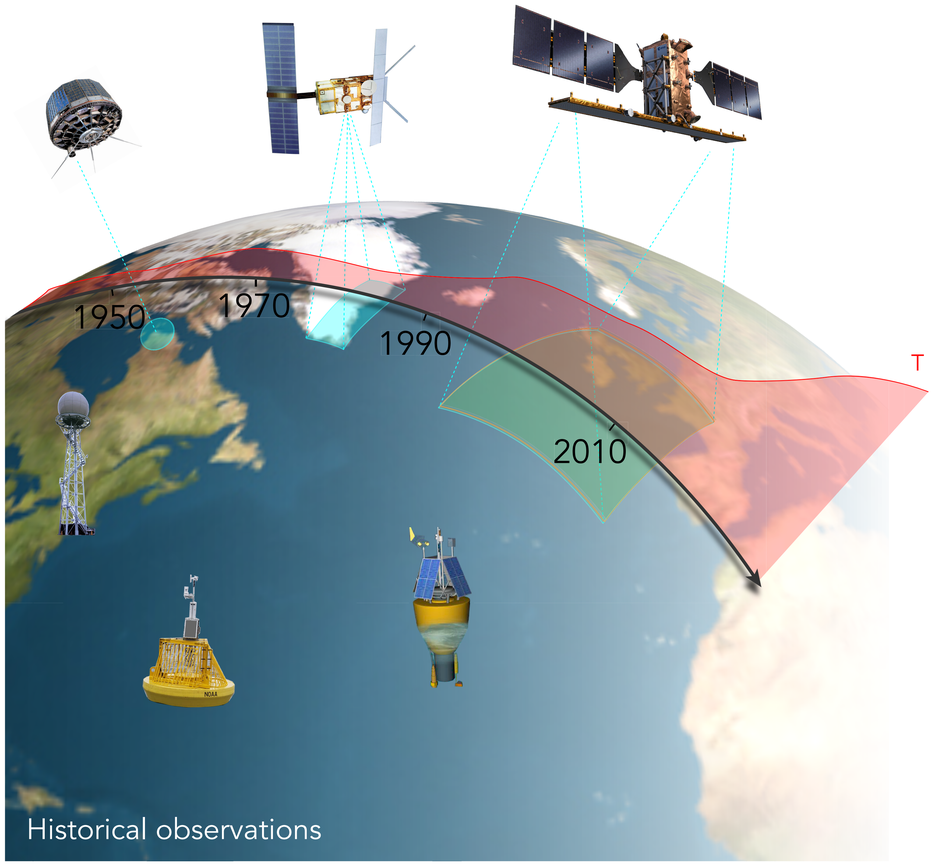


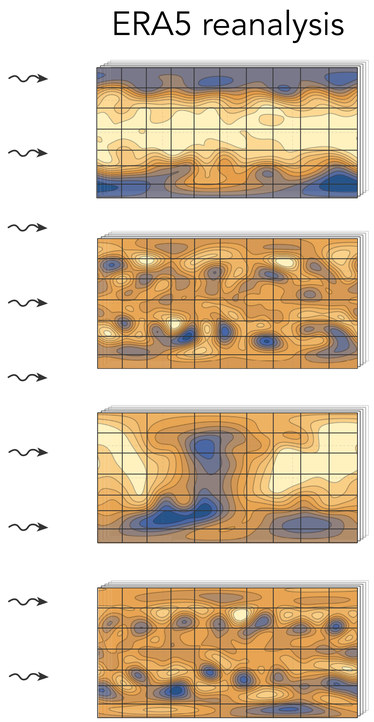


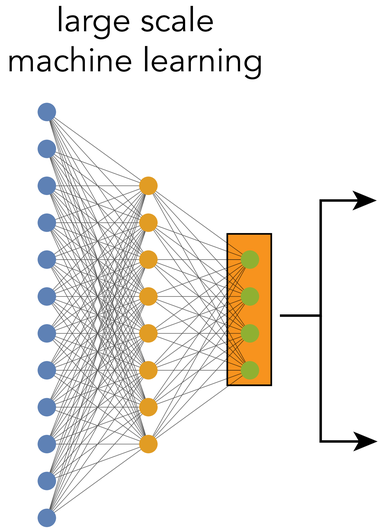


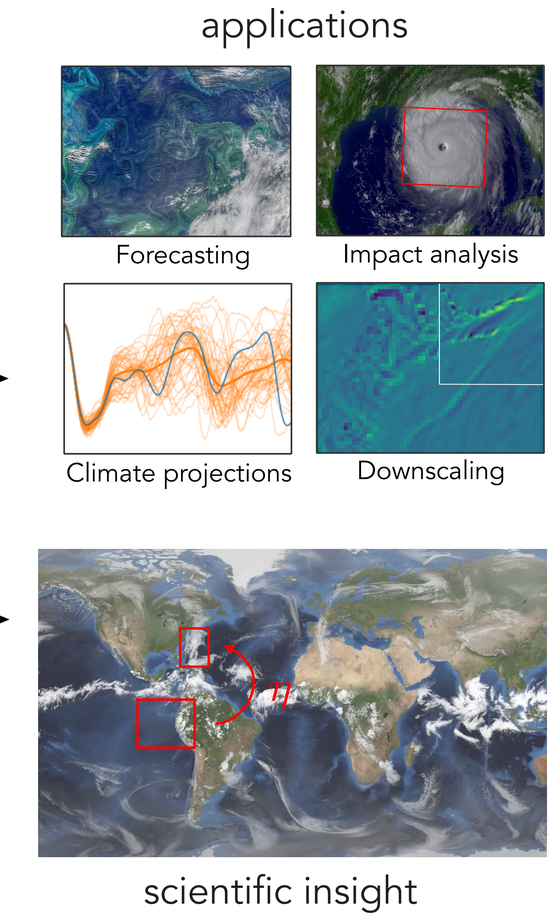


AtmoRep

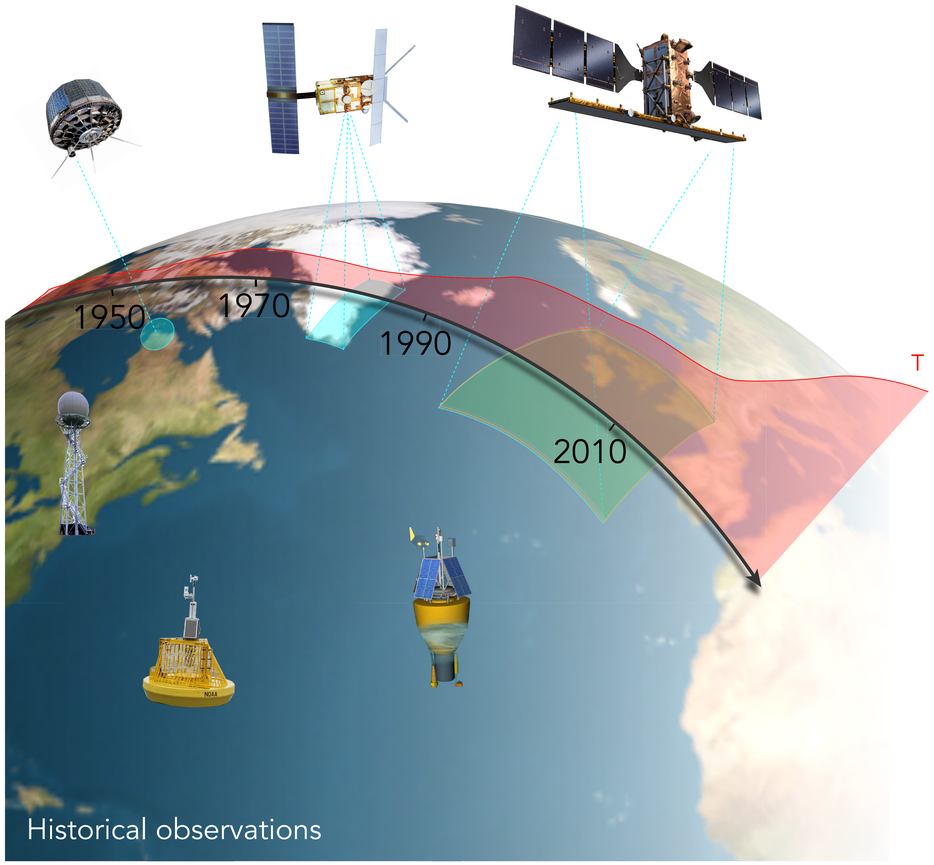


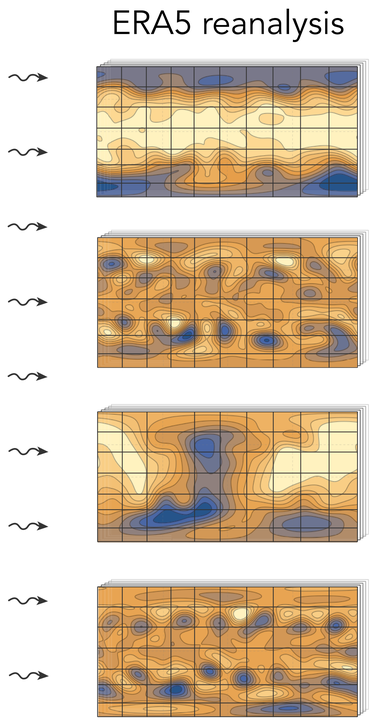


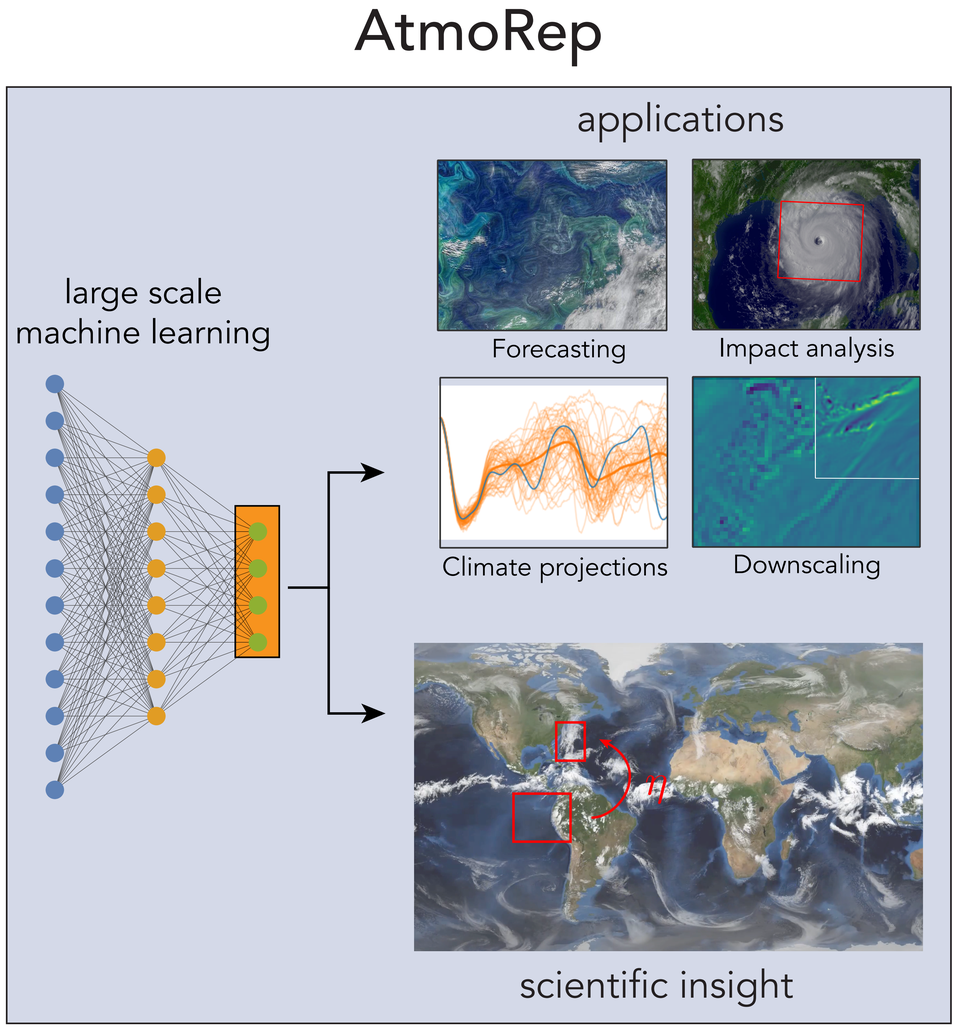




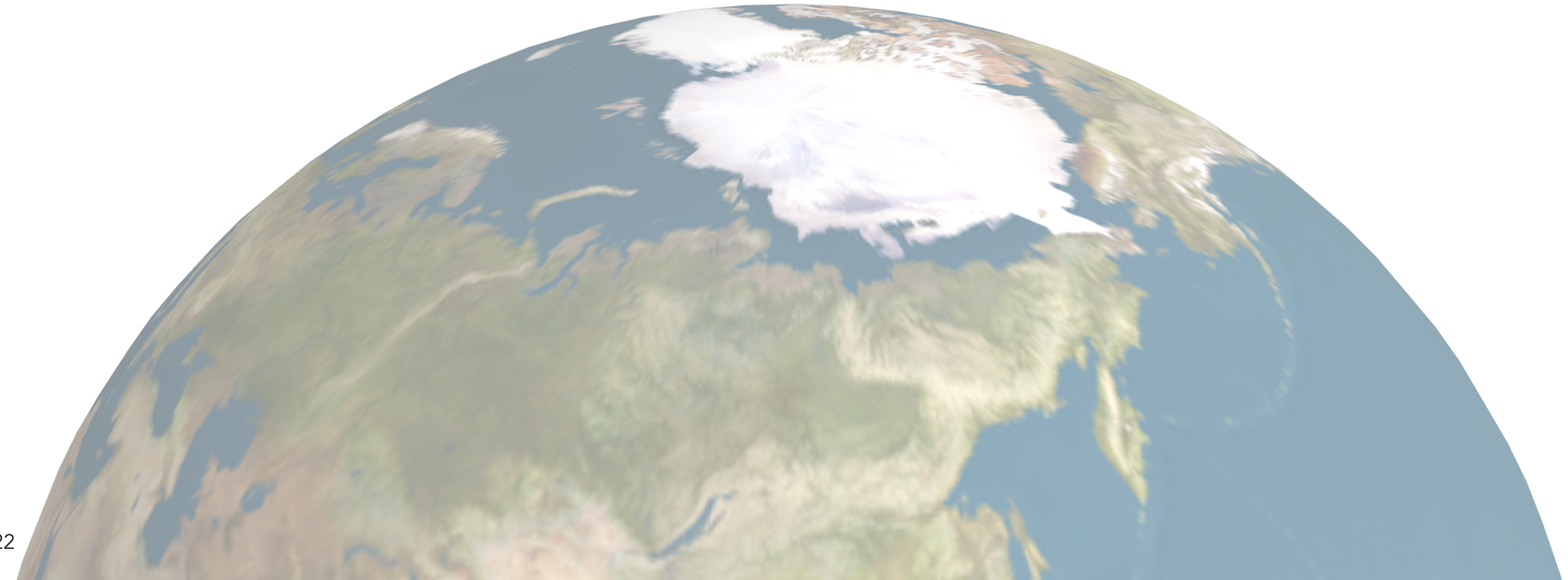
AtmoRep



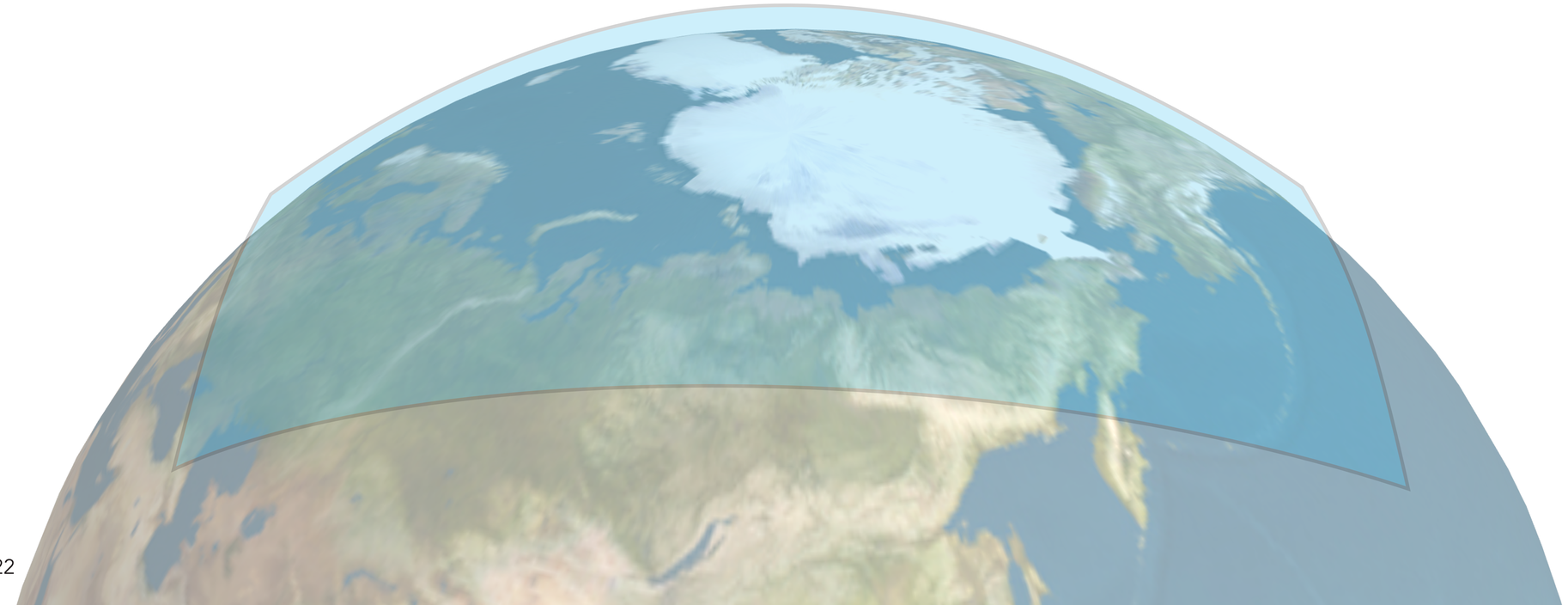




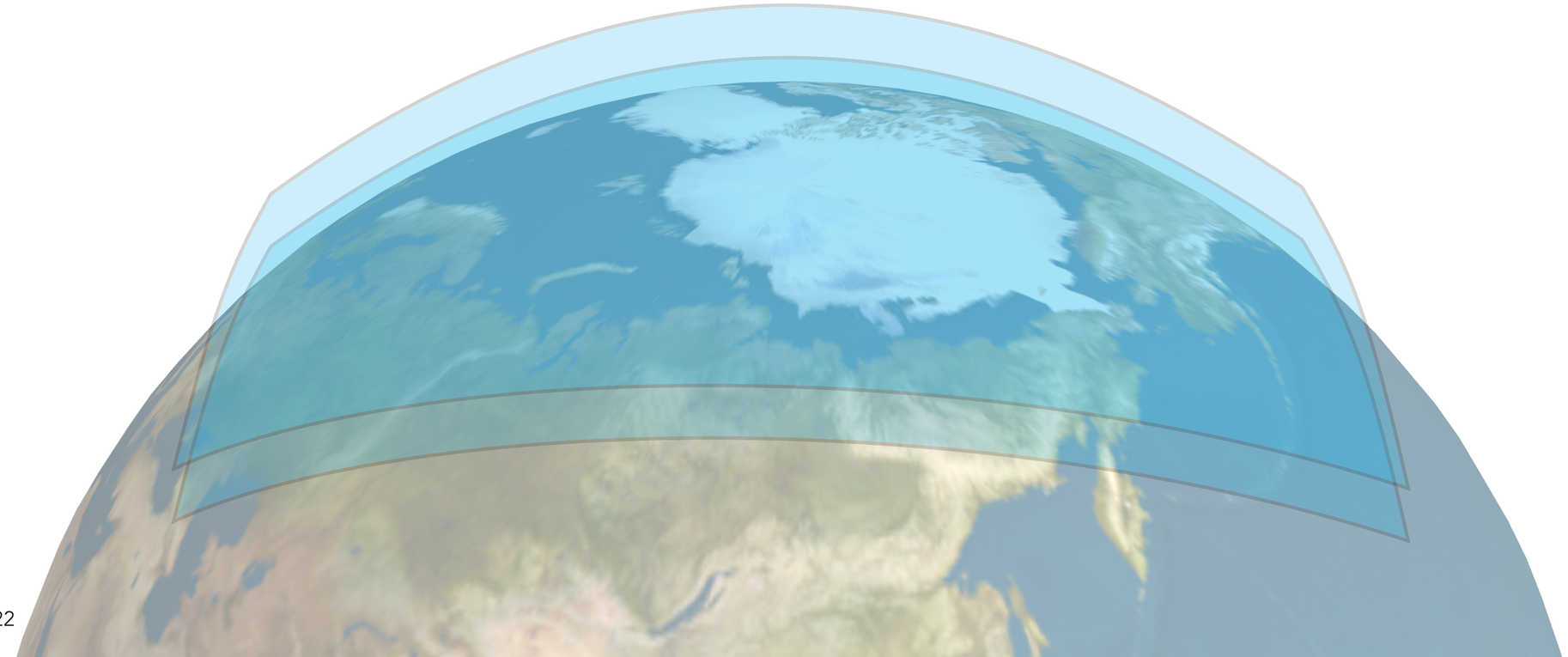
Data: ERA5 reanalysis



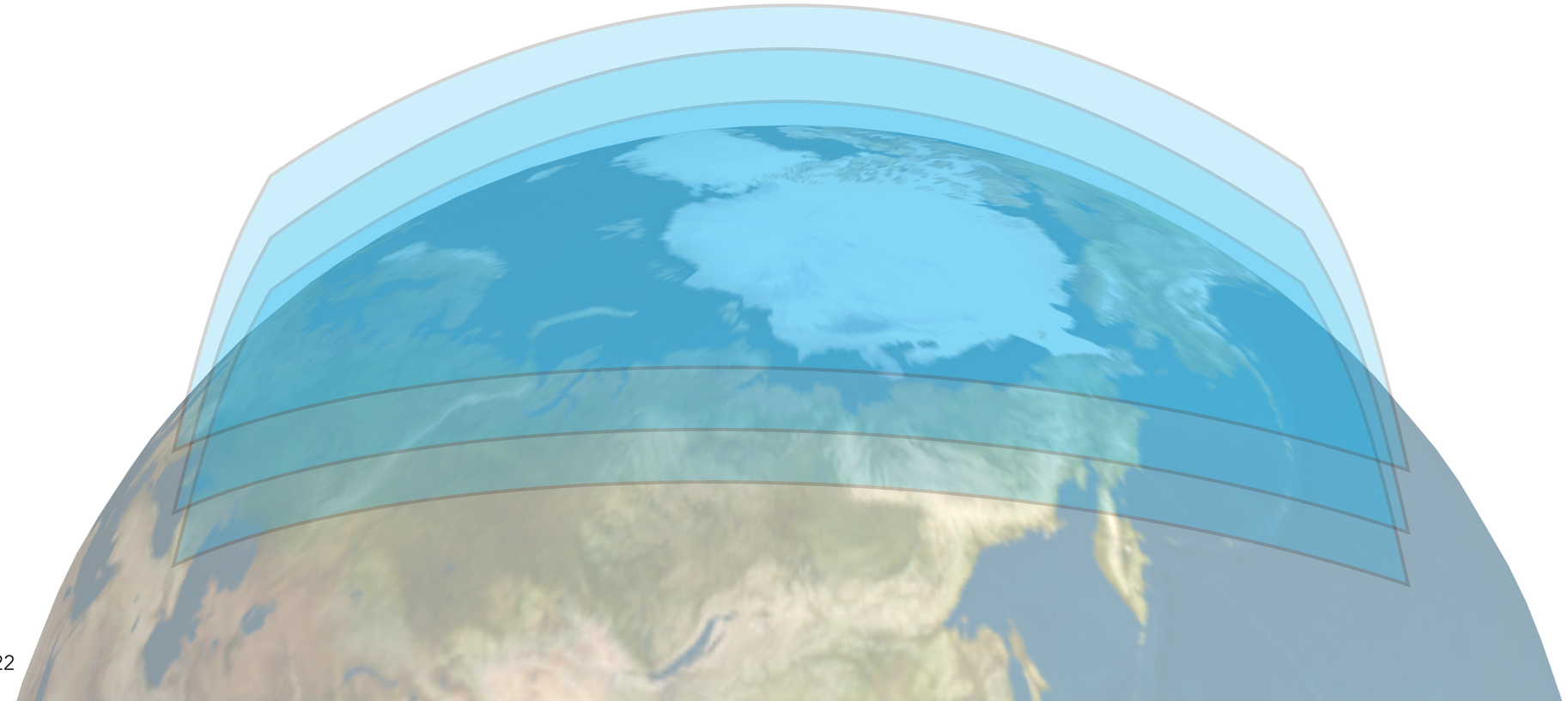
Data: ERA5 reanalysis



Data: ERA5 reanalysis

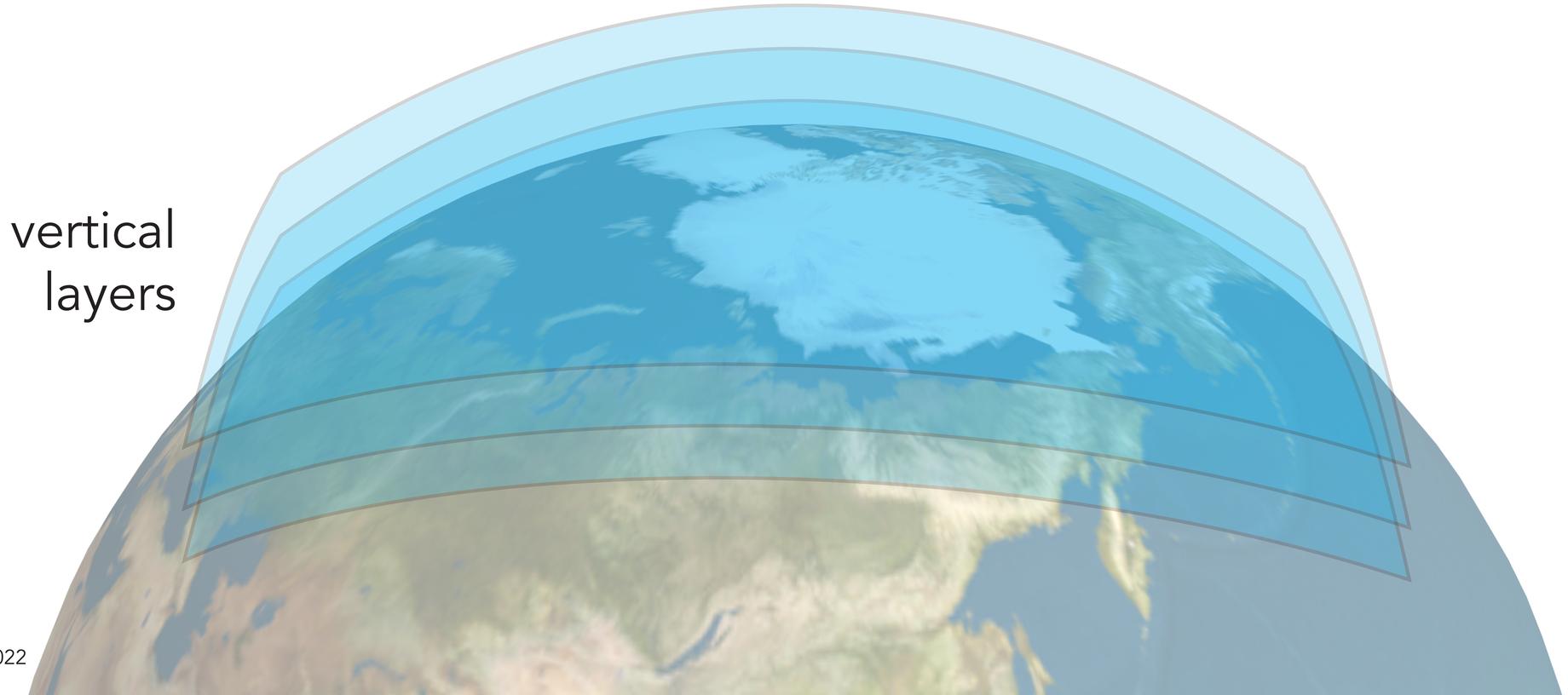


Data: ERA5 reanalysis



Data: ERA5 reanalysis

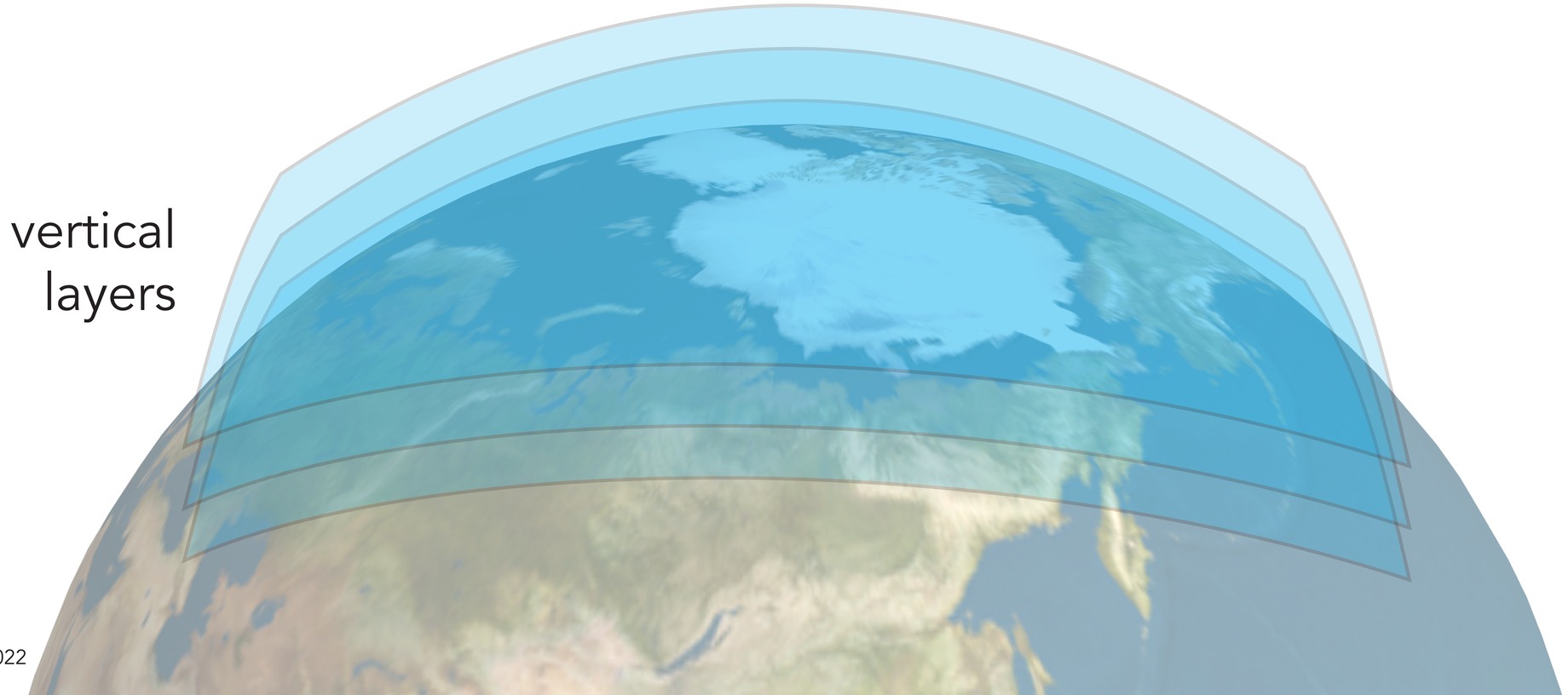
137 vertical layers



Data: ERA5 reanalysis

721x1440 horizontal grid (0.25 degree)

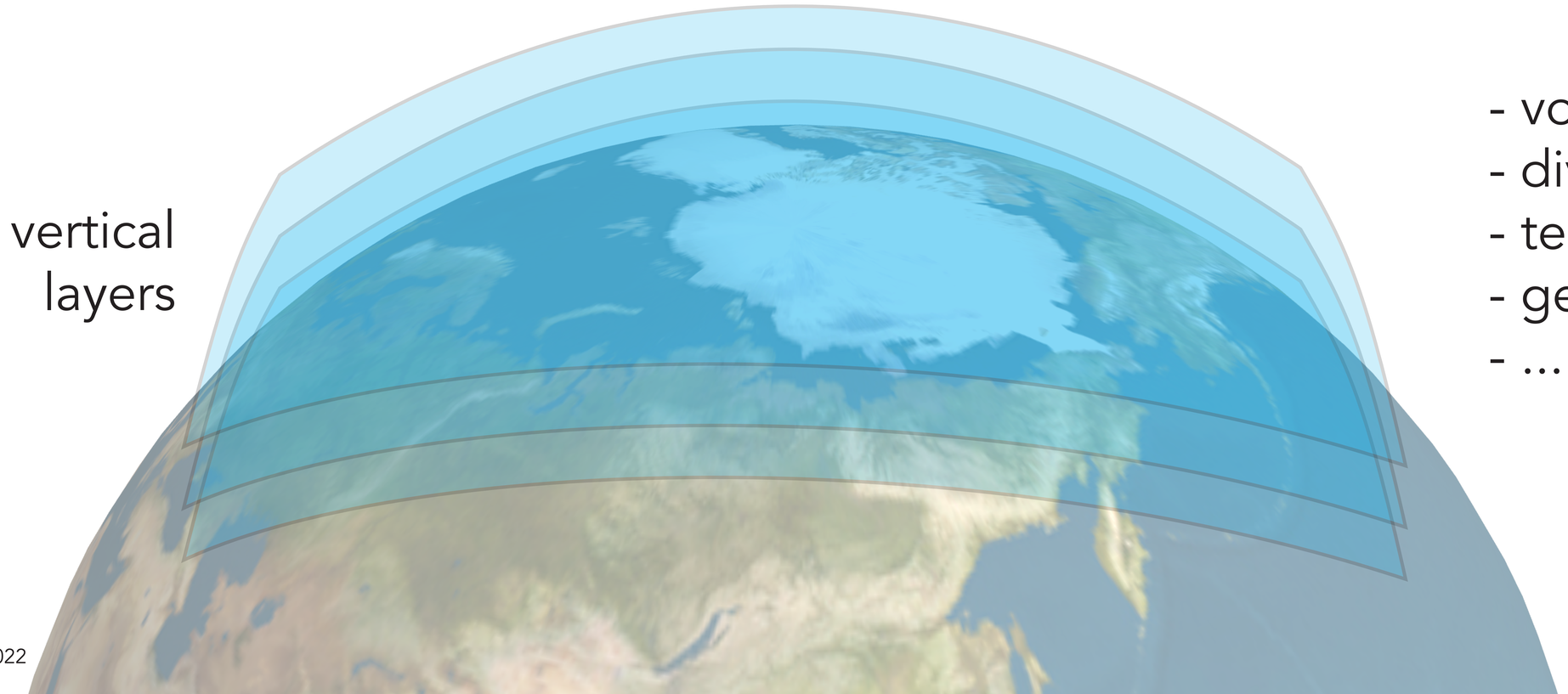
137 vertical layers



Data: ERA5 reanalysis

721x1440 horizontal grid (0.25 degree)

137 vertical layers

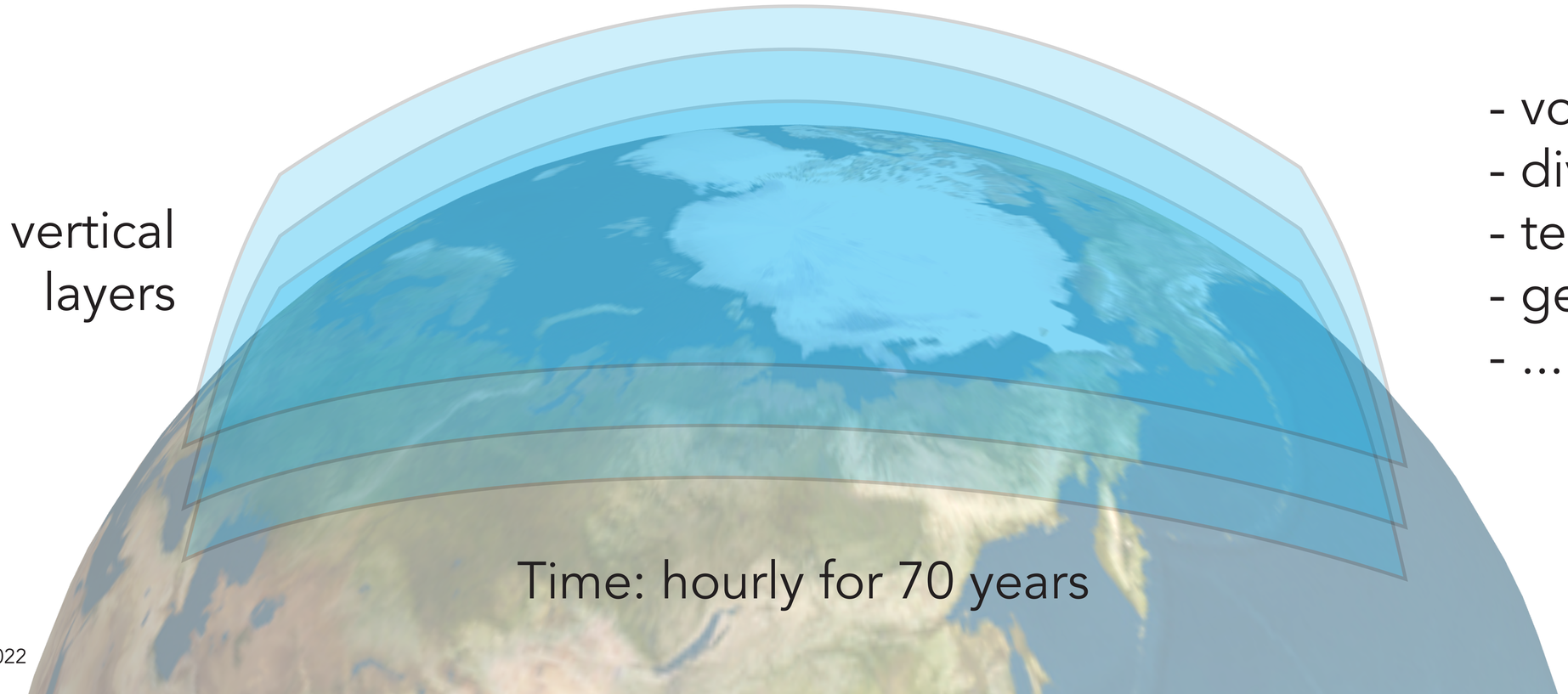


- vorticity
- divergence
- temperature
- geopotential
- ...

Data: ERA5 reanalysis

721x1440 horizontal grid (0.25 degree)

137 vertical layers



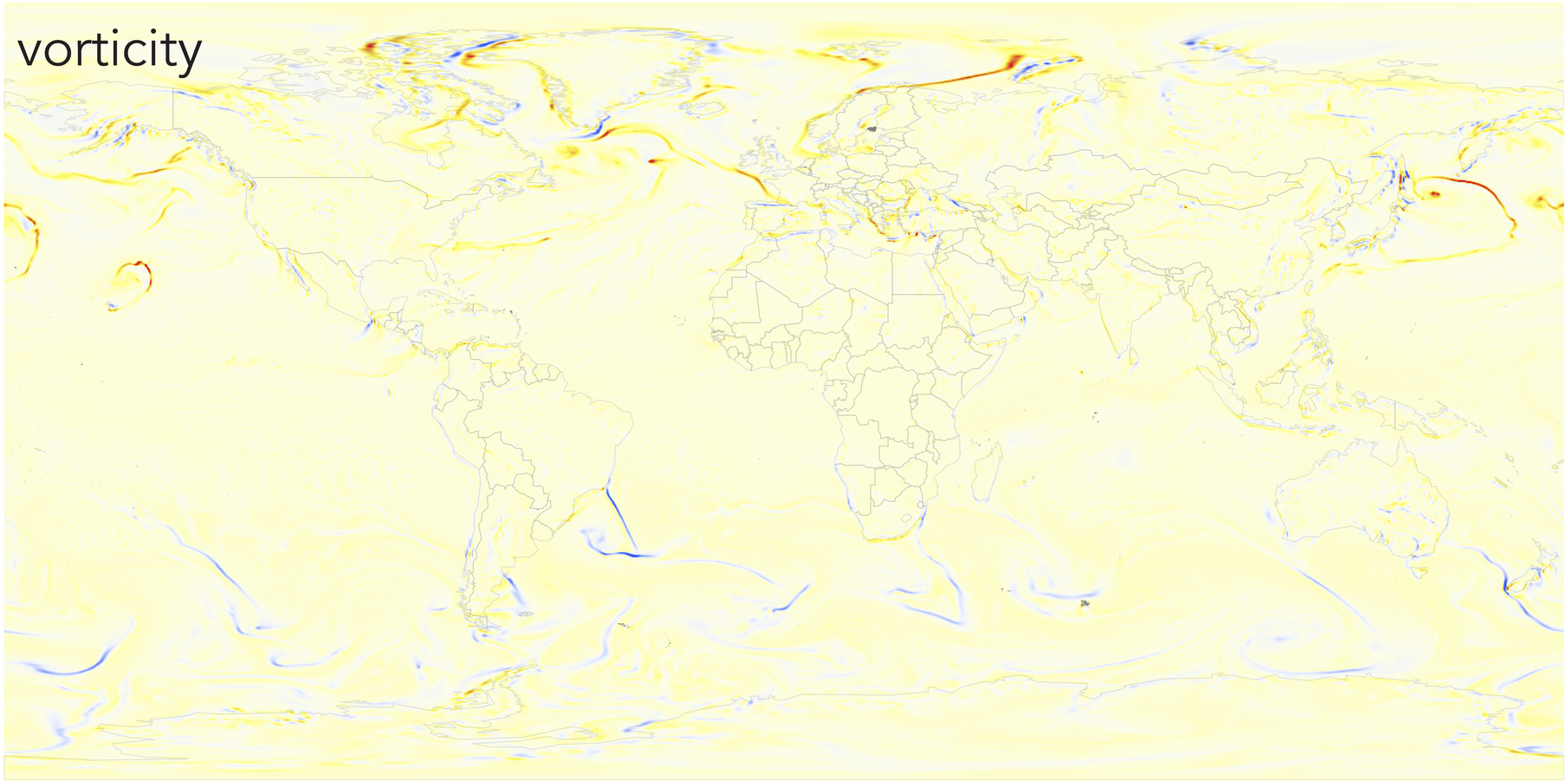
- vorticity
- divergence
- temperature
- geopotential
- ...

Time: hourly for 70 years

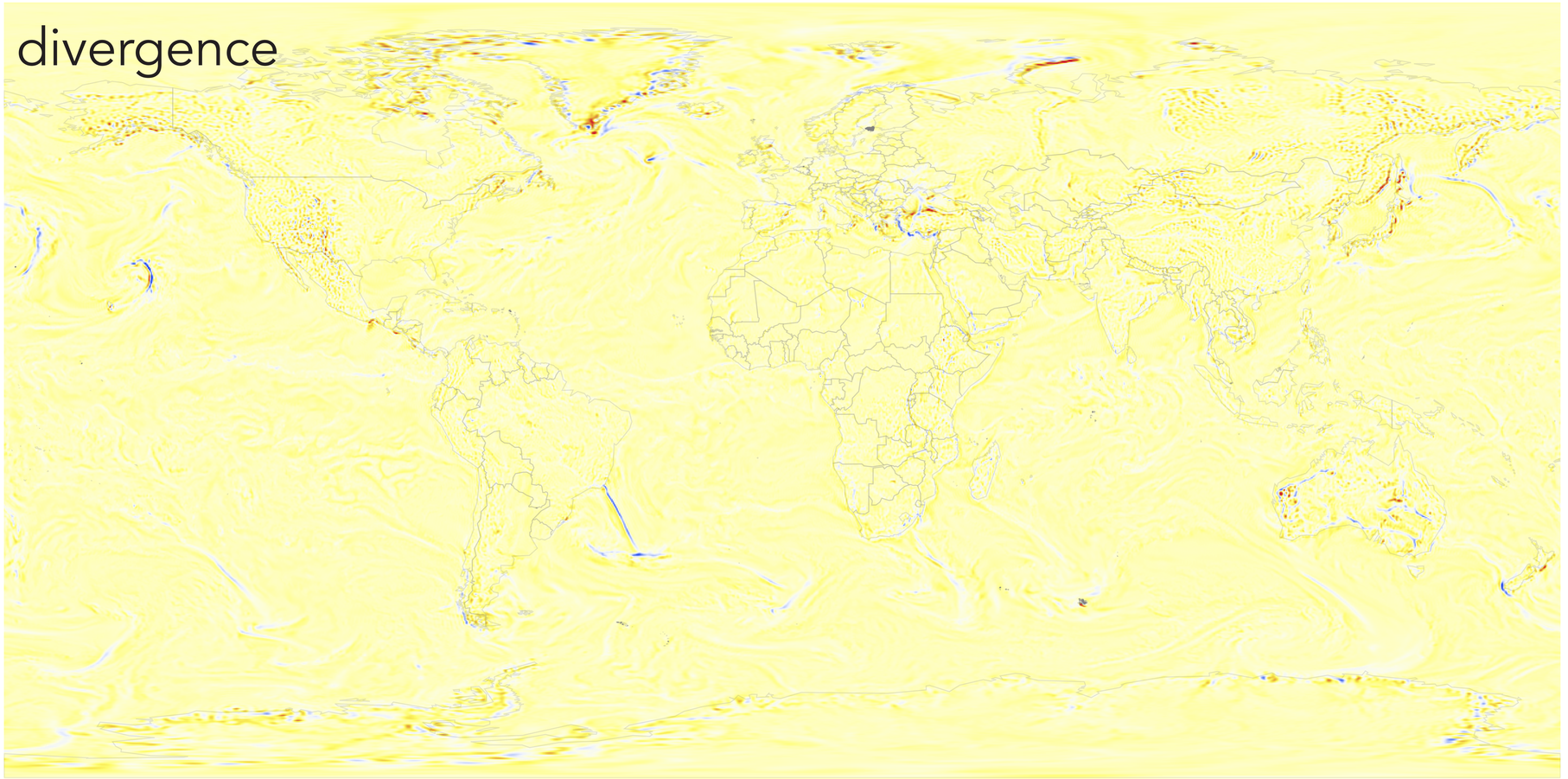
AtmoRep data

- 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

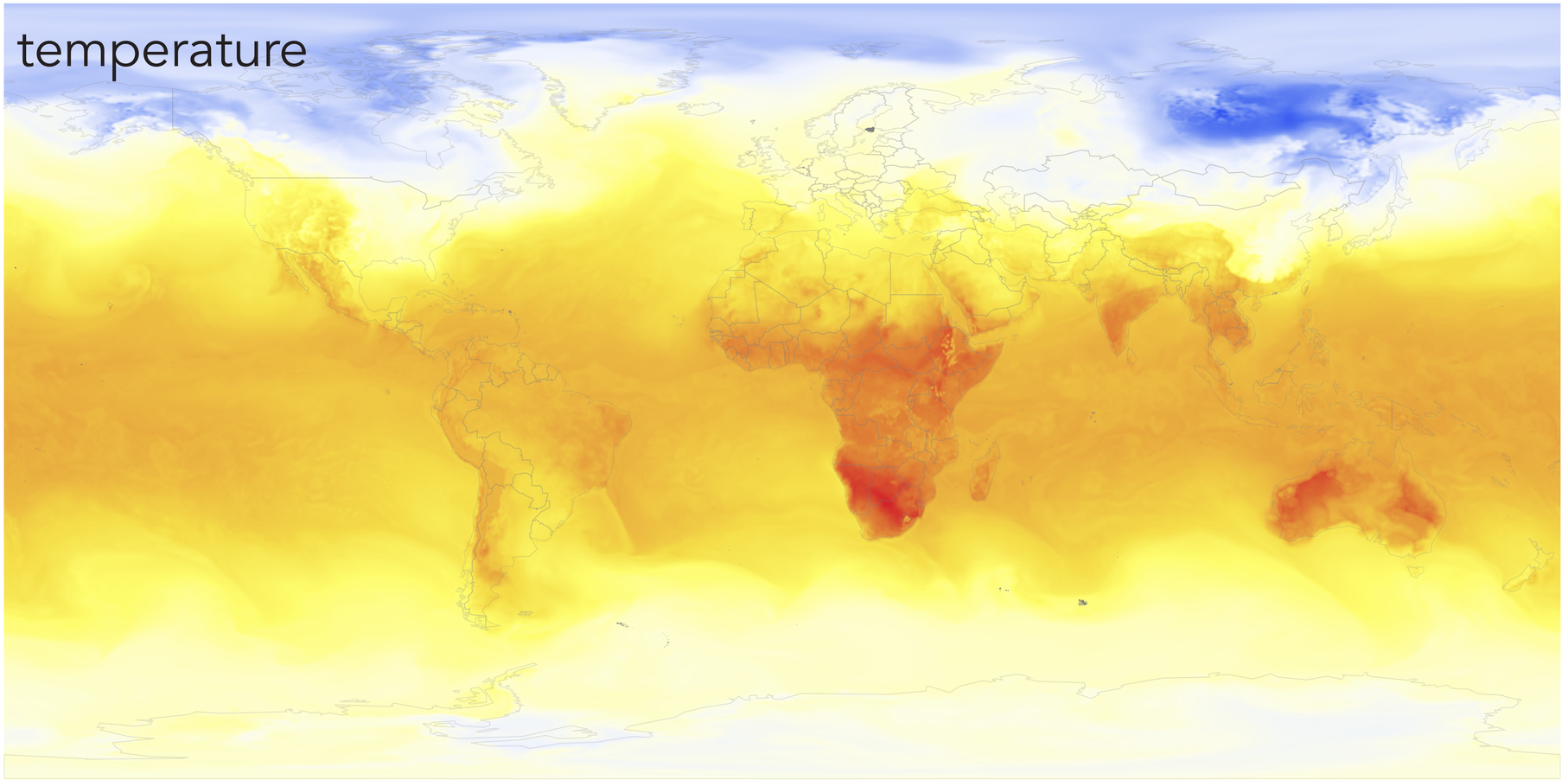
AtmoRep data



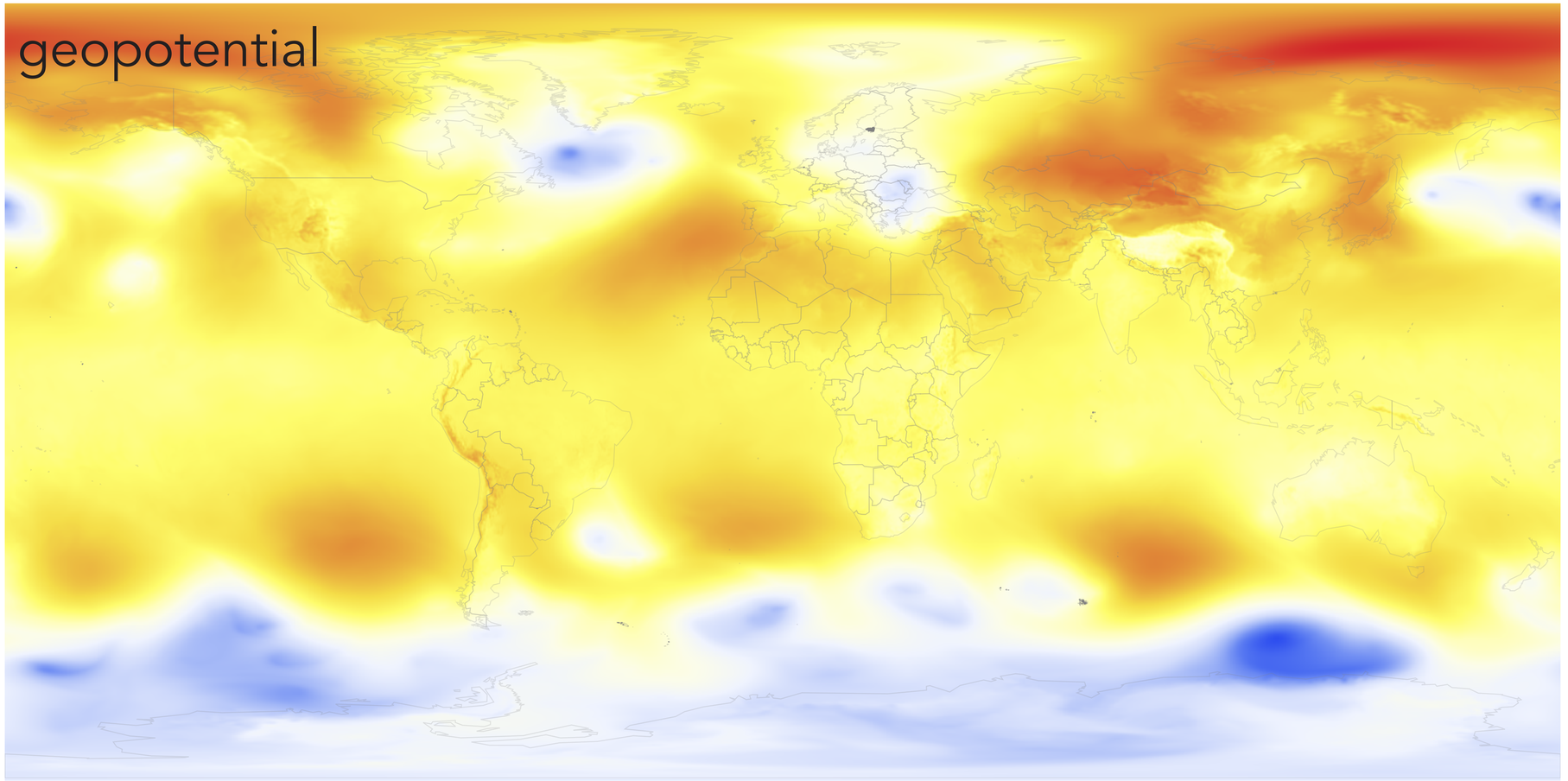
AtmoRep data



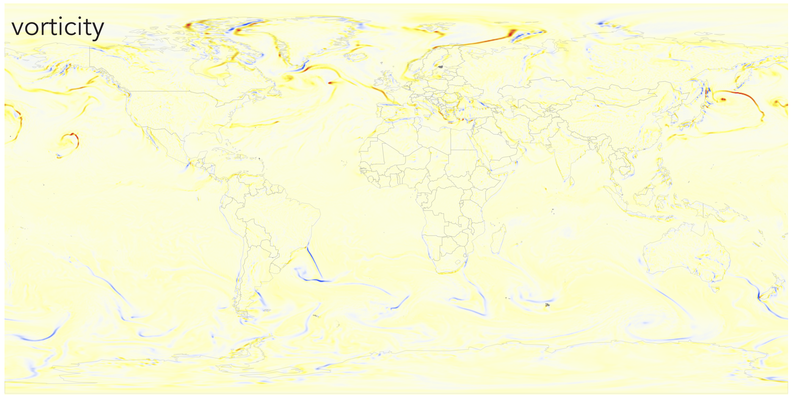
AtmoRep data

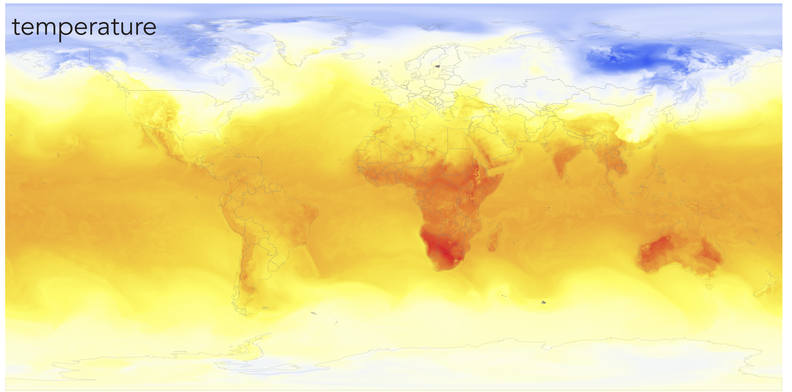


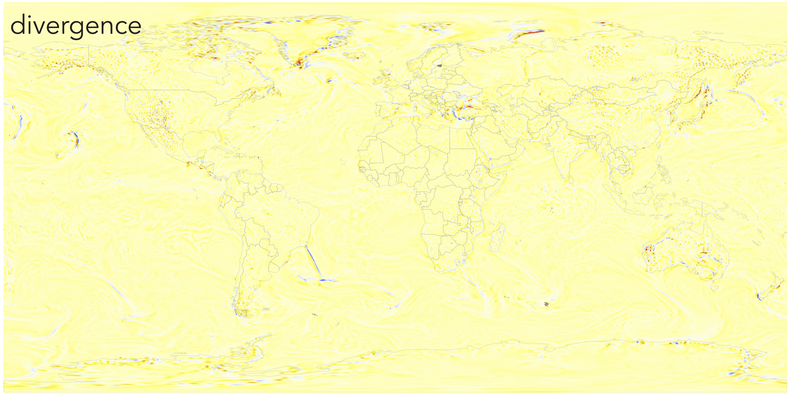
AtmoRep data

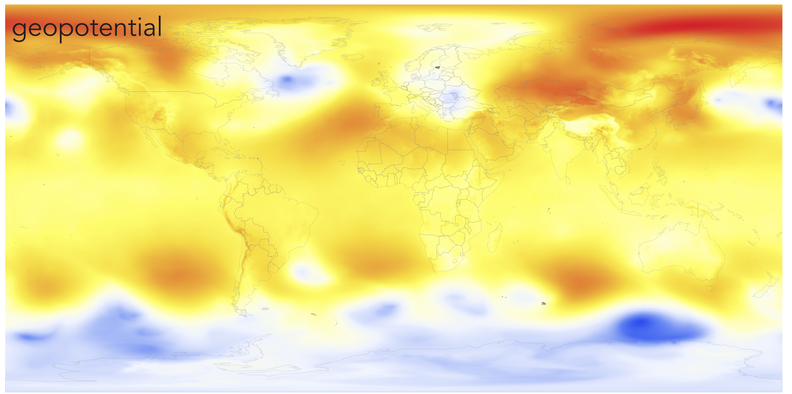


AtmoRep data

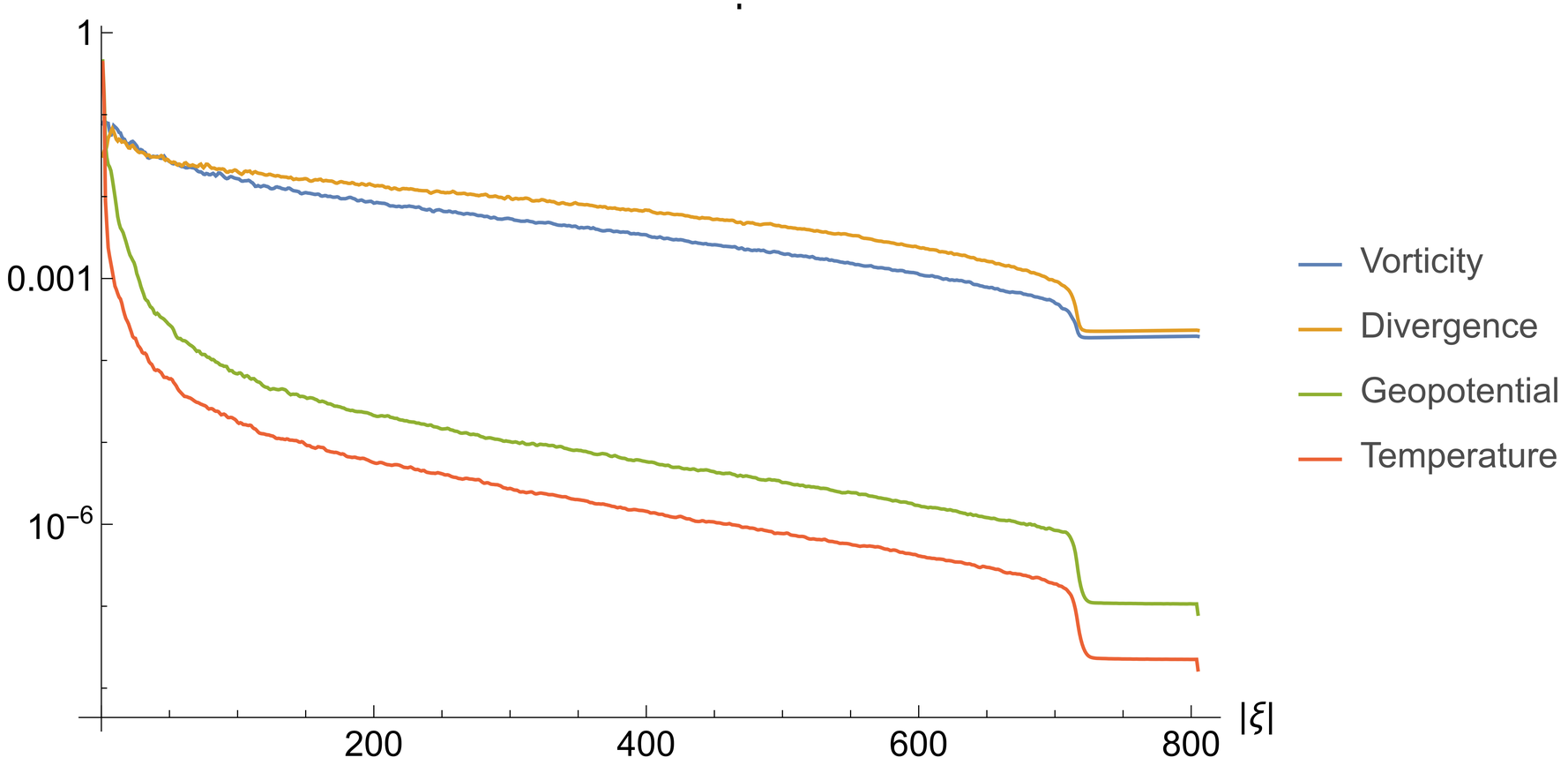




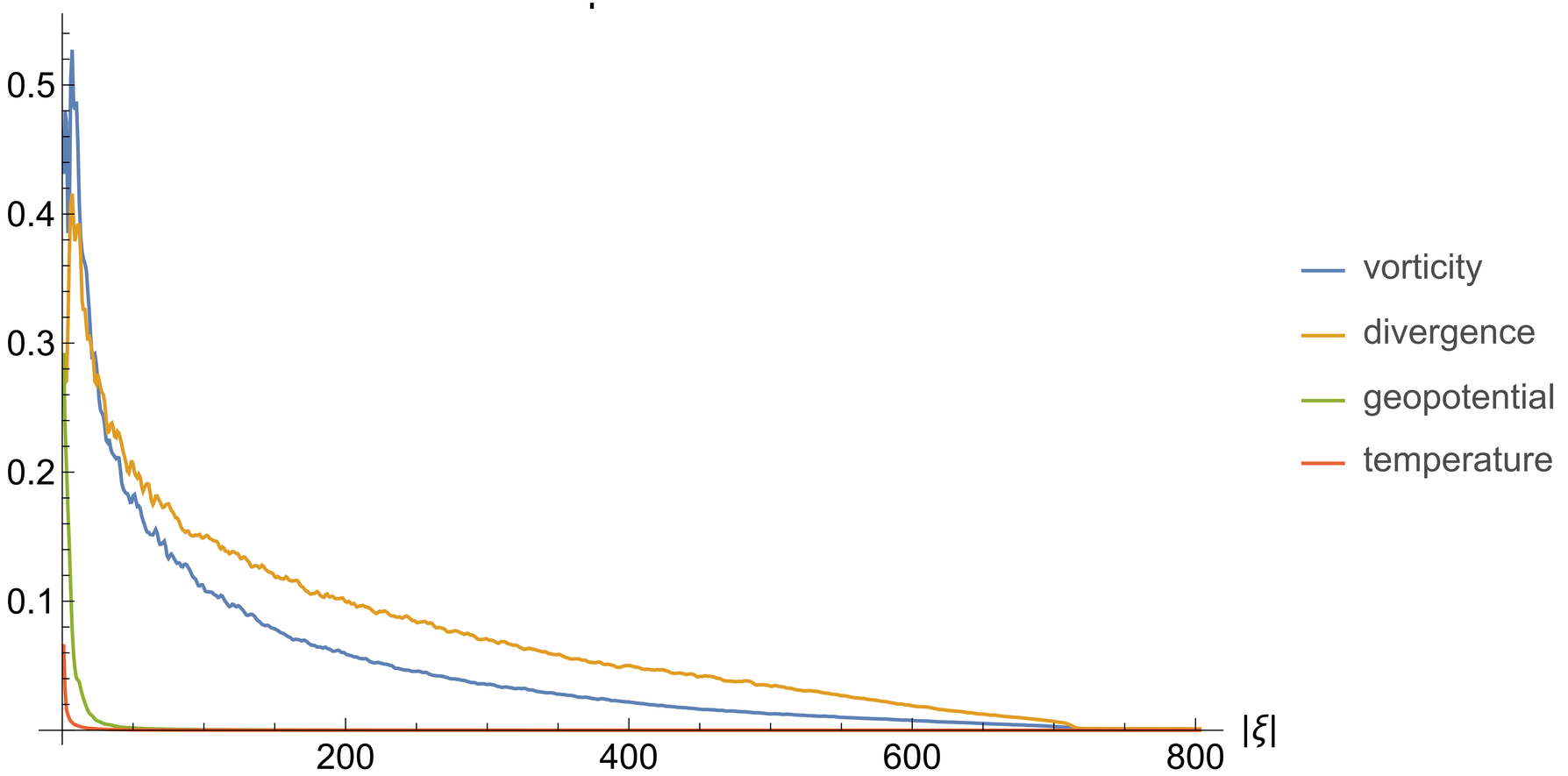




AtmoRep data



AtmoRep data



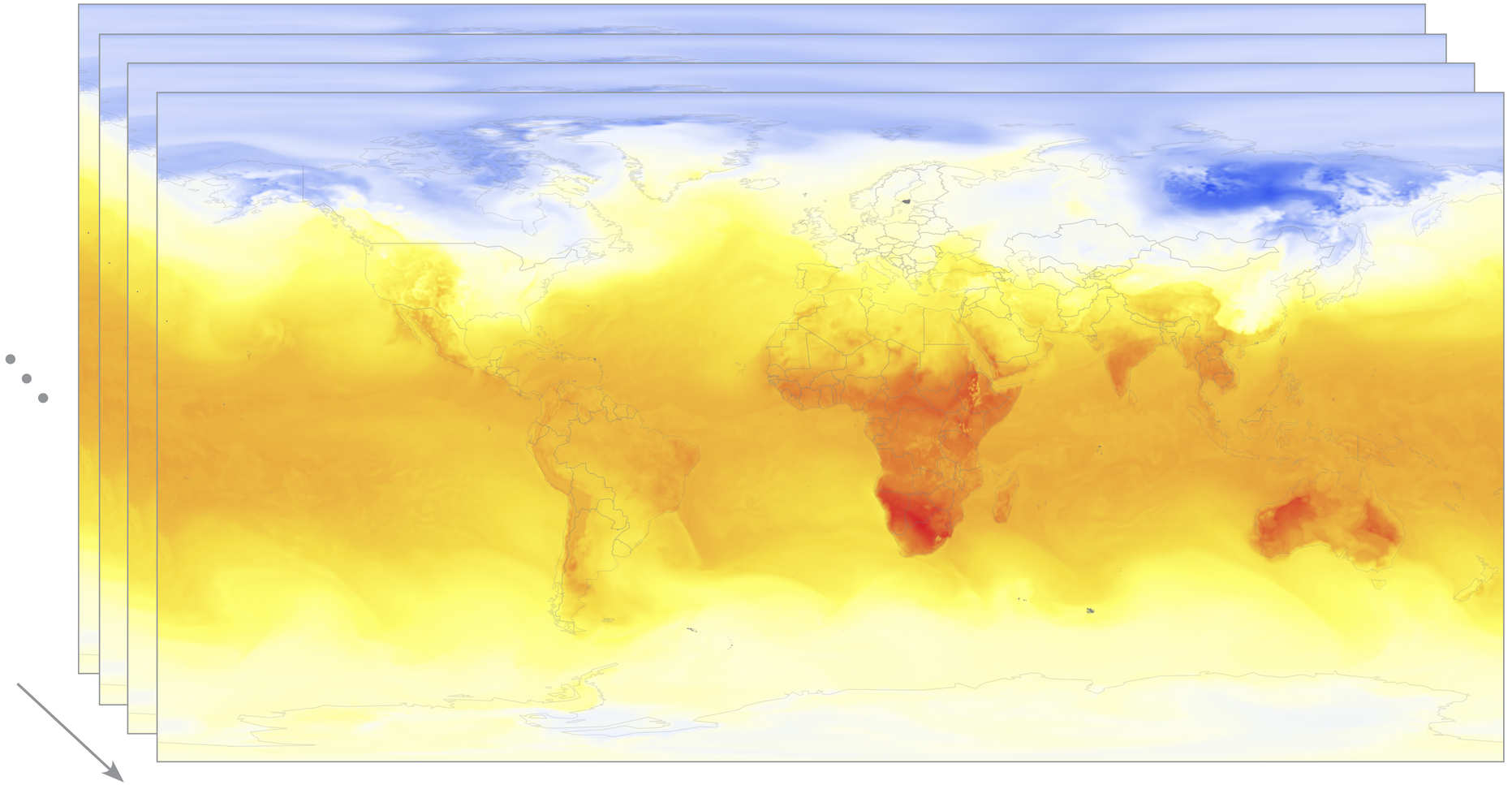
AtmoRep network architecture

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

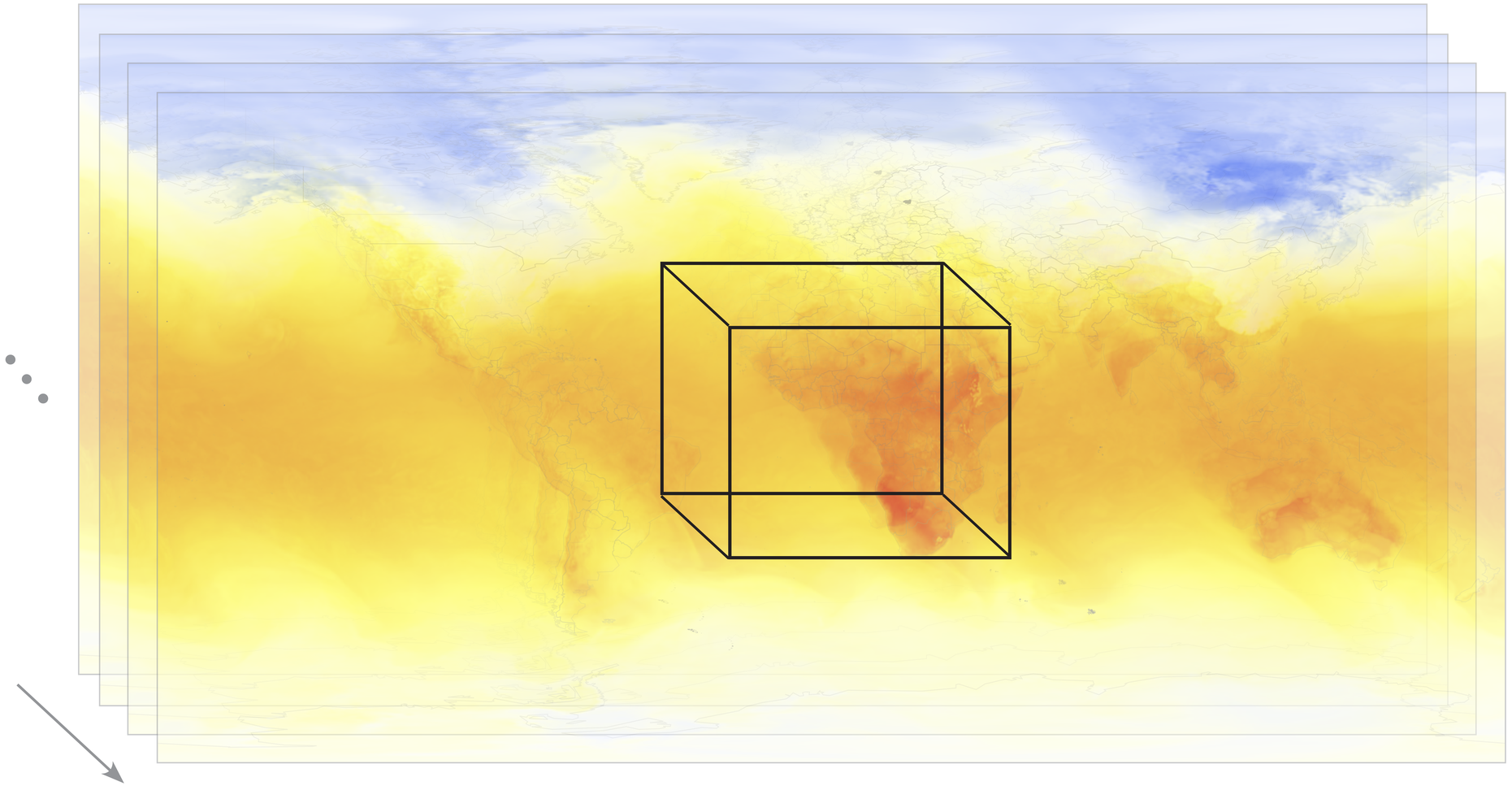
AtmoRep network architecture

- 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

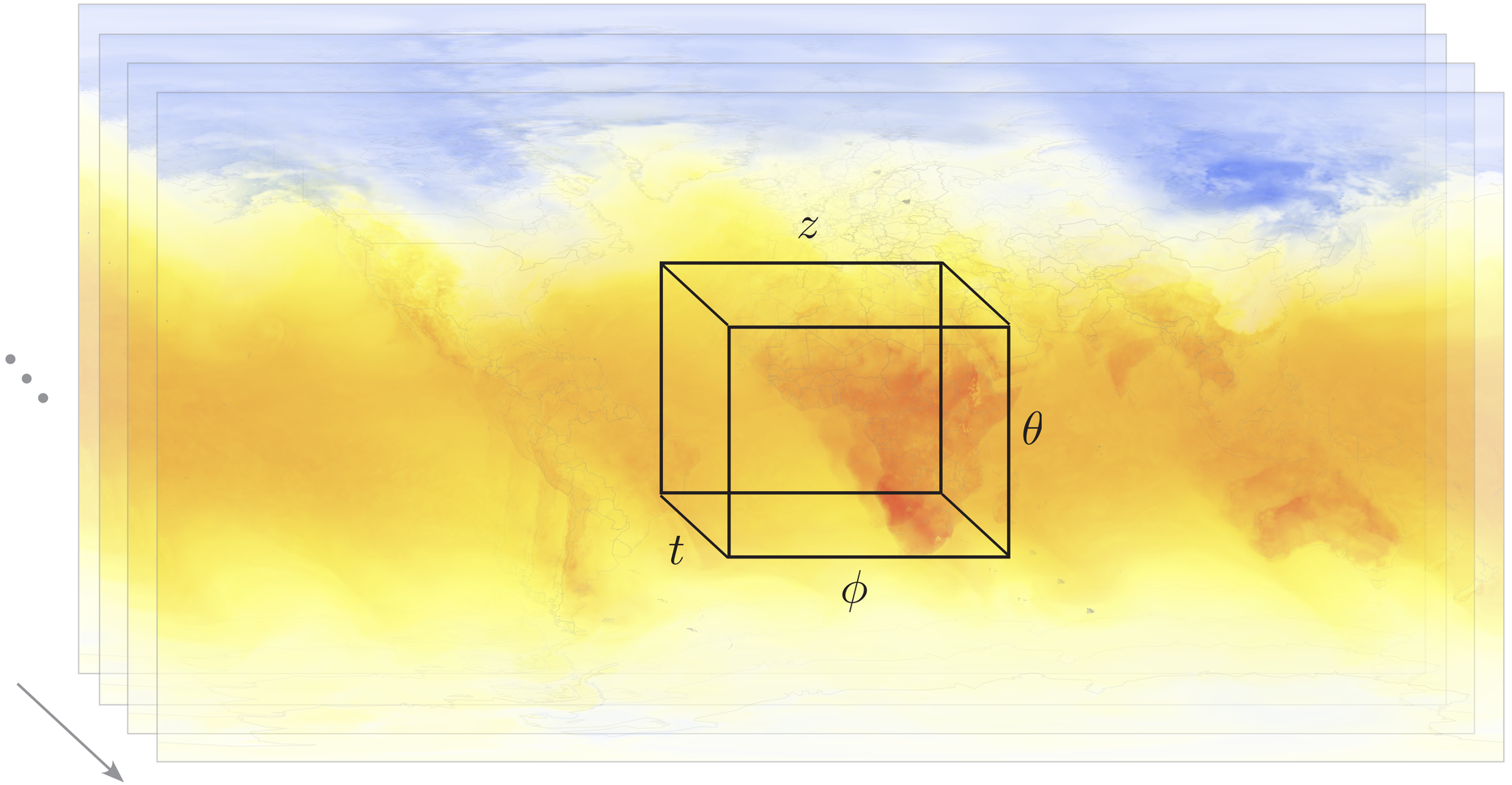
AtmoRep network architecture



AtmoRep network architecture



AtmoRep network architecture

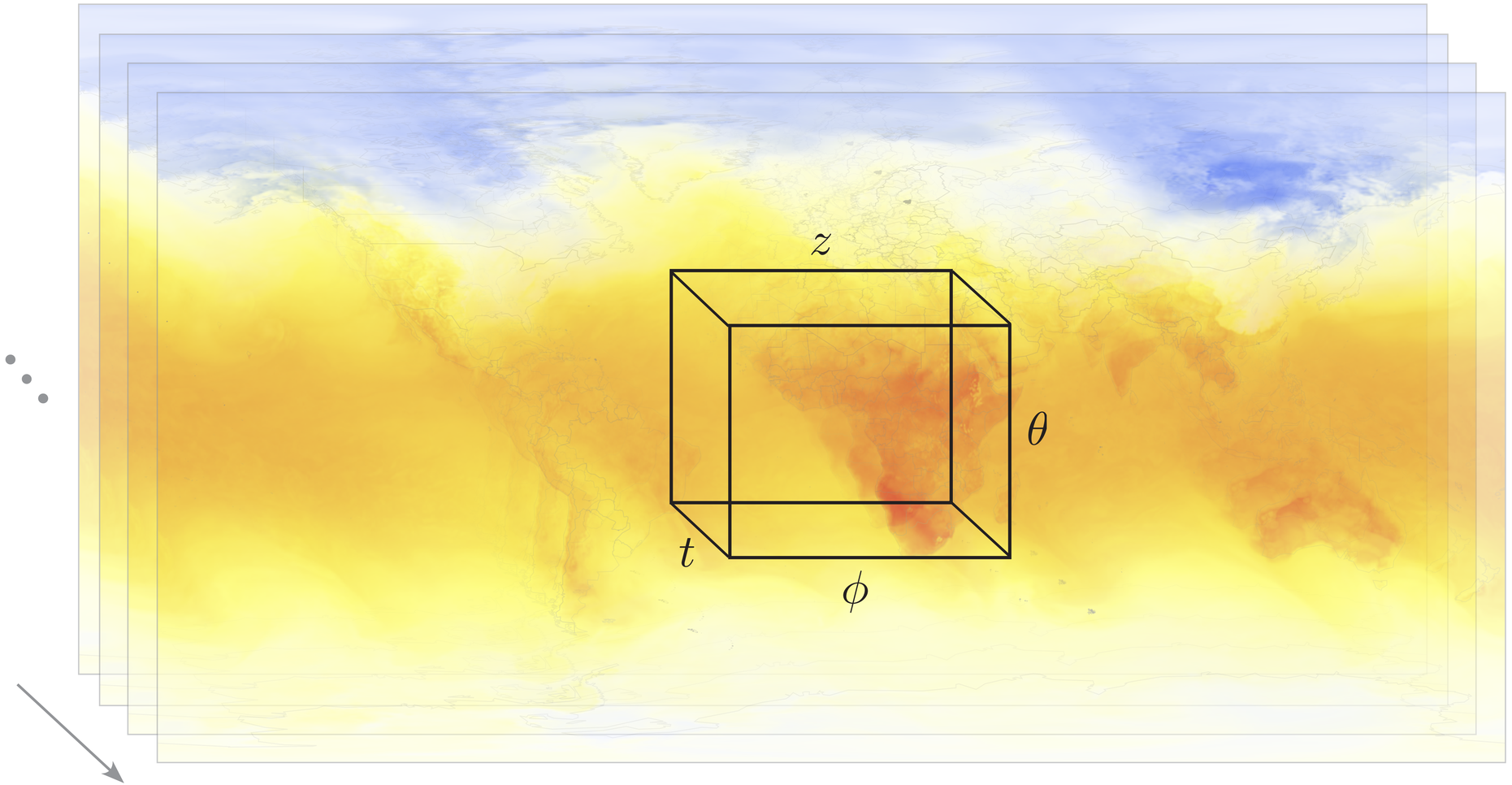


AtmoRep network architecture

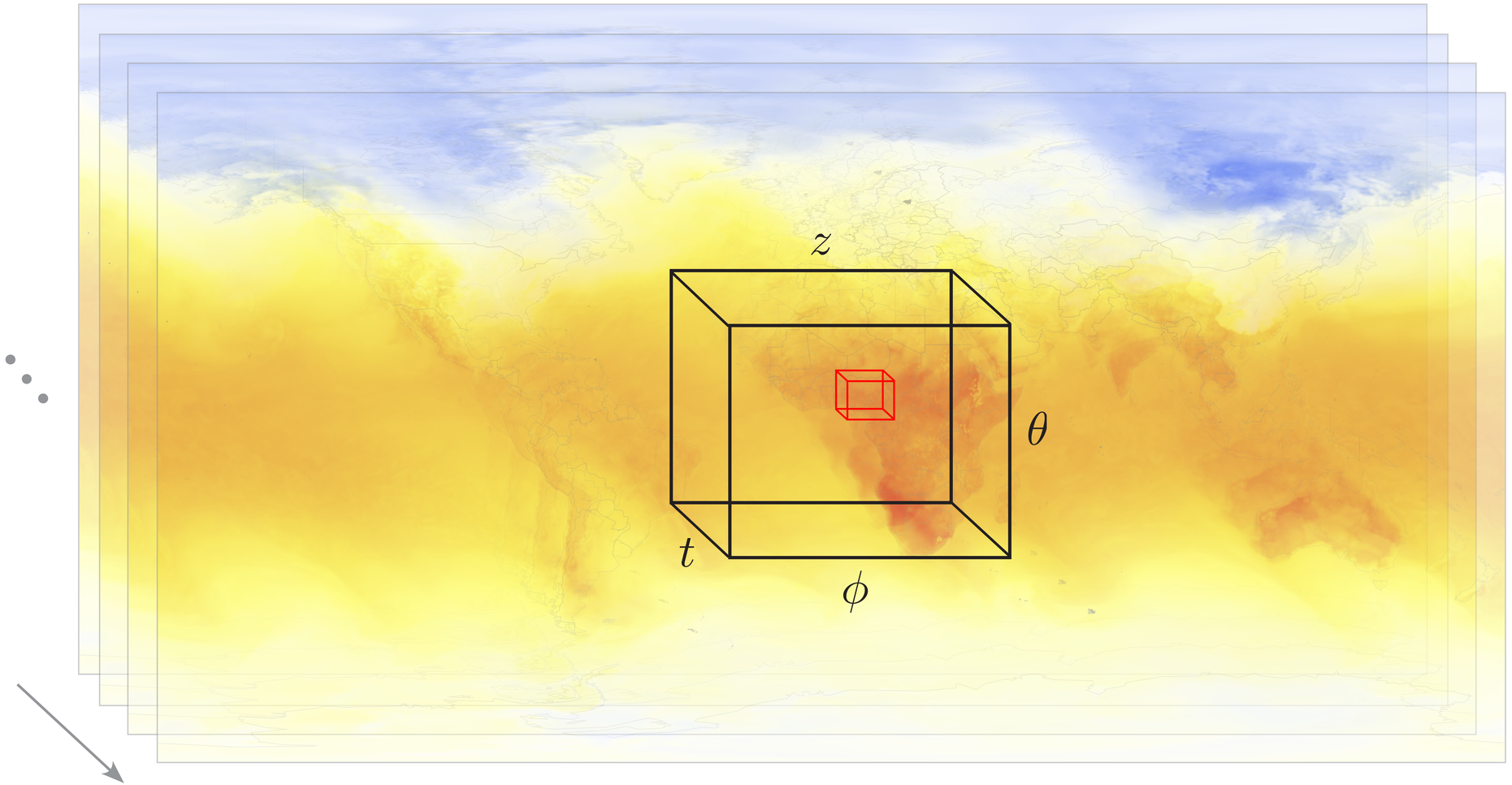
- Transformer encoder-based network architecture
  - Scales well to very large data-sets
  - Generative model (with decoder)
  - Attention maps provide (physical) interpretability
- Network is local in space-time
  - Principal of dynamics are universally valid
  - Local particularities can be learned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

What is a token?

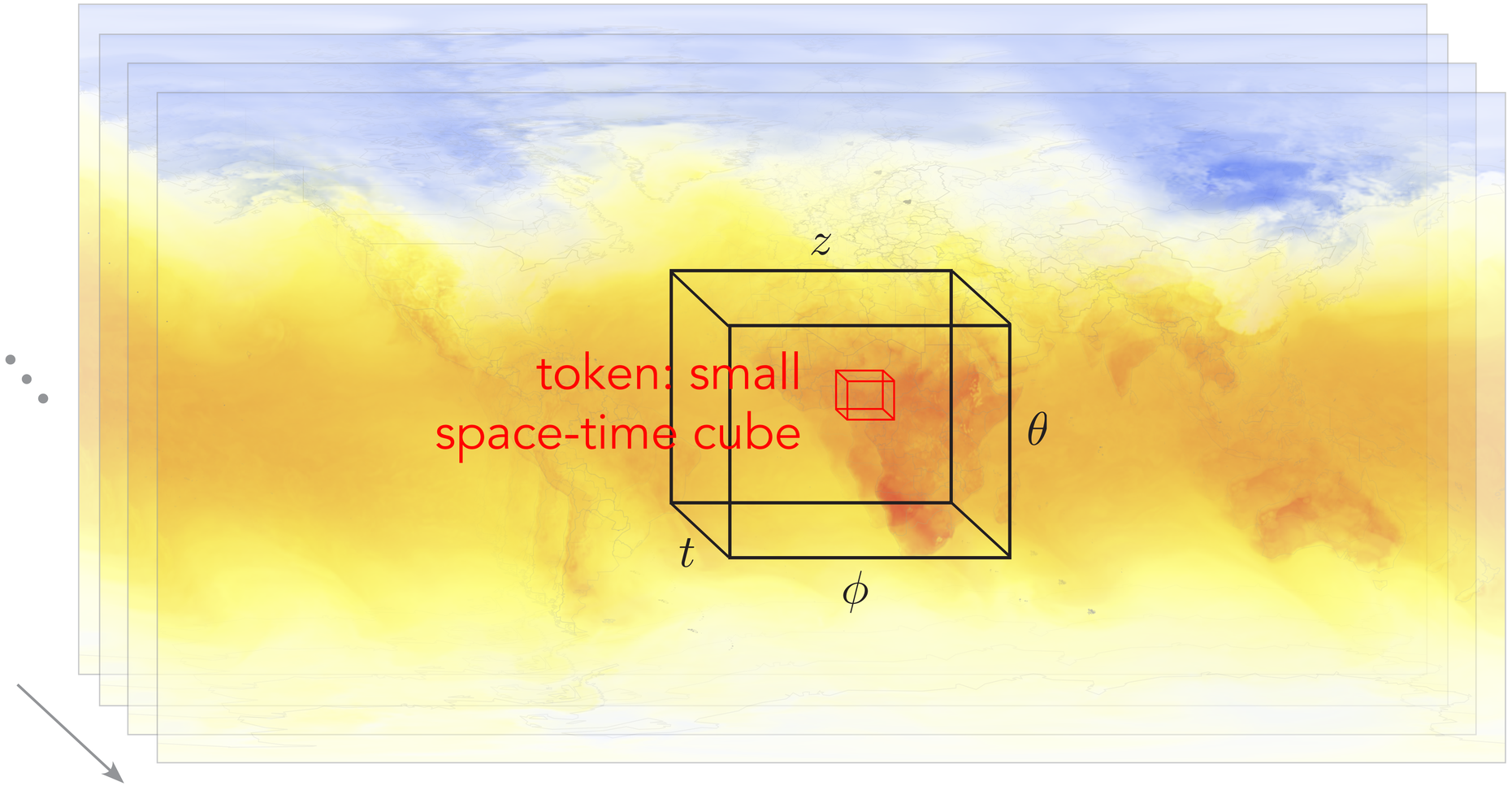
What is a token?



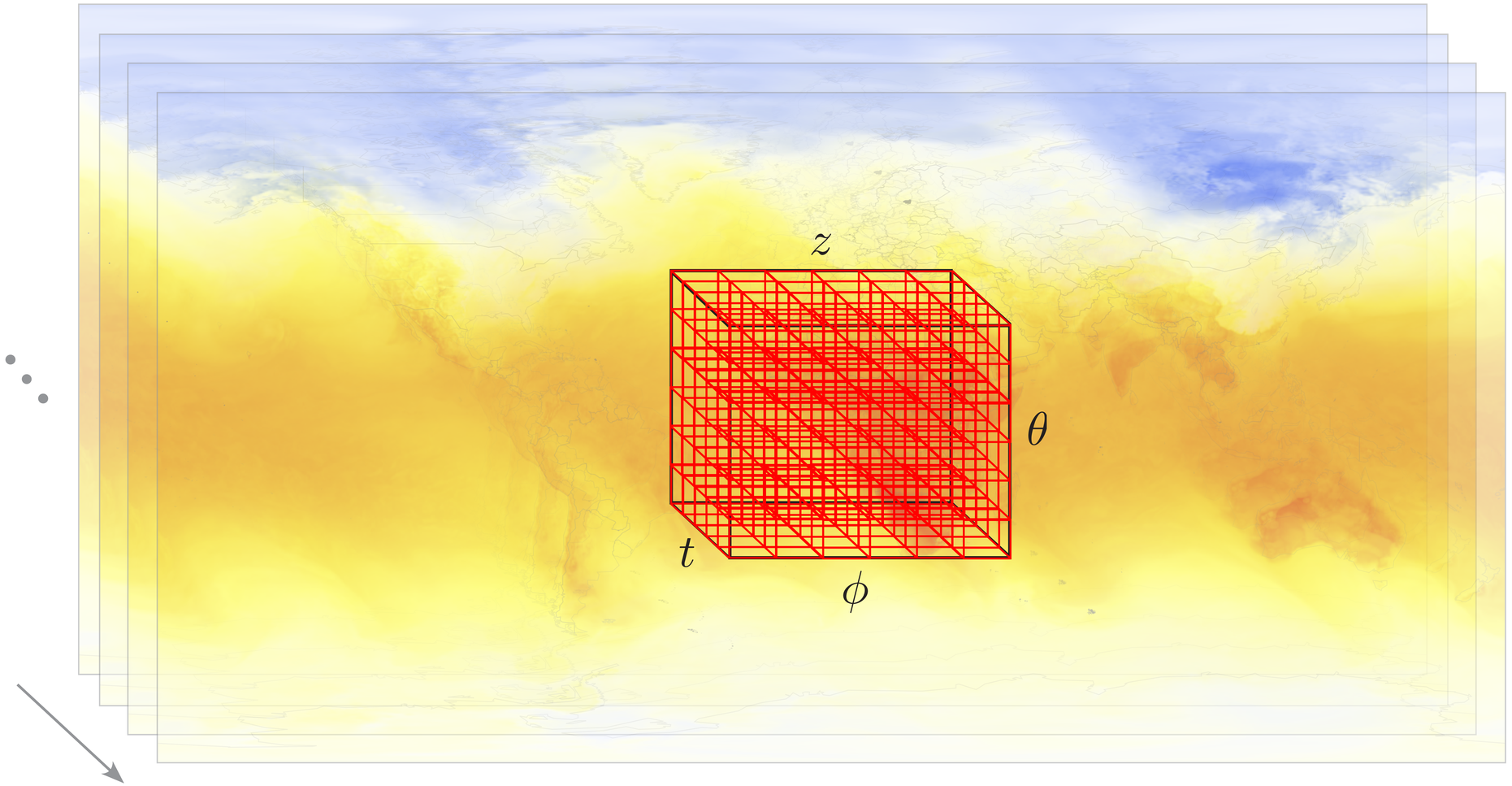
What is a token?



What is a token?

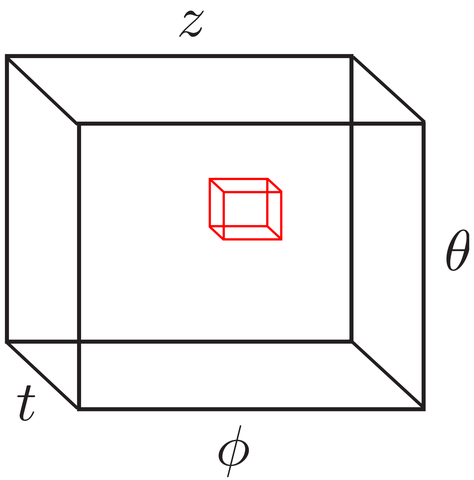


What is a token?



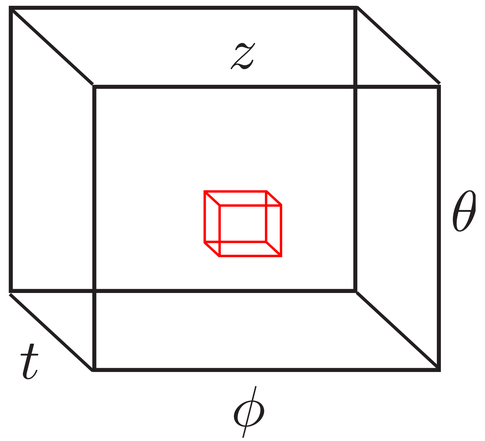
What is a token?

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



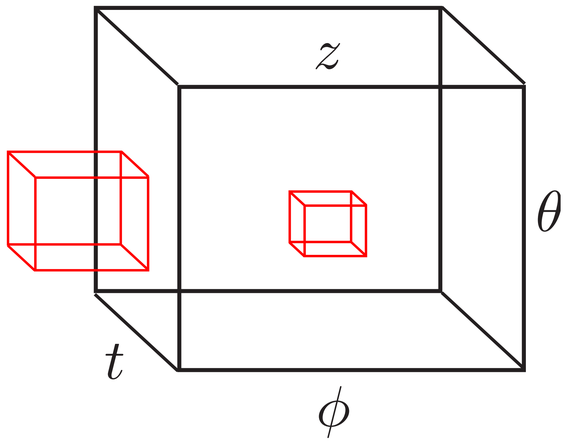
What is a token?

- 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

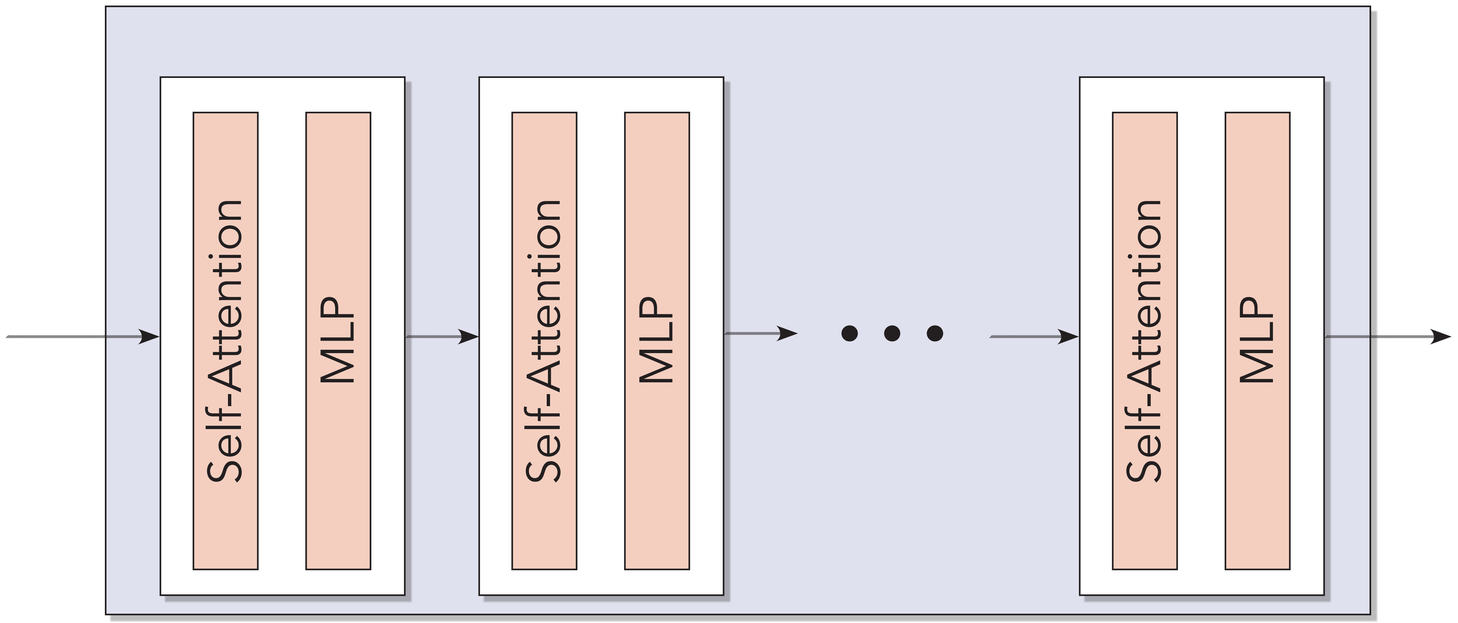


What is a token?

- 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 neighborhood



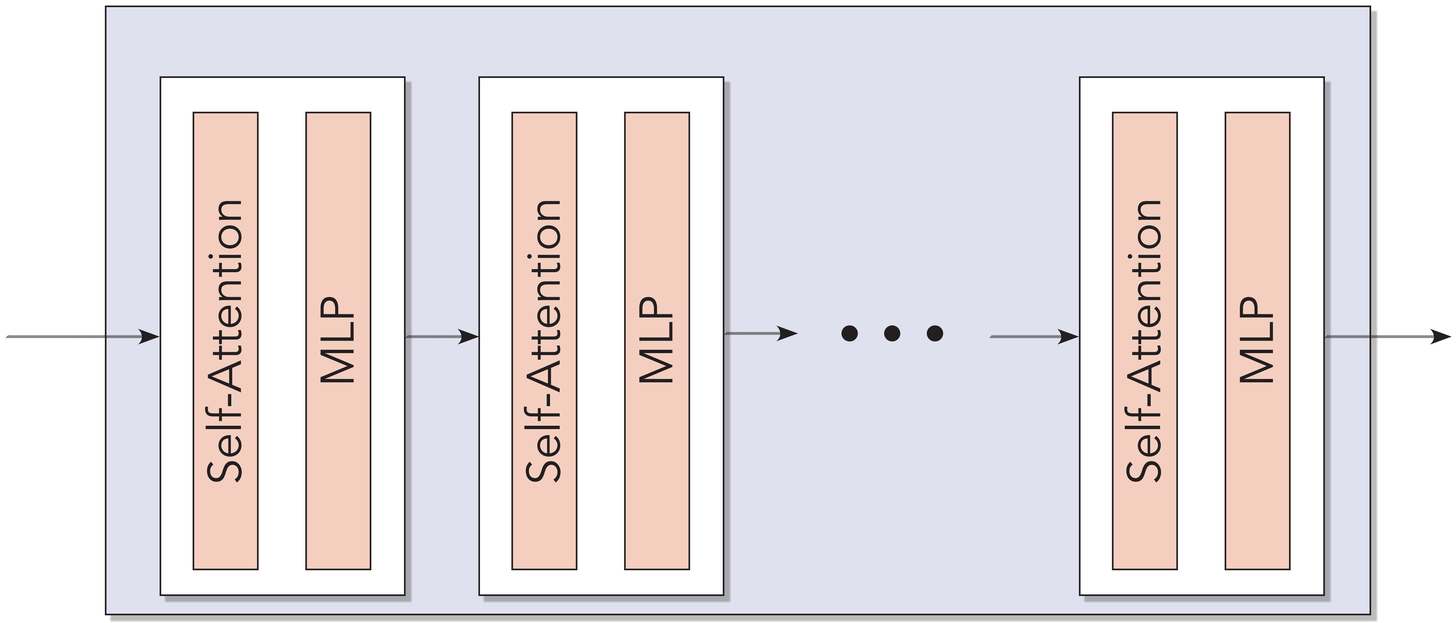
Multiformer: respect the physical fields



Multiformer: respect the physical fields

Self
attention

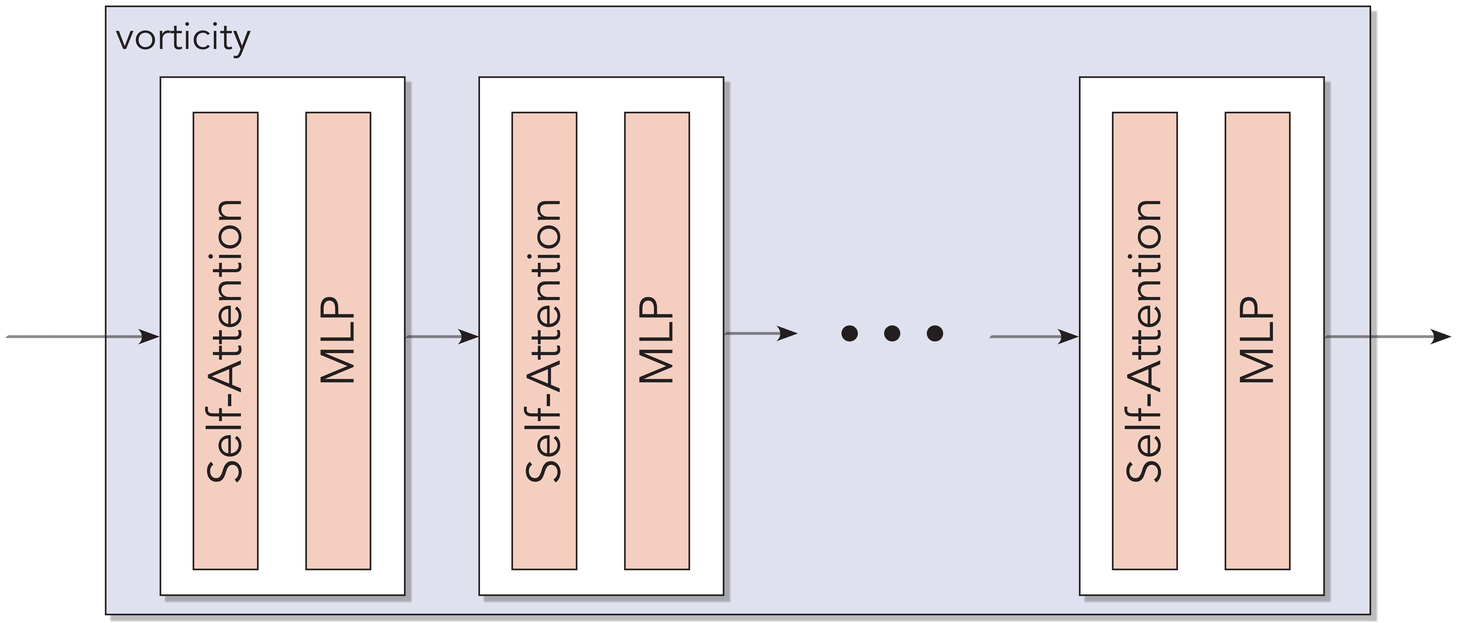
$$\sigma\left(Q\,K^T\right)V$$



Multiformer: respect the physical fields

Self
attention

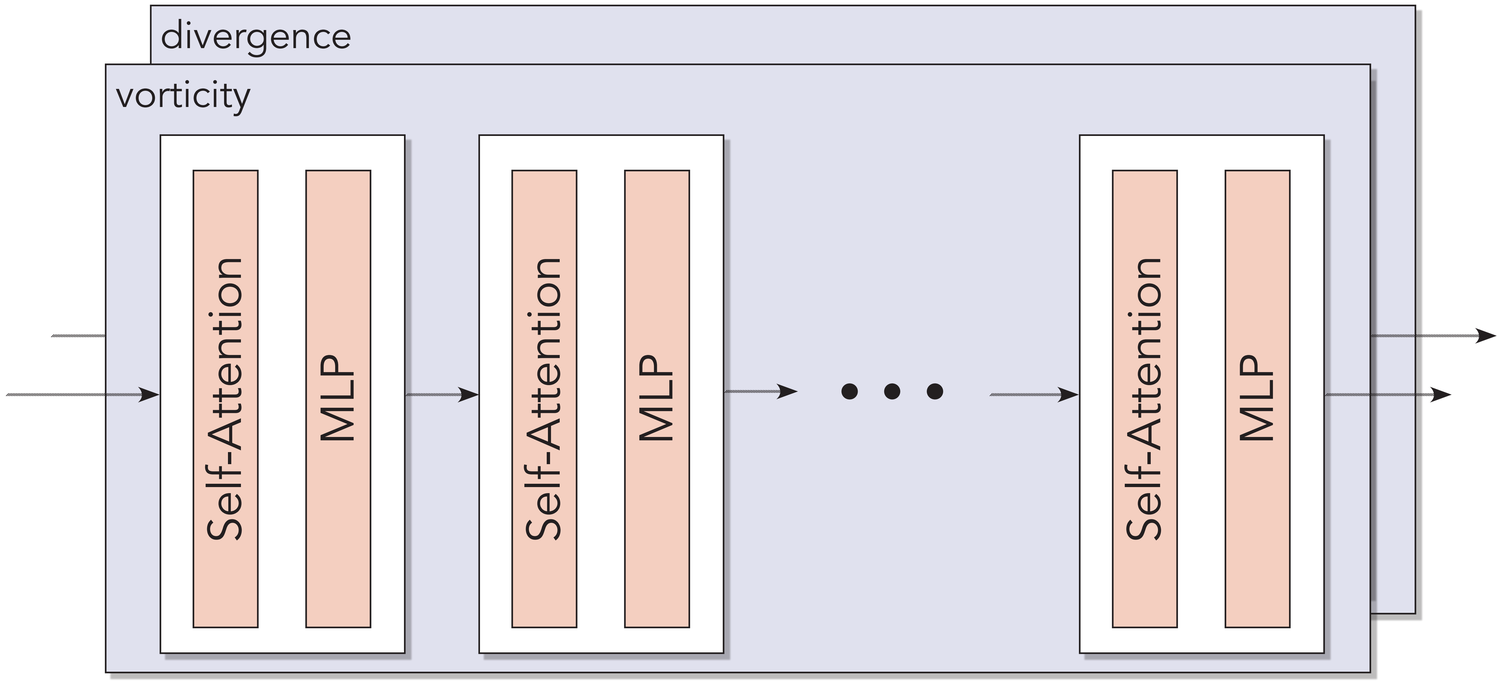
$$\sigma\left(Q\,K^T\right)V$$



Multiformer: respect the physical fields

Self
attention

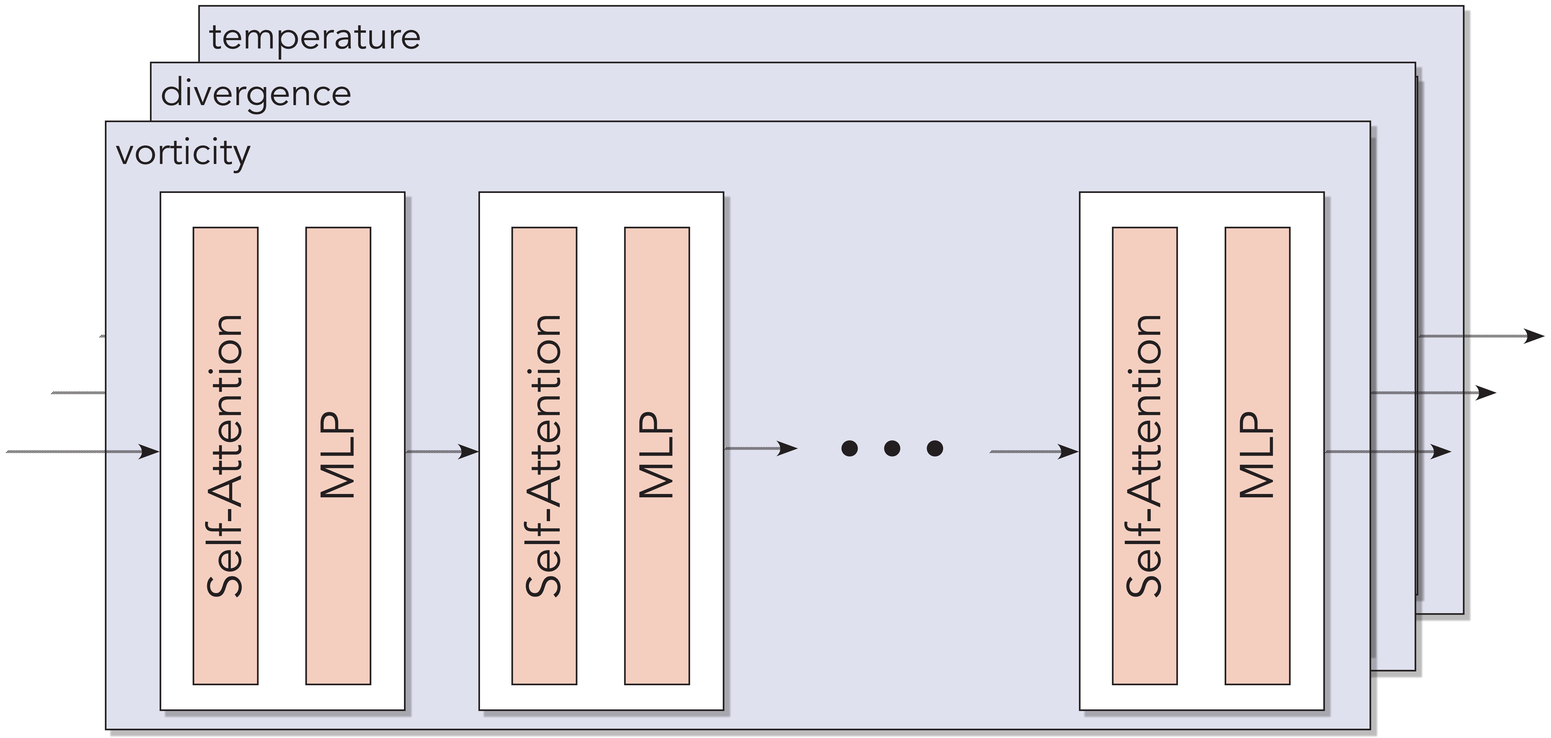
$$\sigma\left(Q\,K^T\right)V$$



Multiformer: respect the physical fields

Self
attention

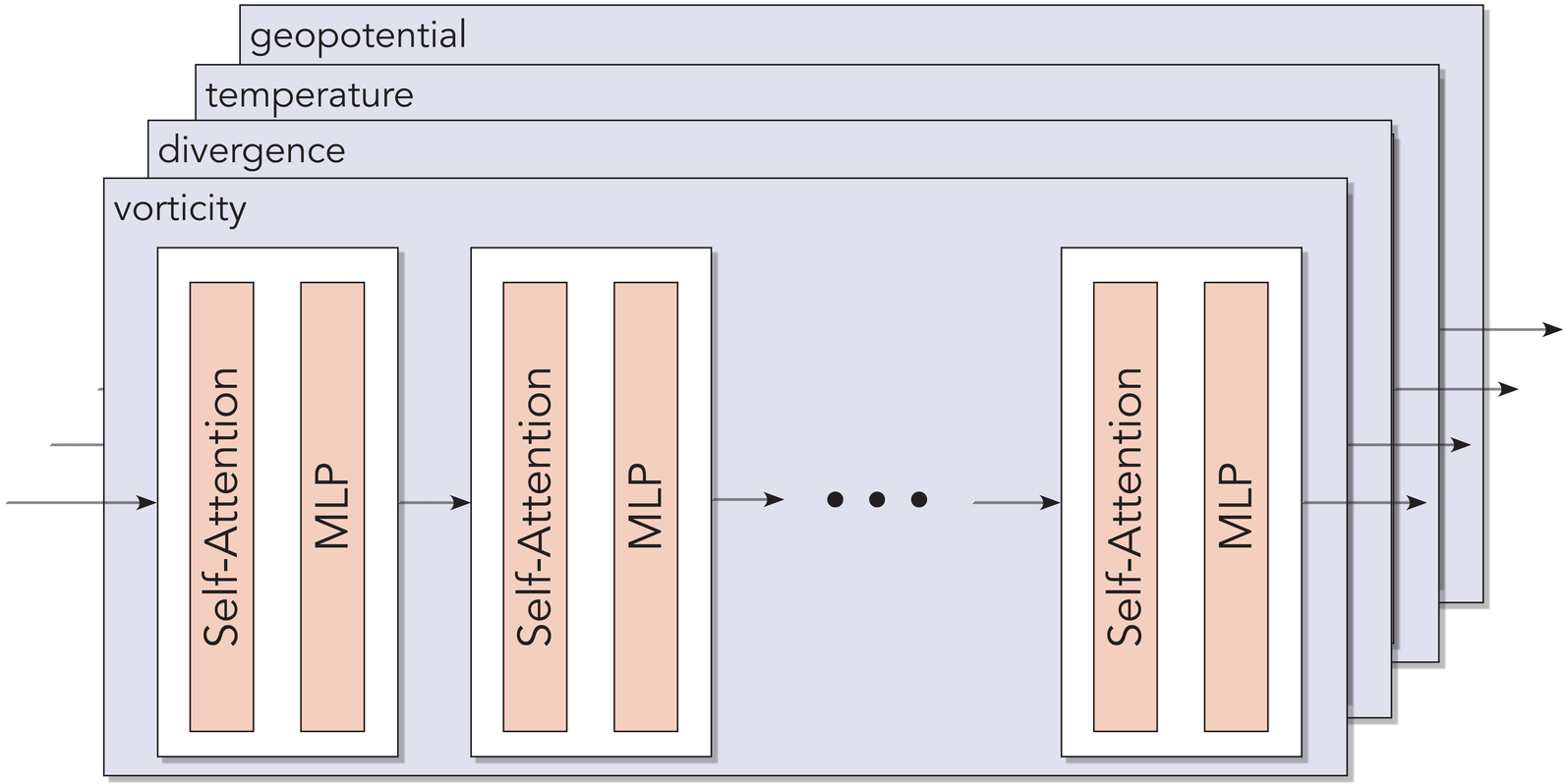
$$\sigma\left(Q\,K^T\right)V$$



Multiformer: respect the physical fields

Self attention

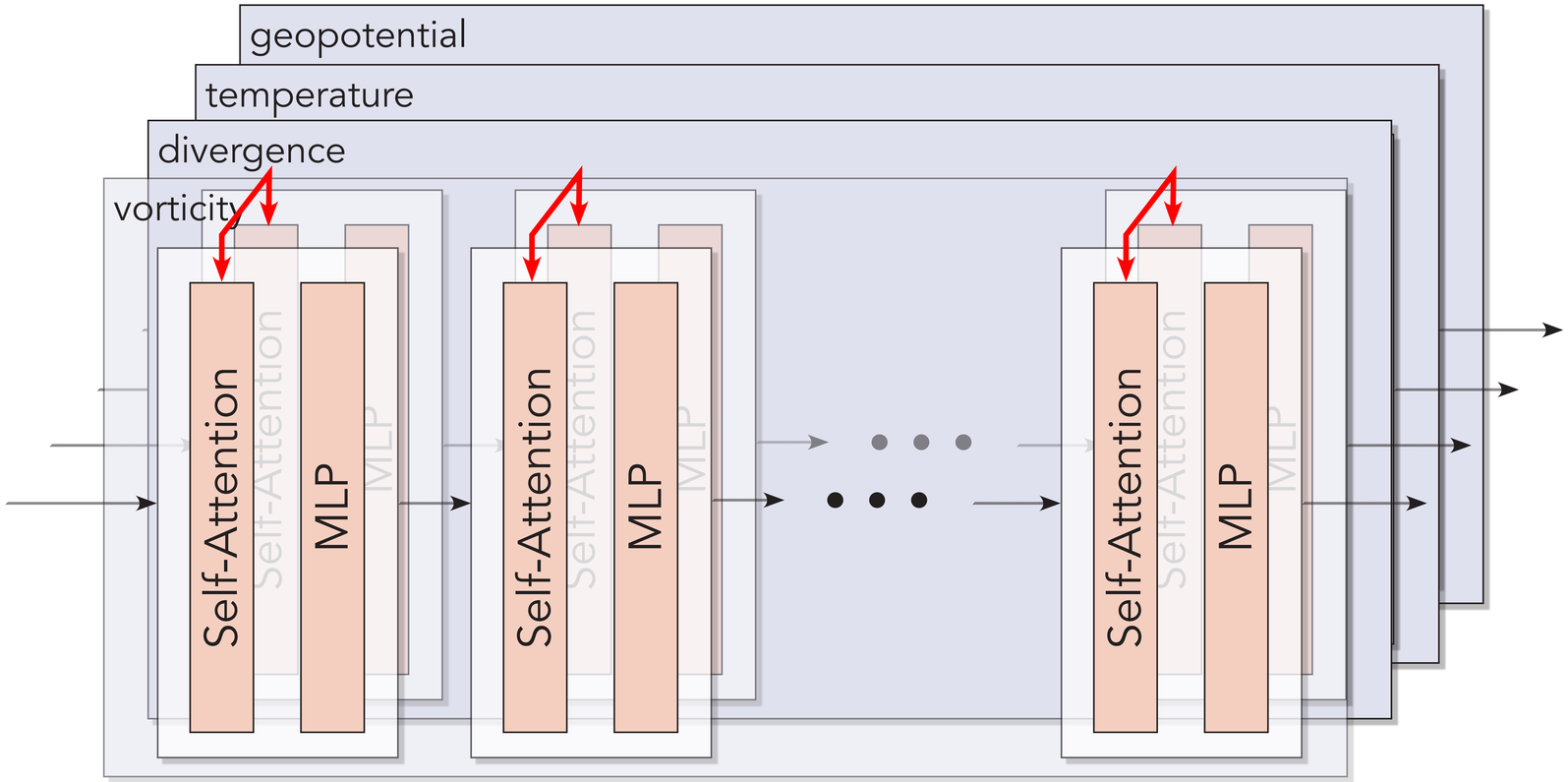
$$\sigma\left(Q\,K^T\right)V$$



Multiformer: respect the physical fields

Self
attention

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



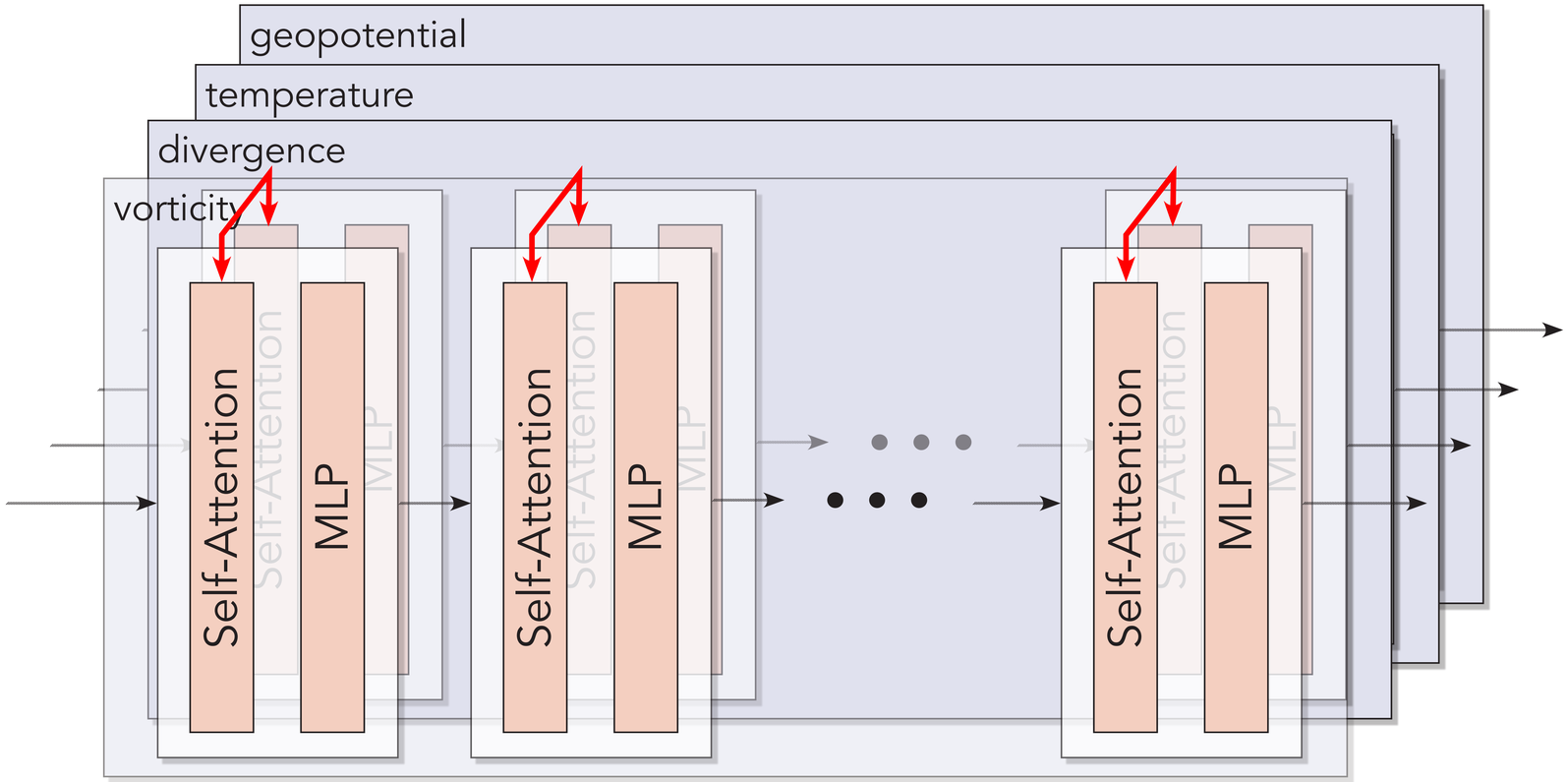
Multiformer: respect the physical fields

Self attention

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

Cross attention

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



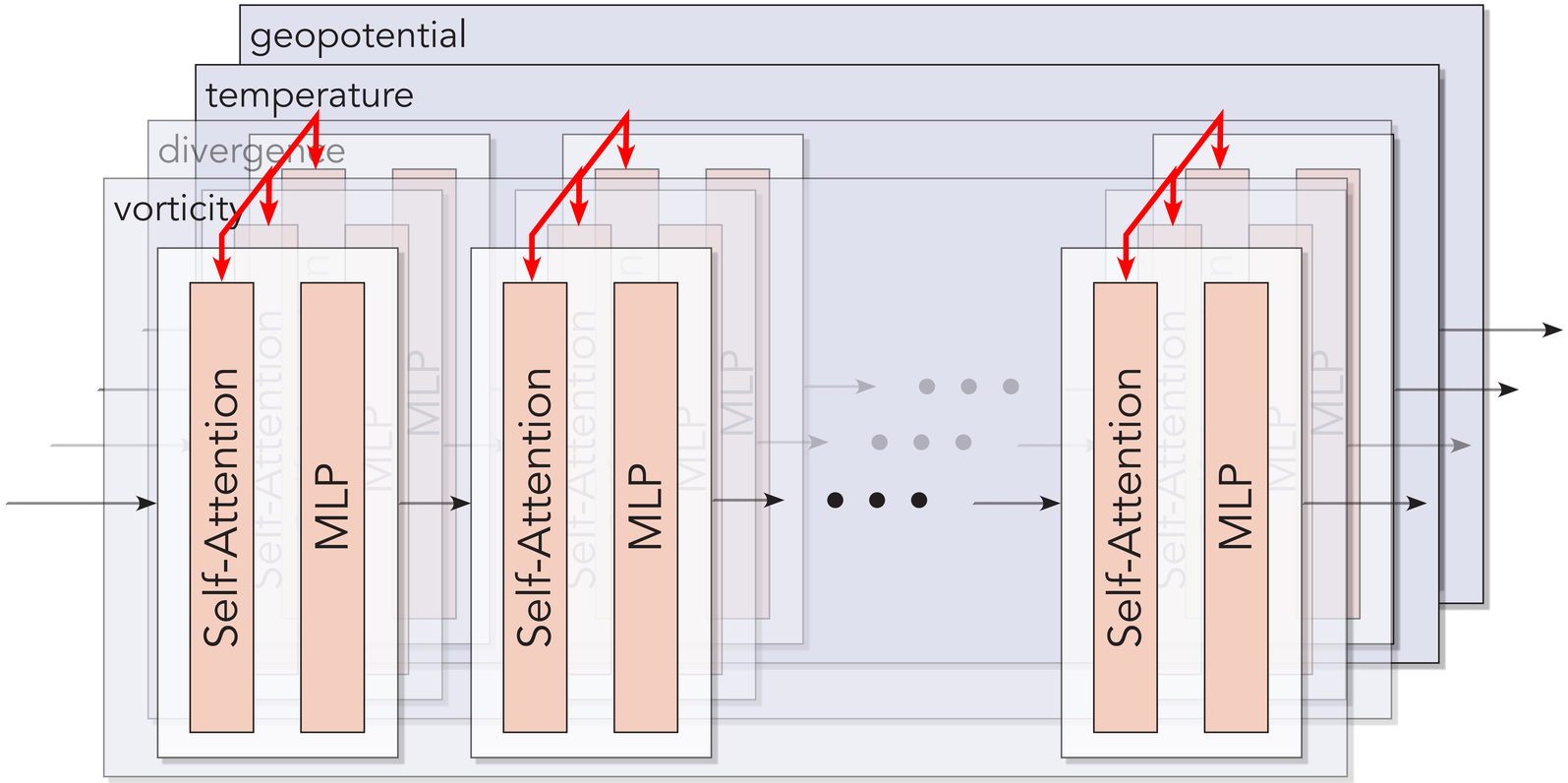
Multiformer: respect the physical fields

Self attention

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

Cross attention

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



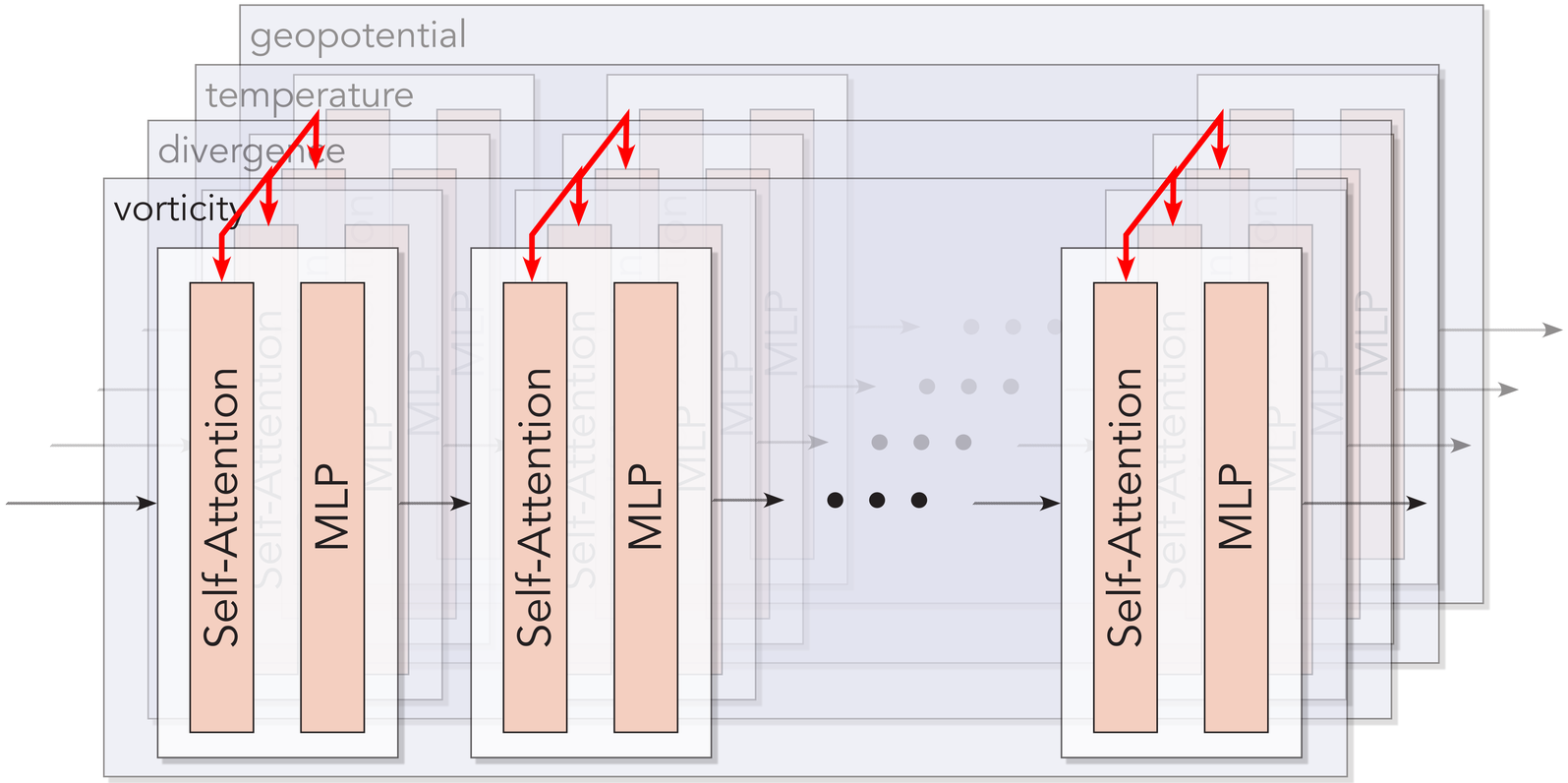
Multiformer: respect the physical fields

Self attention

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

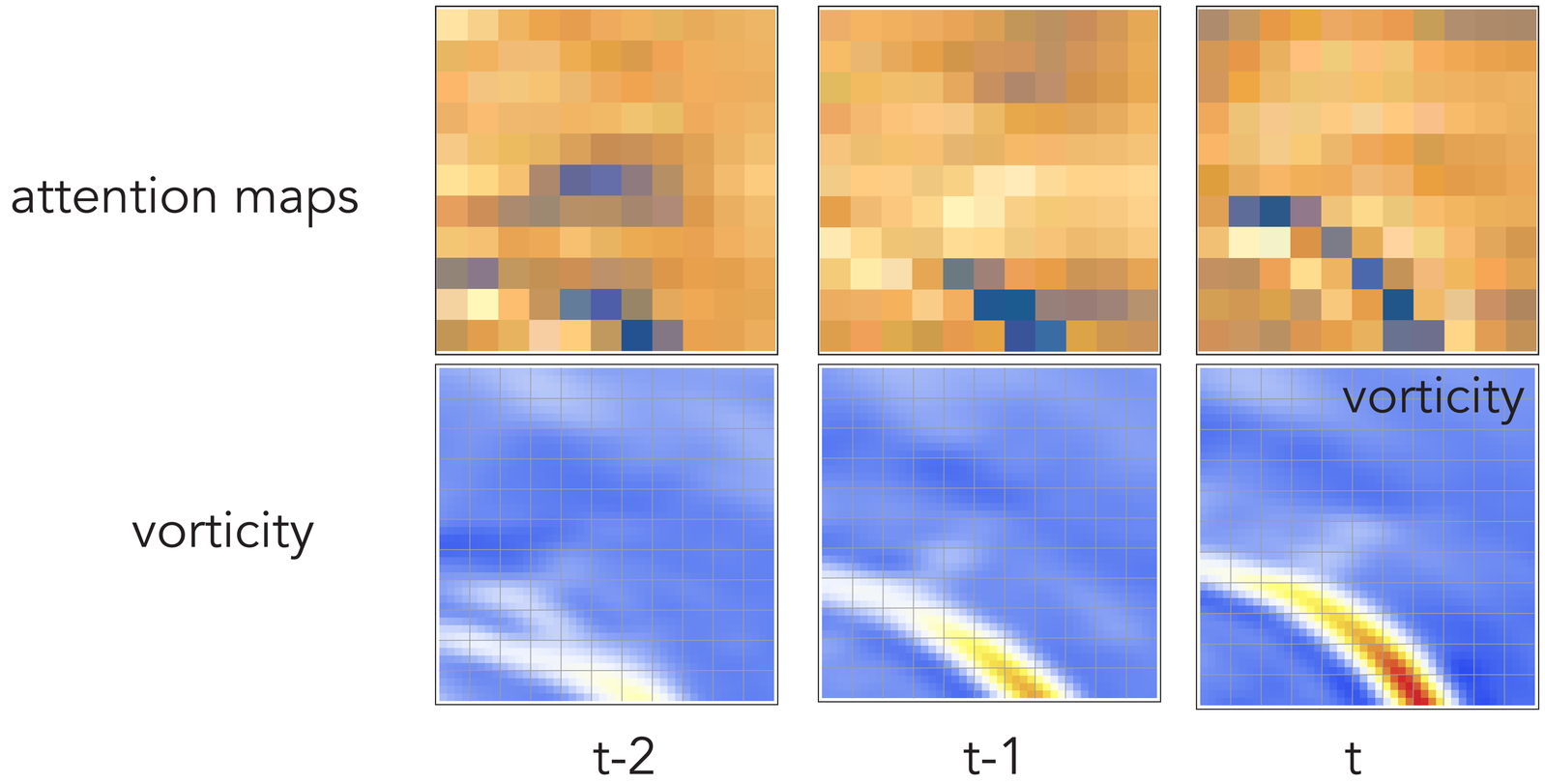
Cross attention

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



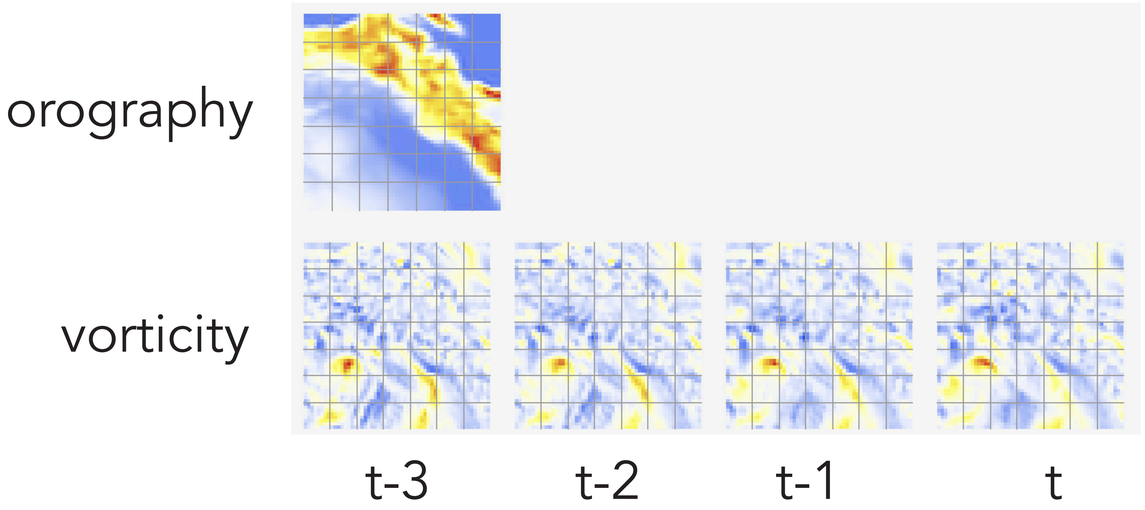
Multiformer: respect the physical fields

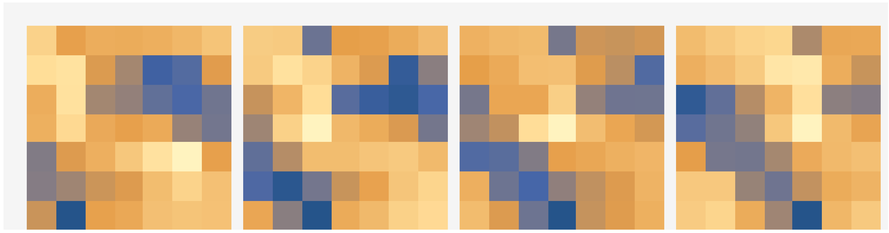
- Attention maps:



Multiformer: respect the physical fields

- Attention maps:

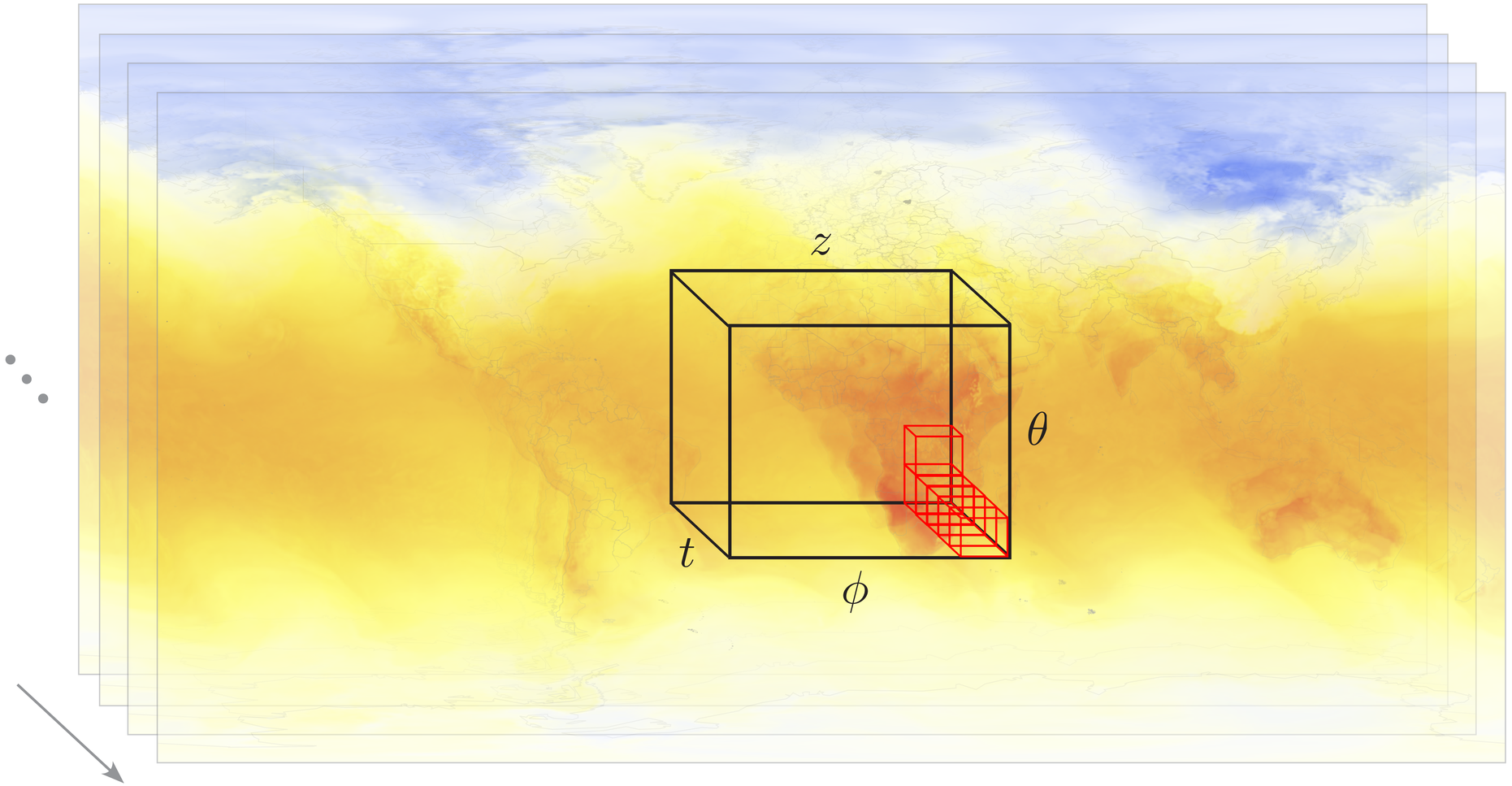




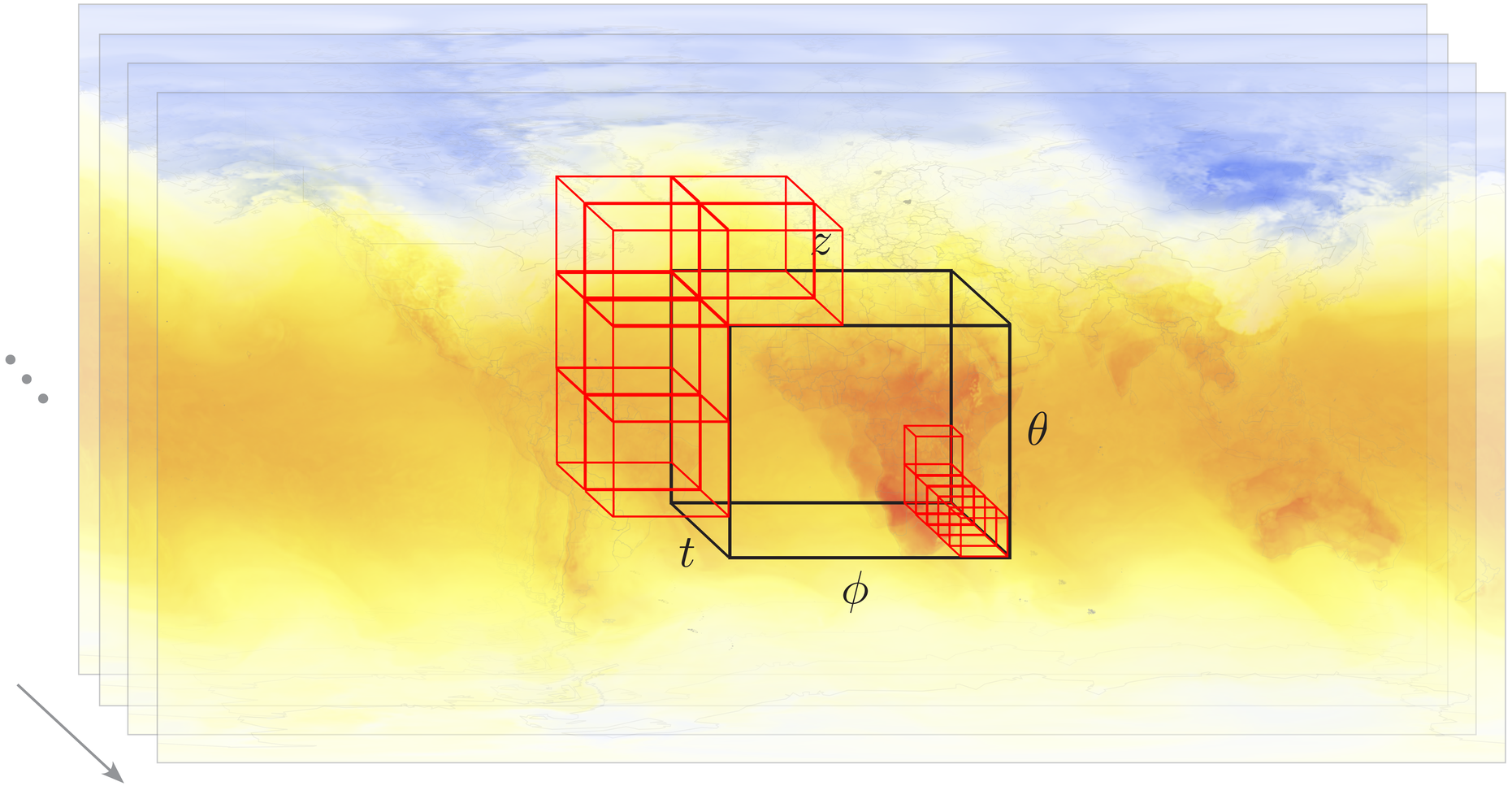
Embedding of tokens

- 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



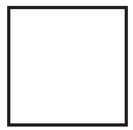
Embedding of tokens

- 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

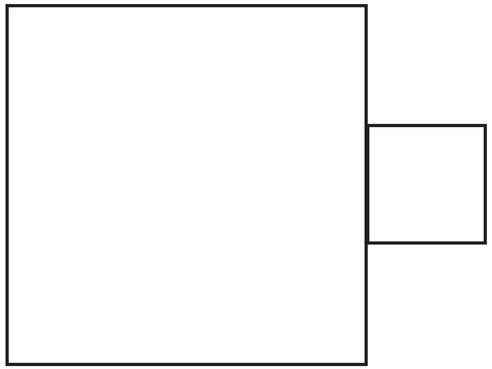
Embedding of tokens

- Allow for tokens of different size in space-time



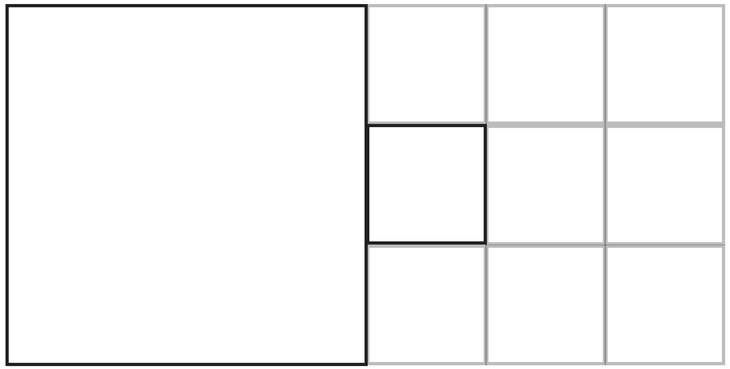
Embedding of tokens

- Allow for tokens of different size in space-time



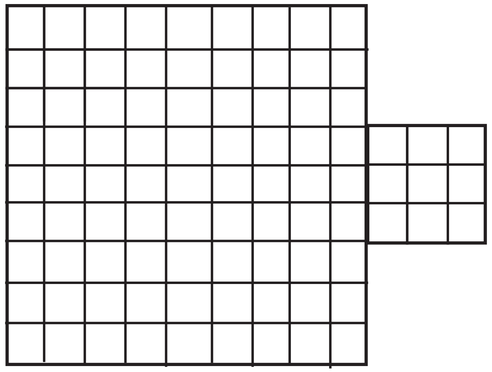
Embedding of tokens

- Allow for tokens of different size in space-time



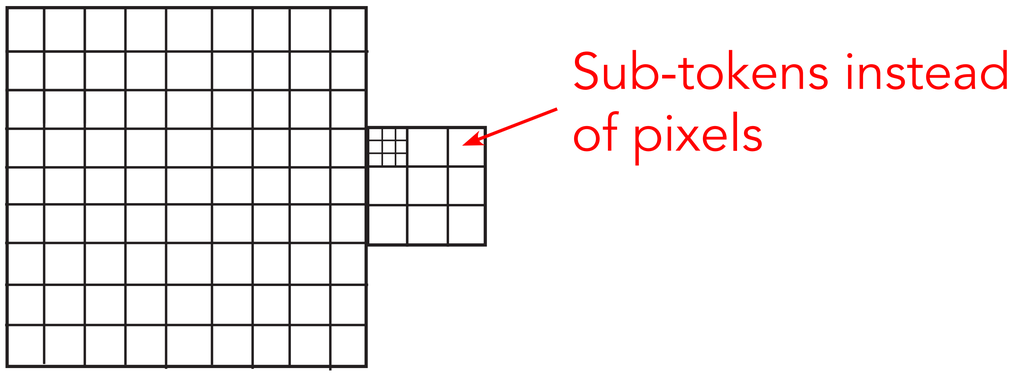
Embedding of tokens

- Allow for tokens of different size in space-time



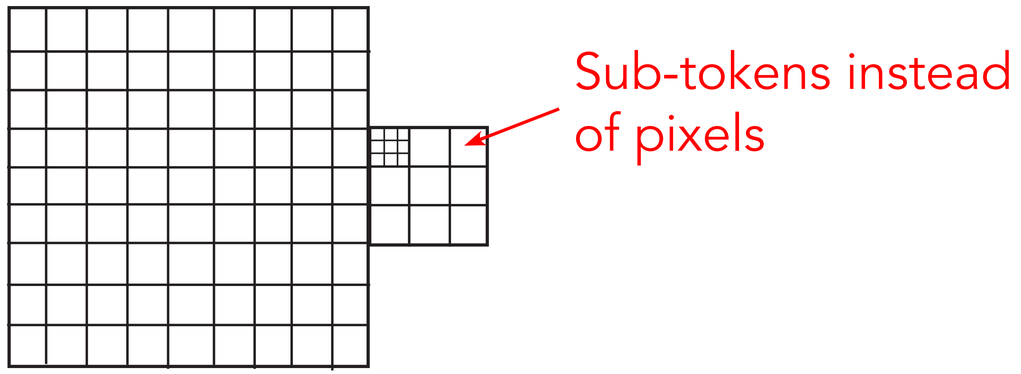
Embedding of tokens

- Allow for tokens of different size in space-time



Embedding of tokens

- Allow for tokens of different size in space-time

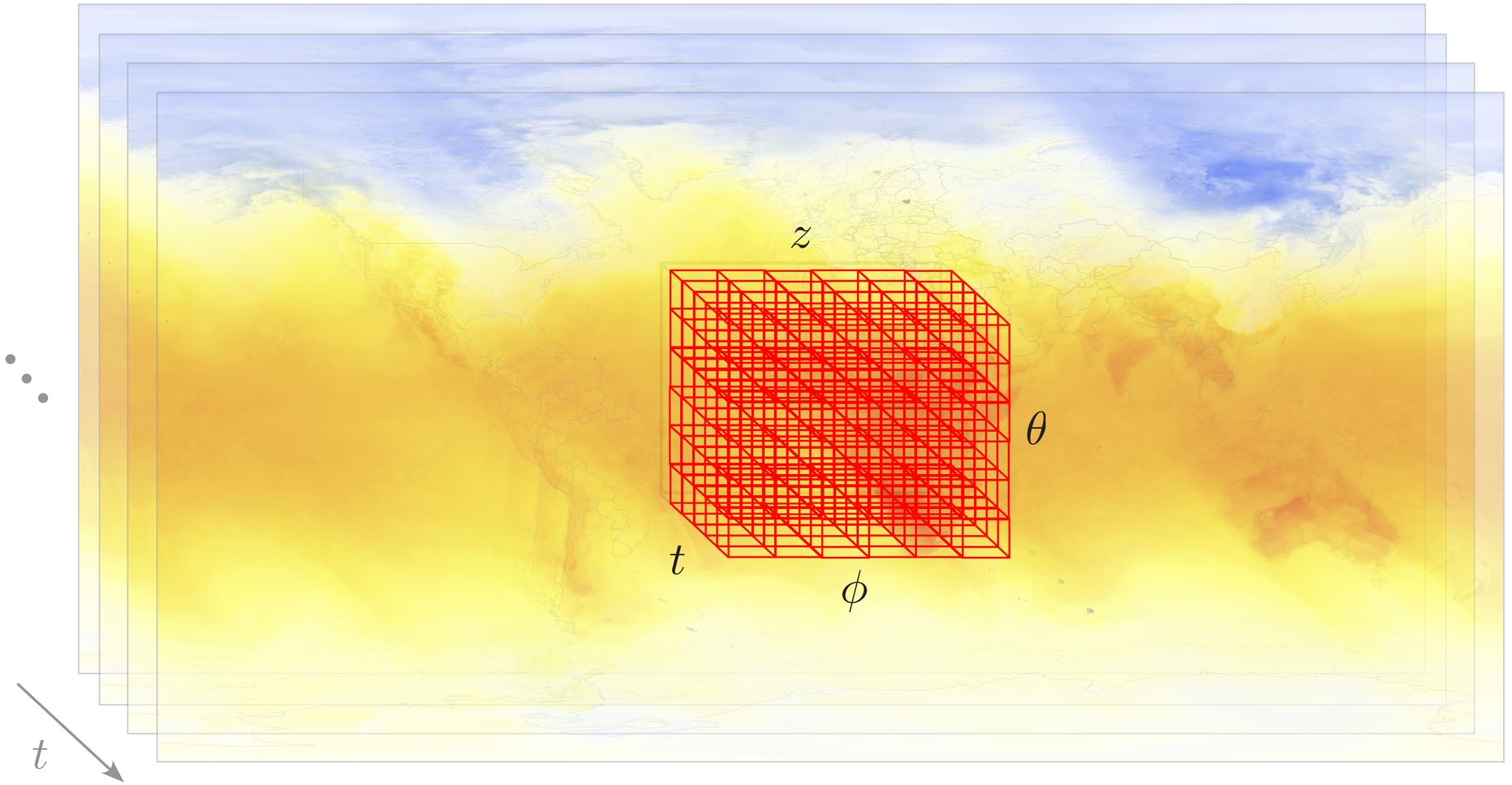


- Transformer takes an arbitrary number of tokens as input
  - Training yields consistent embedding

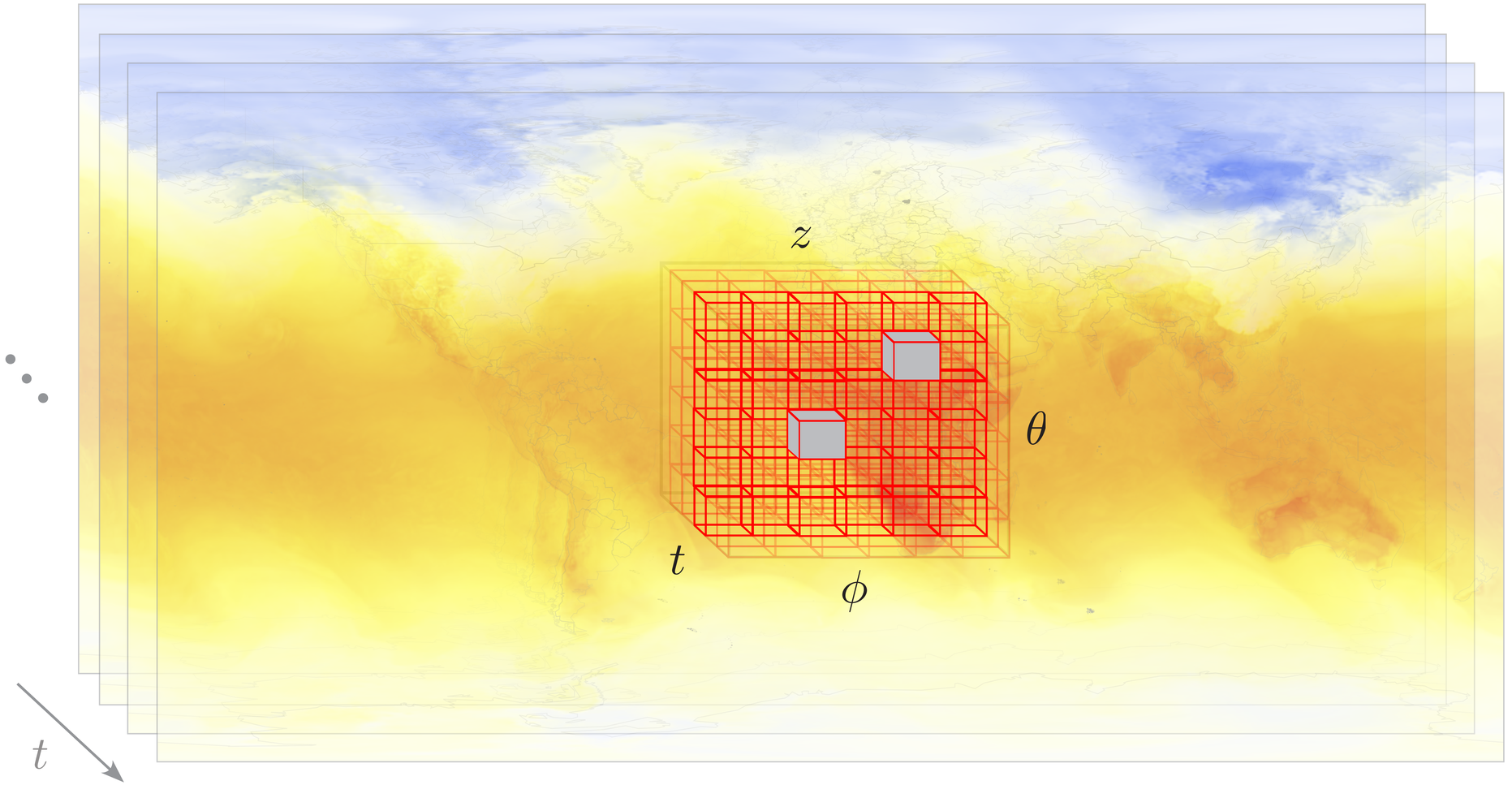
Training of embedding network

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

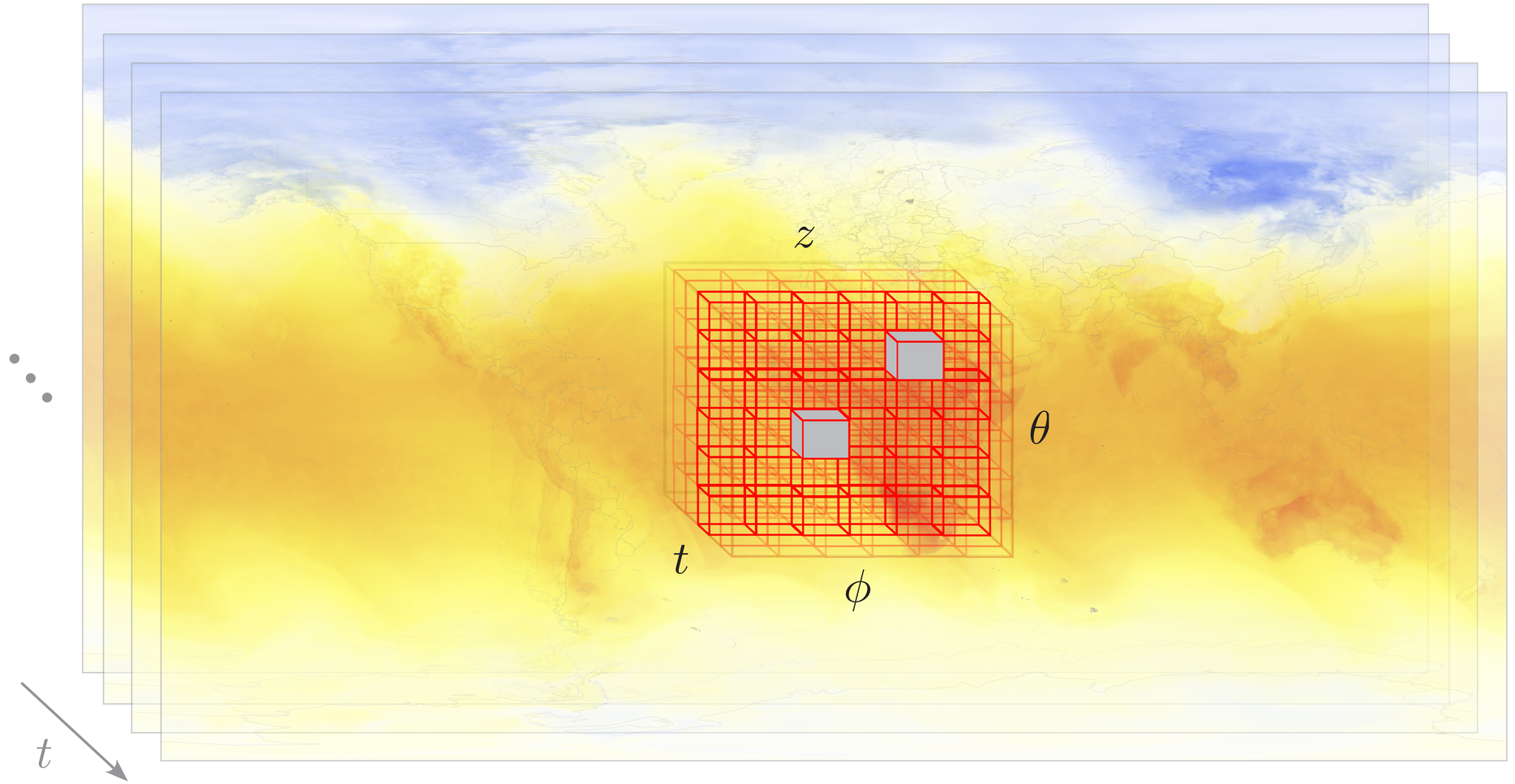
AtmoRep training



AtmoRep training



AtmoRep training



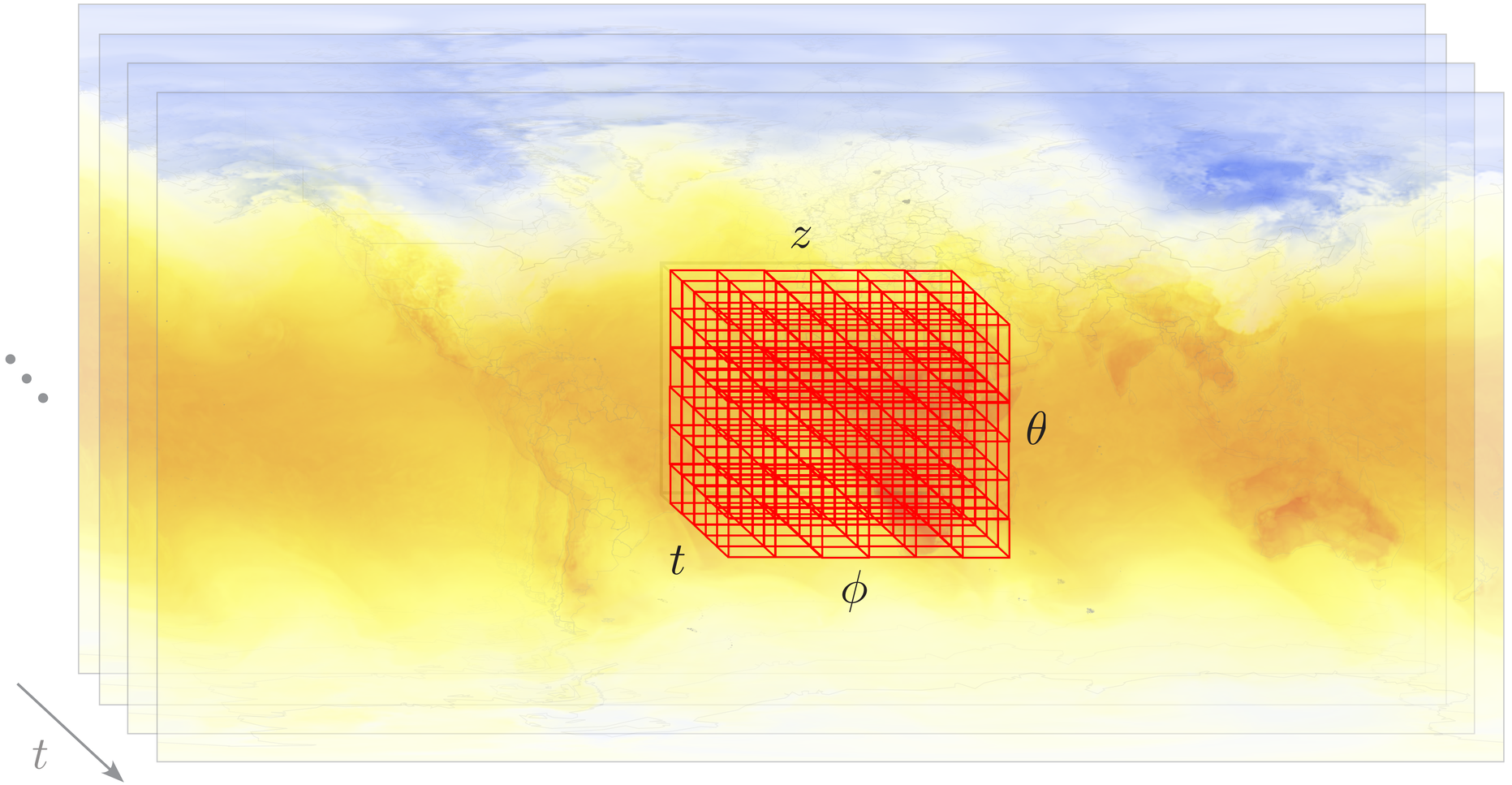
Training of embedding network

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

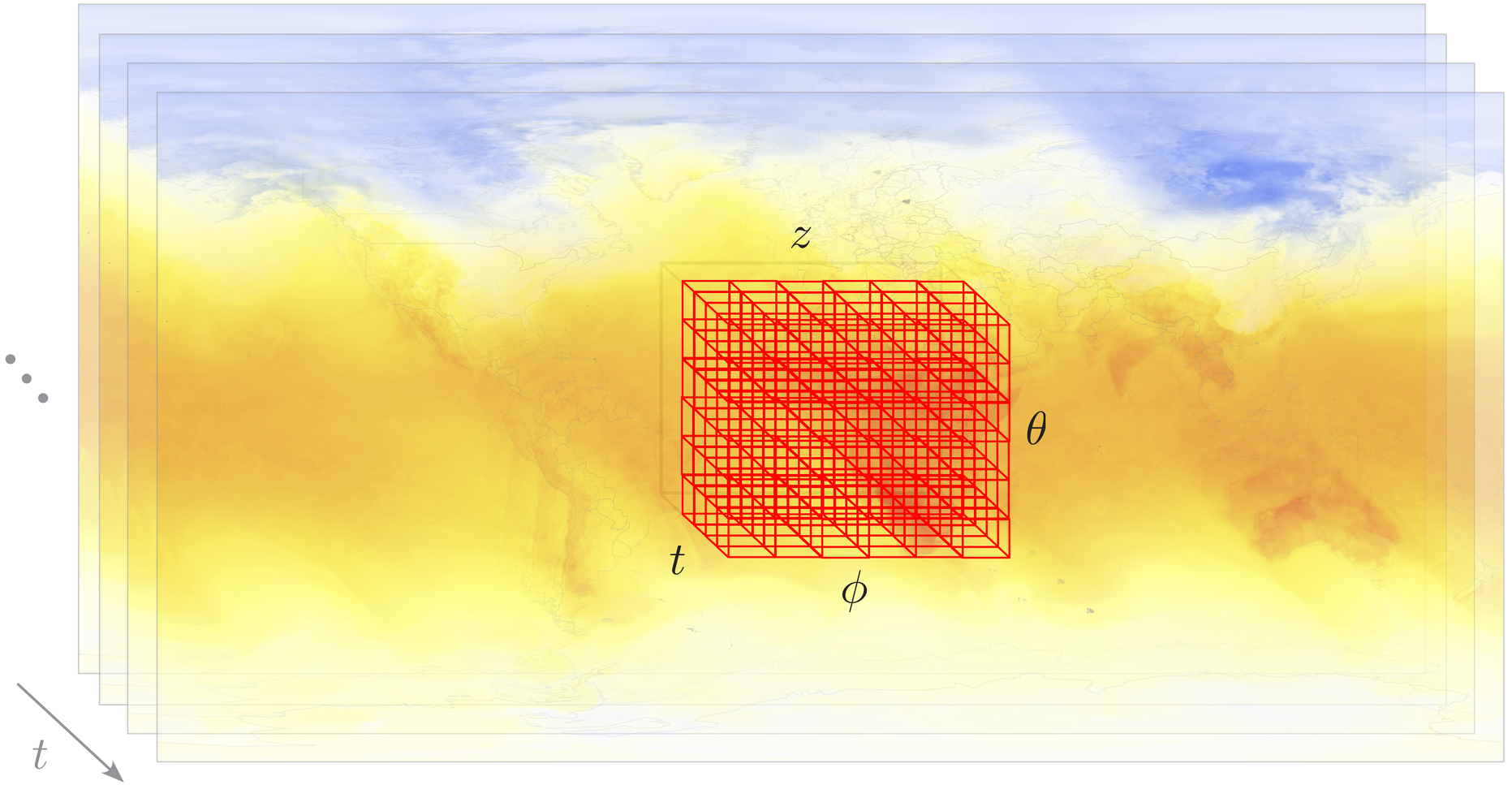
Training of embedding network

- 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

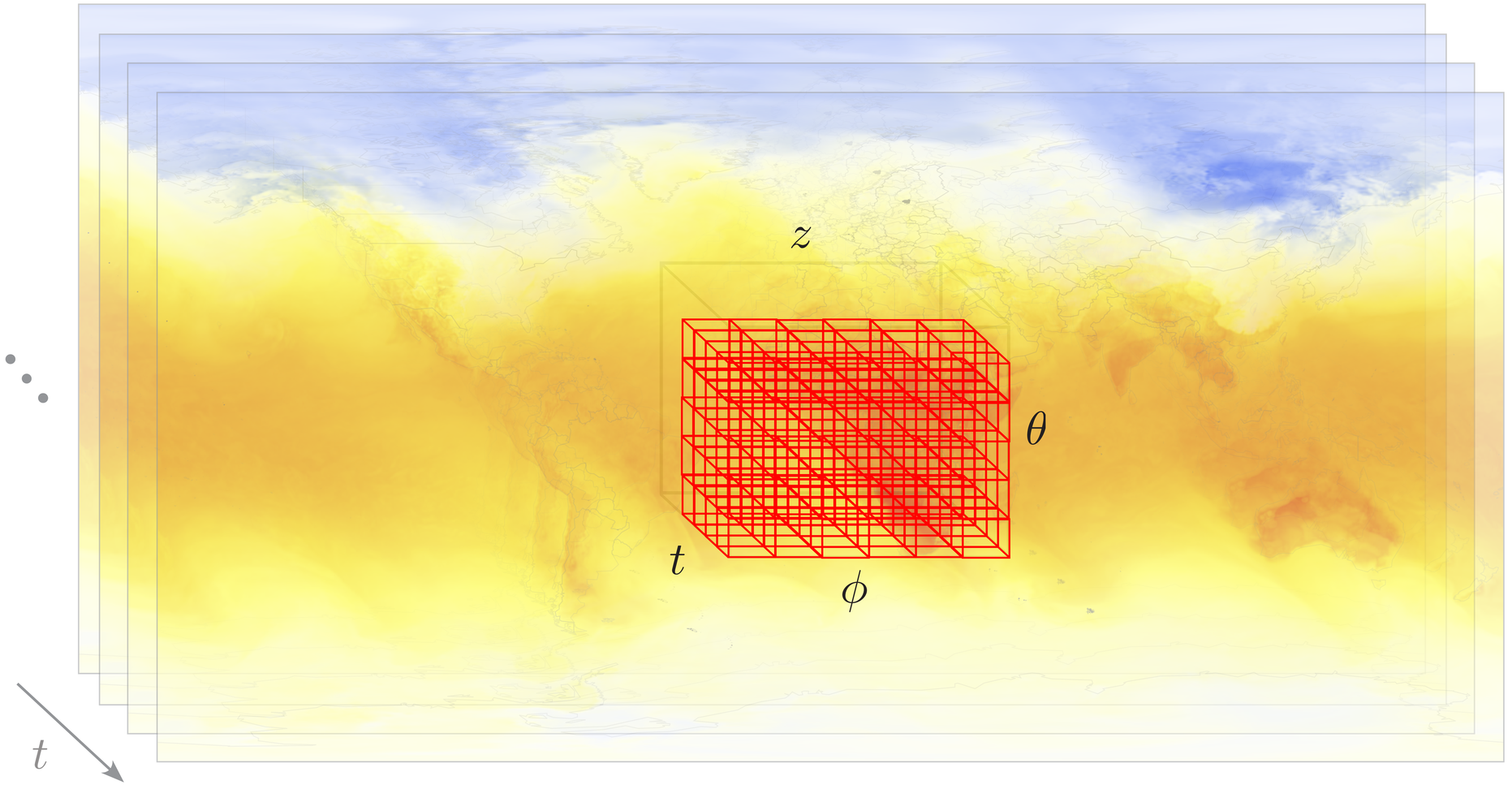
AtmoRep training



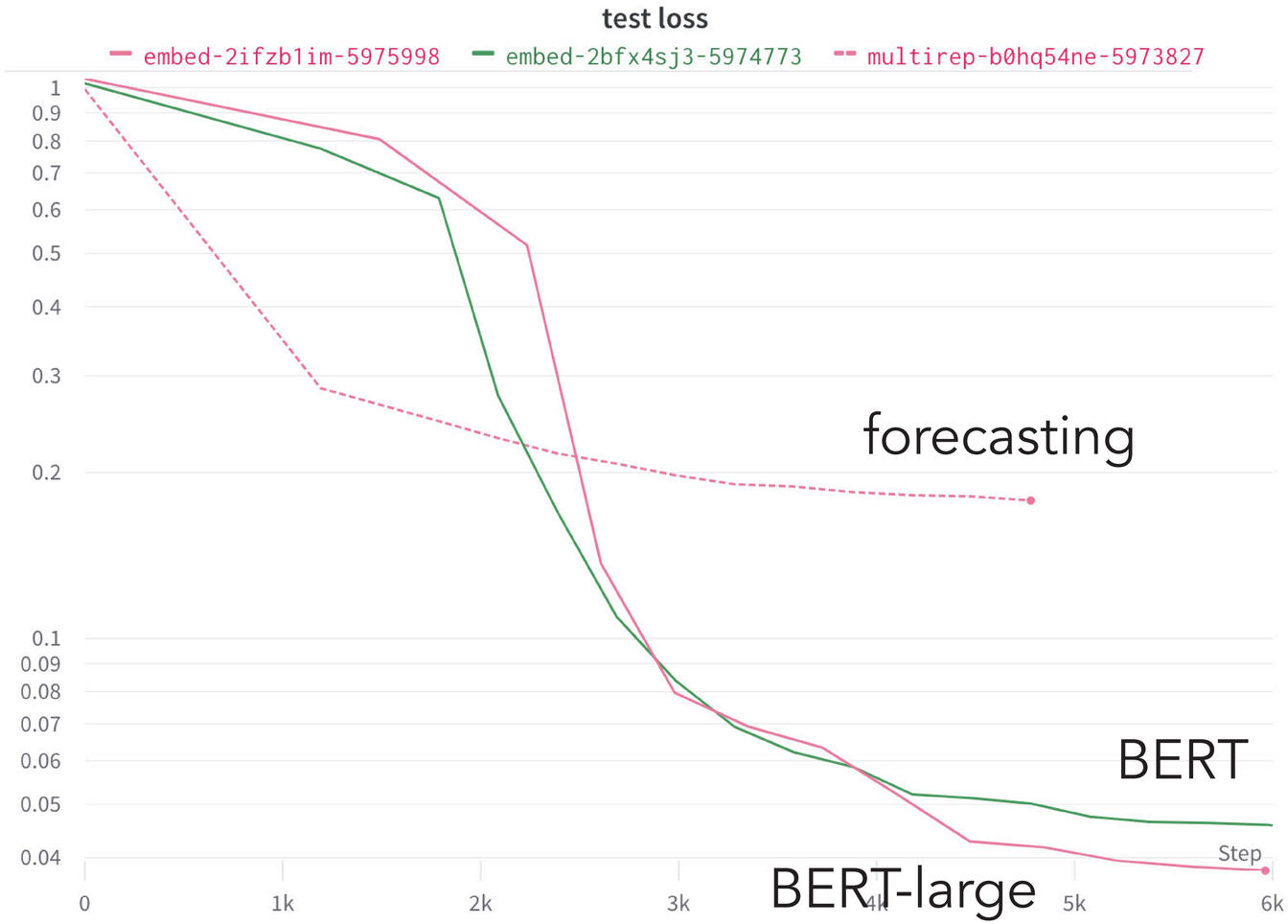
AtmoRep training



AtmoRep training



AtmoRep training

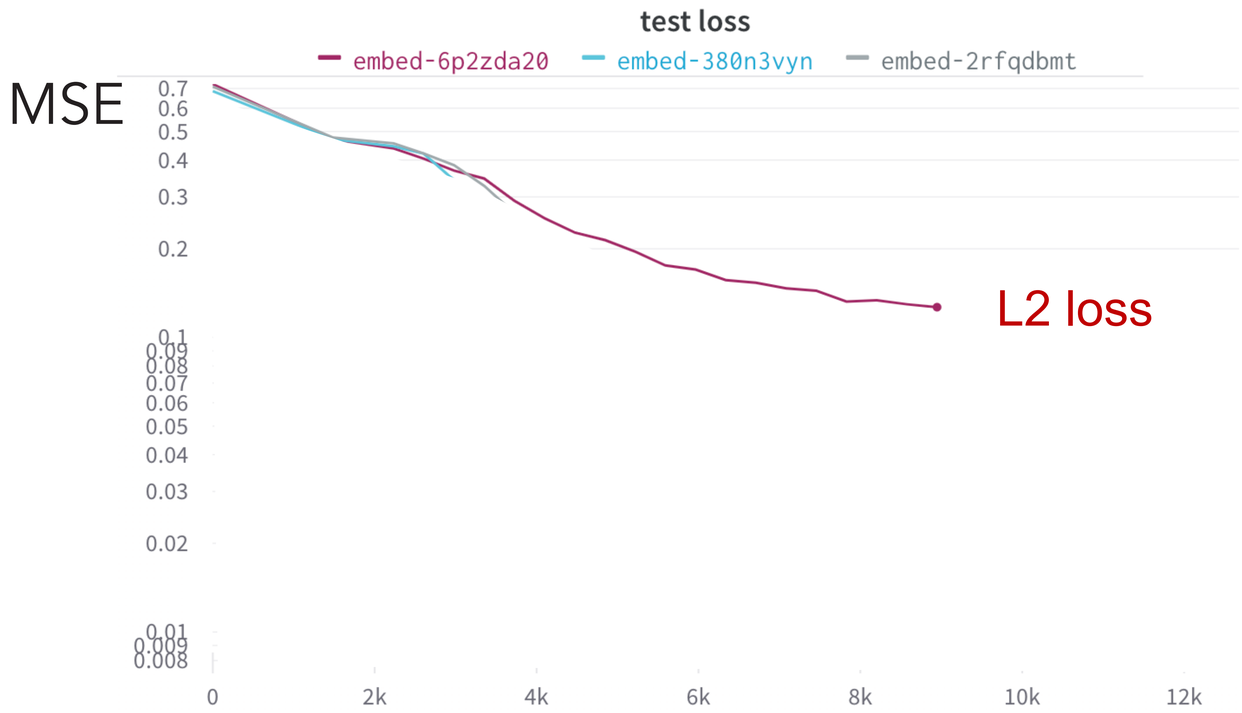


Statistical loss: respect the stochasticity

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

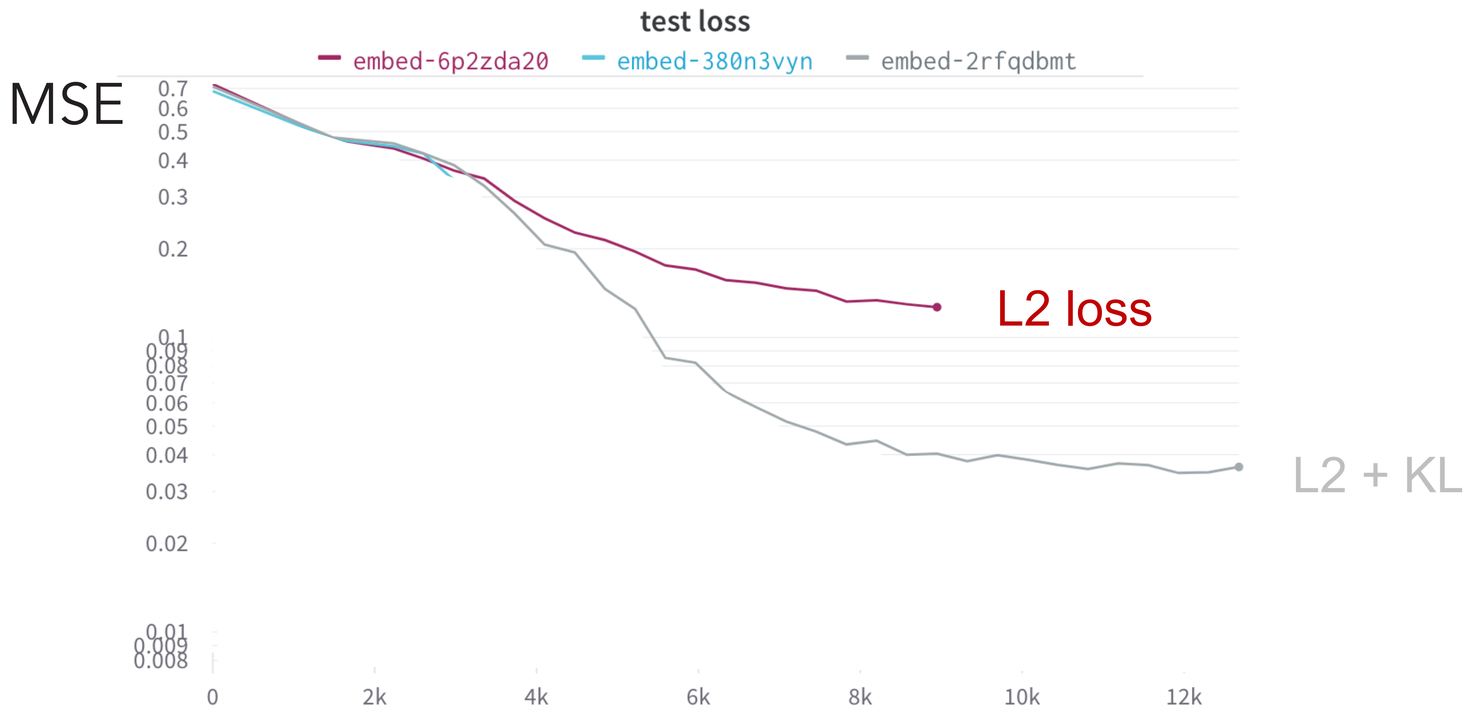
Statistical loss: respect the stochasticity

- 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.

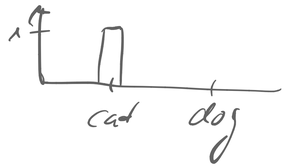
The advantage of this procedure is that the Transformer encoder does not know which words it will be asked to predict or which have been replaced by random words, so it is forced to keep a distributional contextual representation of *every* input token. Additionally, because random replacement only occurs 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.

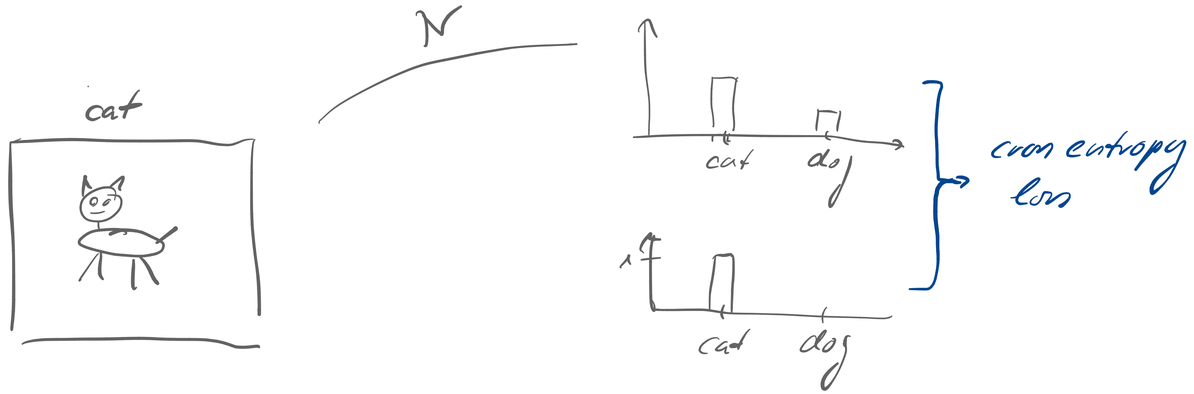
Statistical loss: respect the stochasticity

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

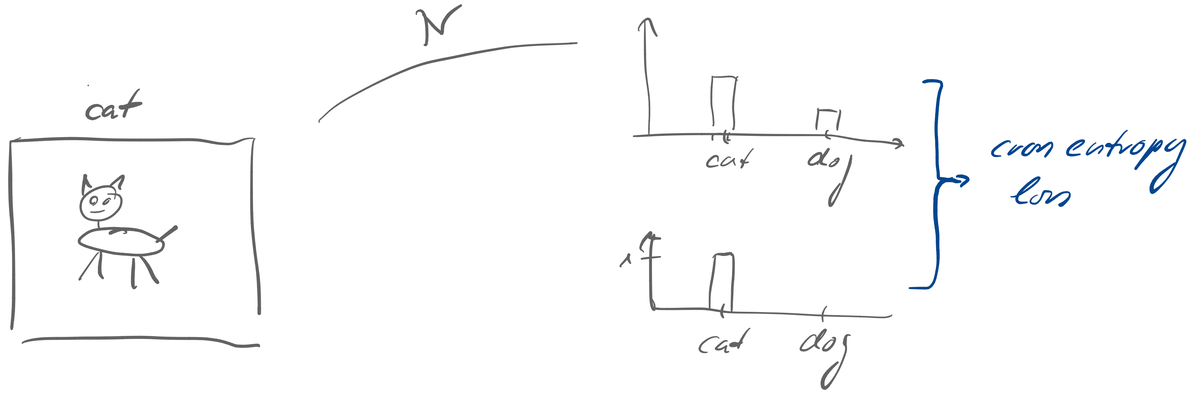
Statistical loss: respect the stochasticity

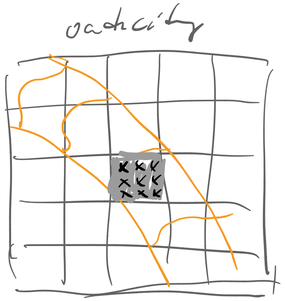


Statistical loss: respect the stochasticity

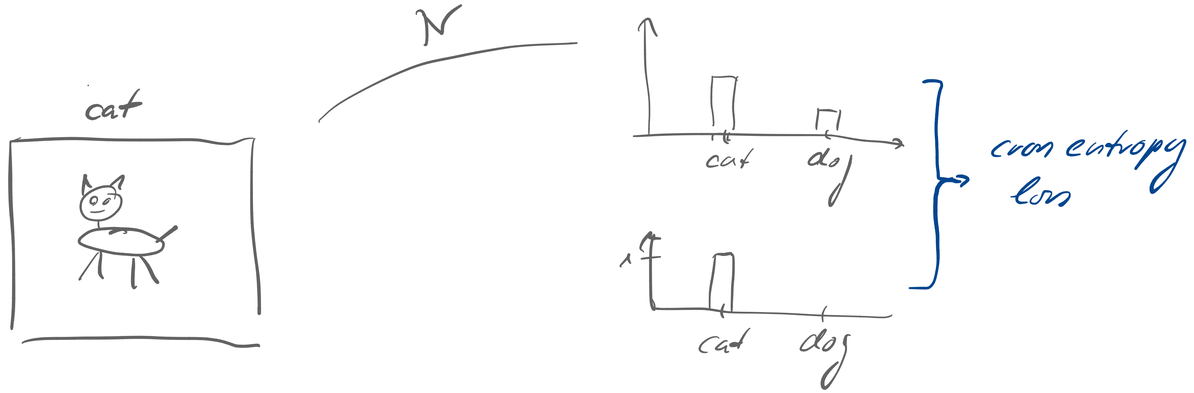


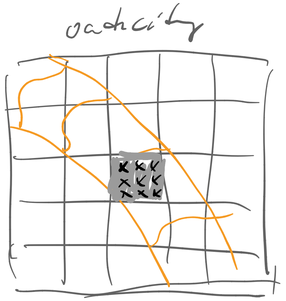
Statistical loss: respect the stochasticity





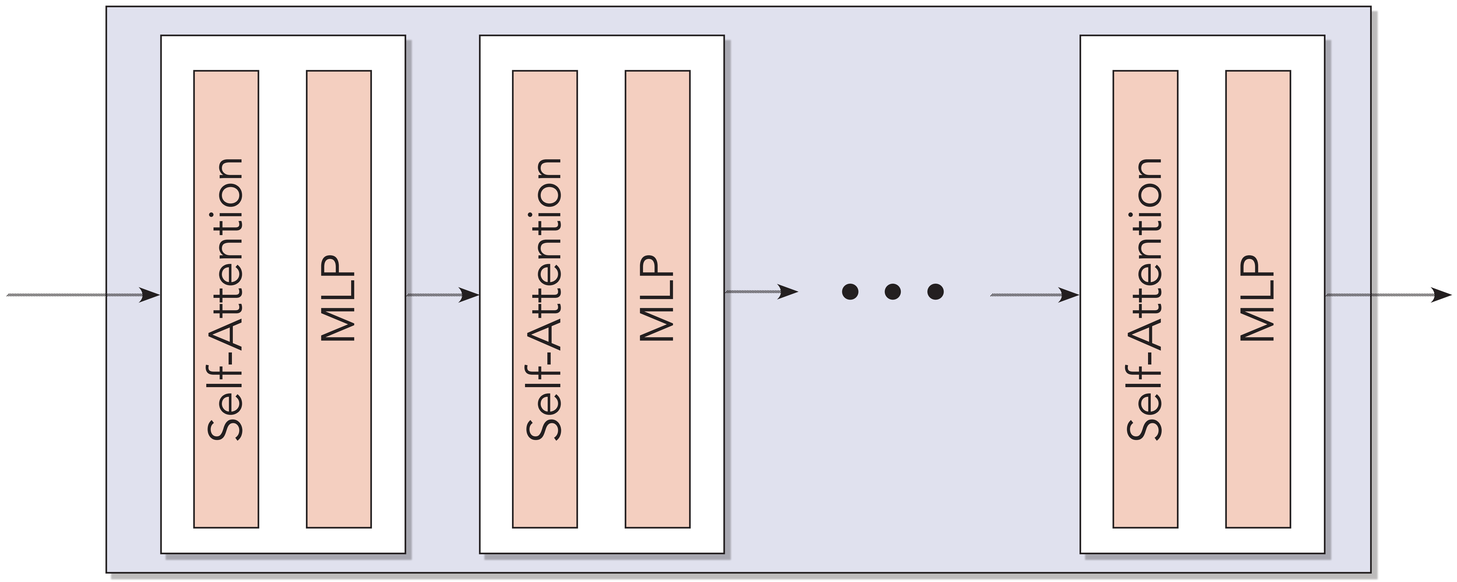
Statistical loss: respect the stochasticity



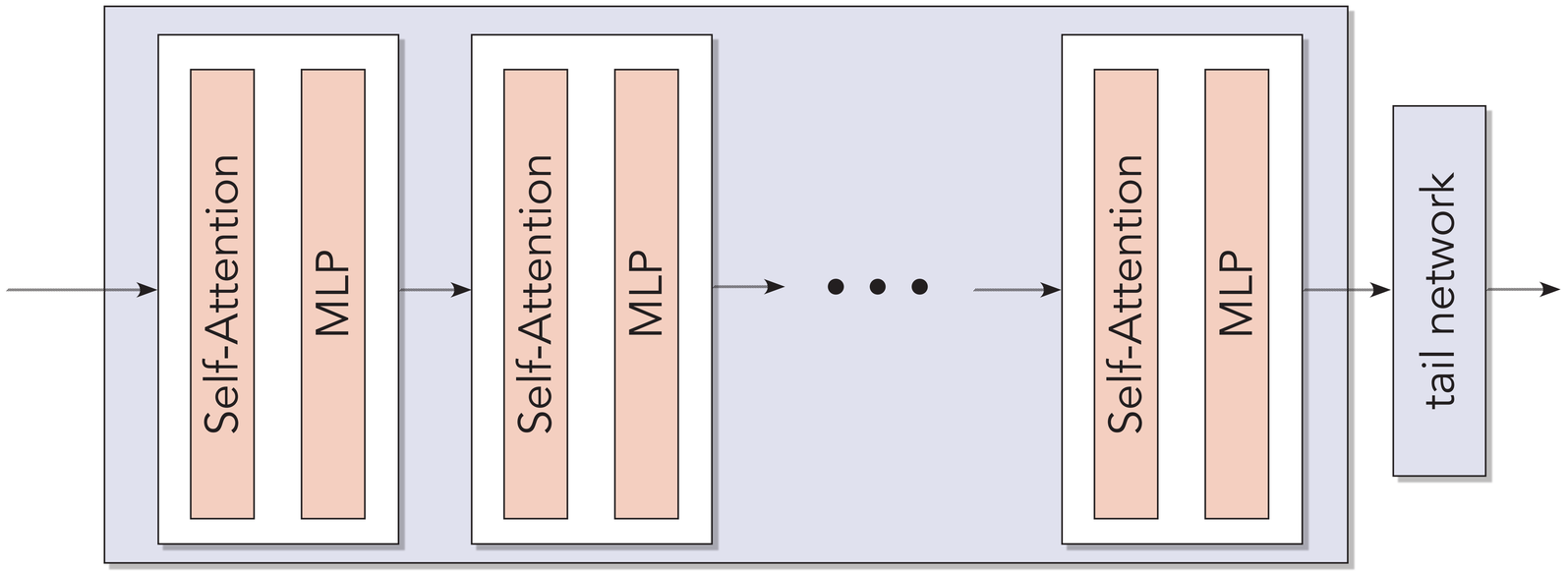


?

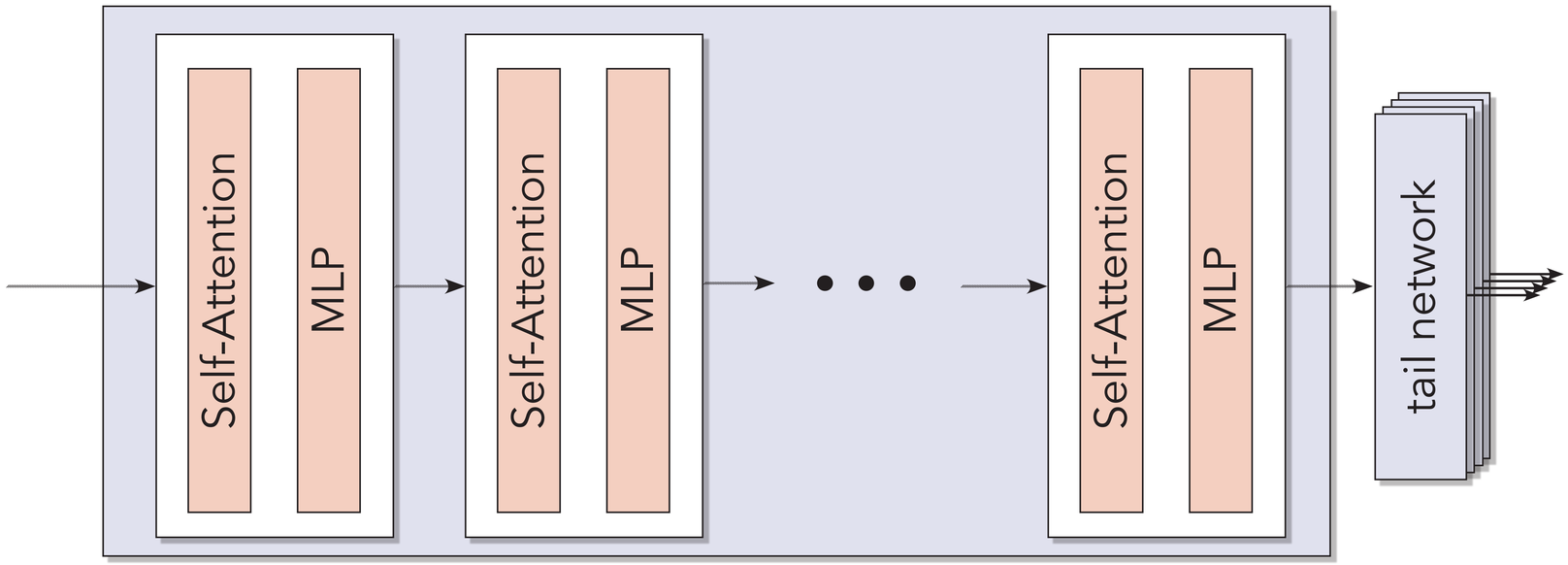
Statistical loss: respect the stochasticity



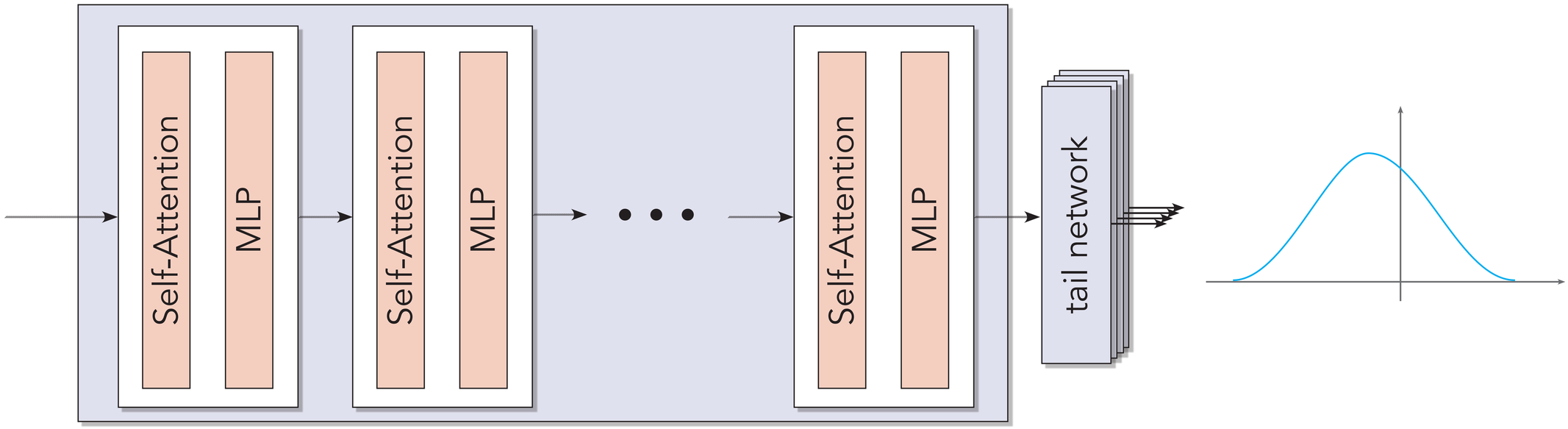
Statistical loss: respect the stochasticity



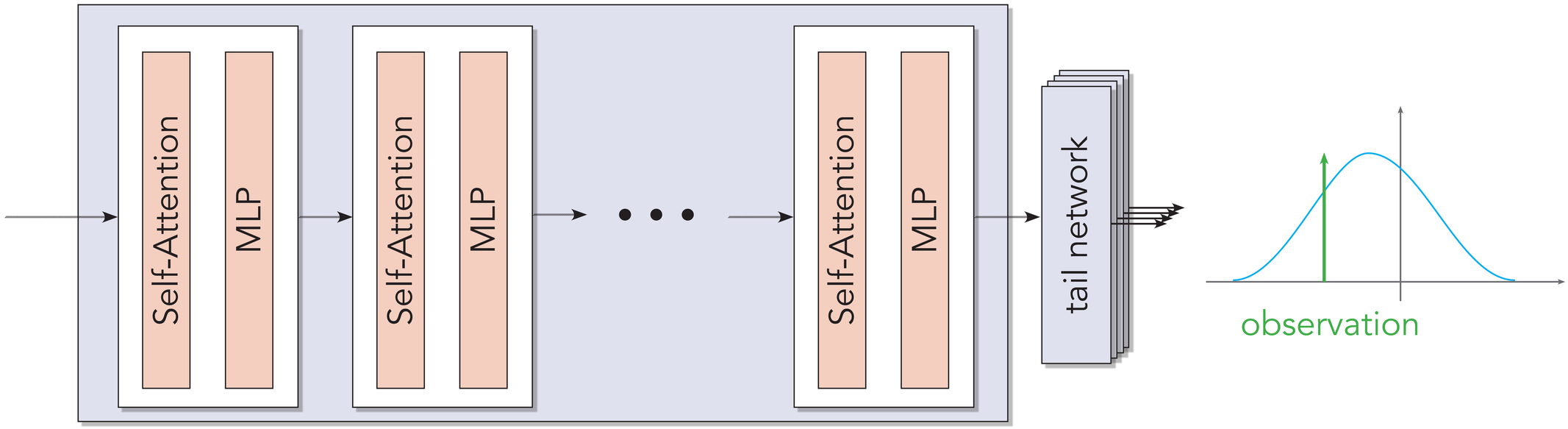
Statistical loss: respect the stochasticity



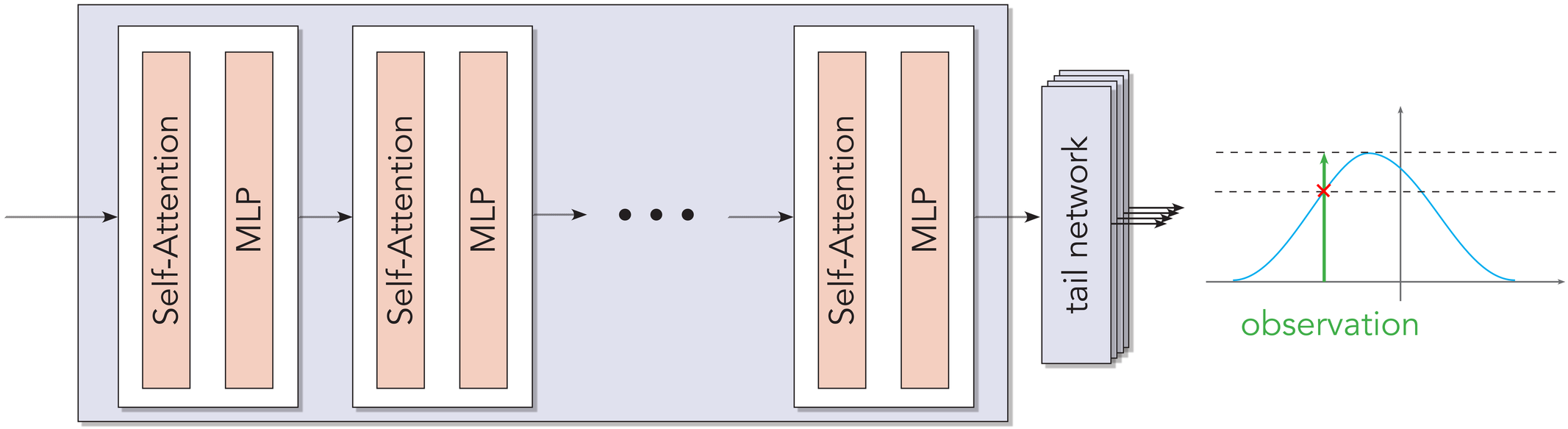
Statistical loss: respect the stochasticity



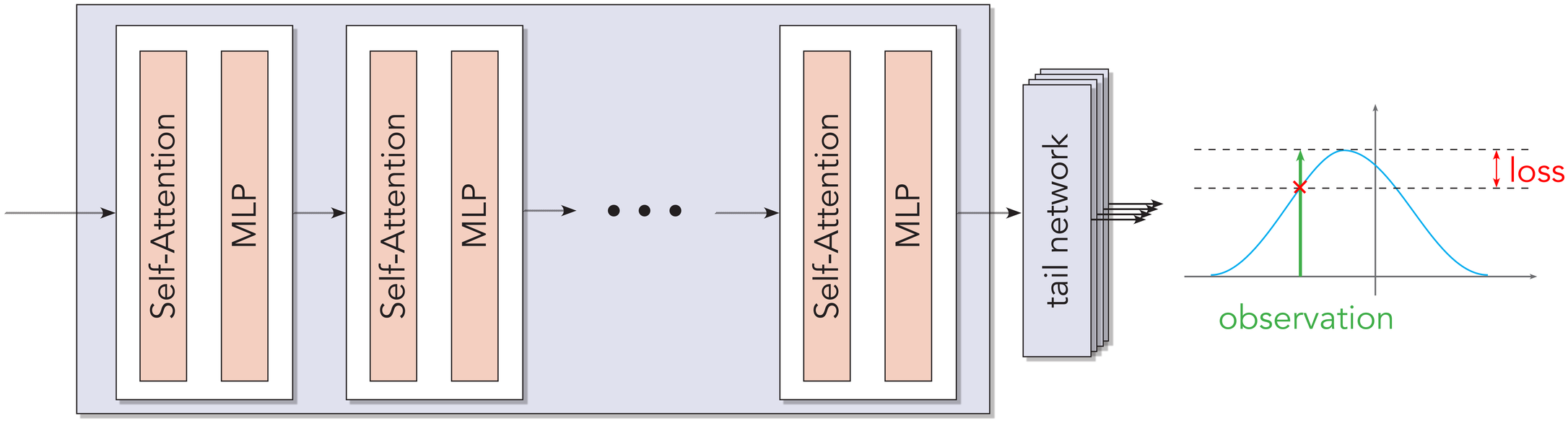
Statistical loss: respect the stochasticity



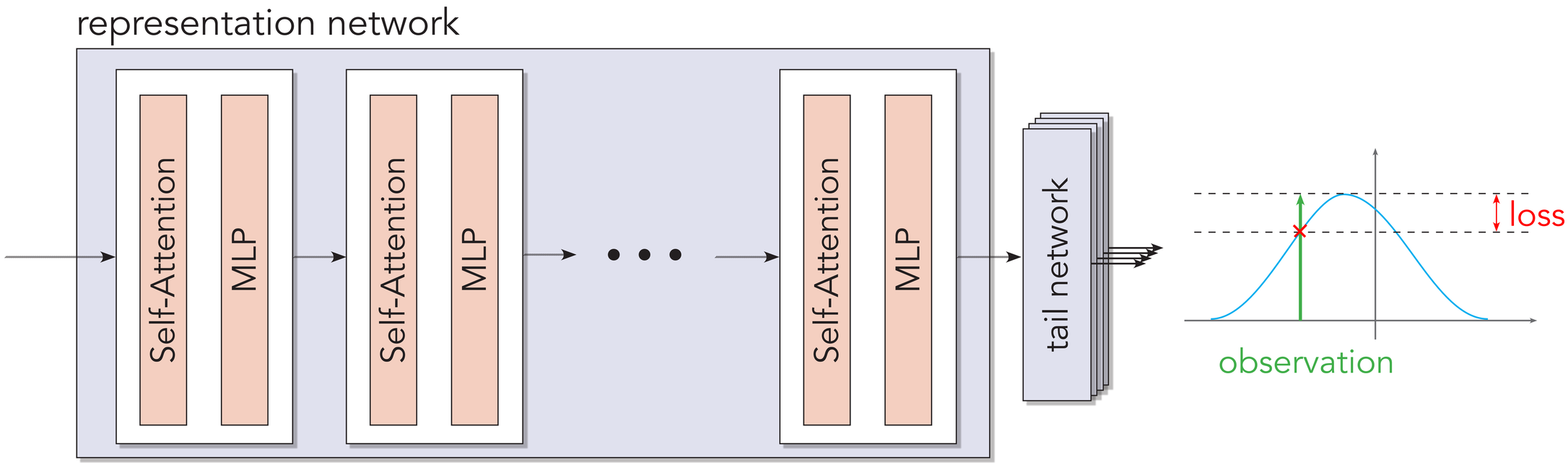
Statistical loss: respect the stochasticity



Statistical loss: respect the stochasticity



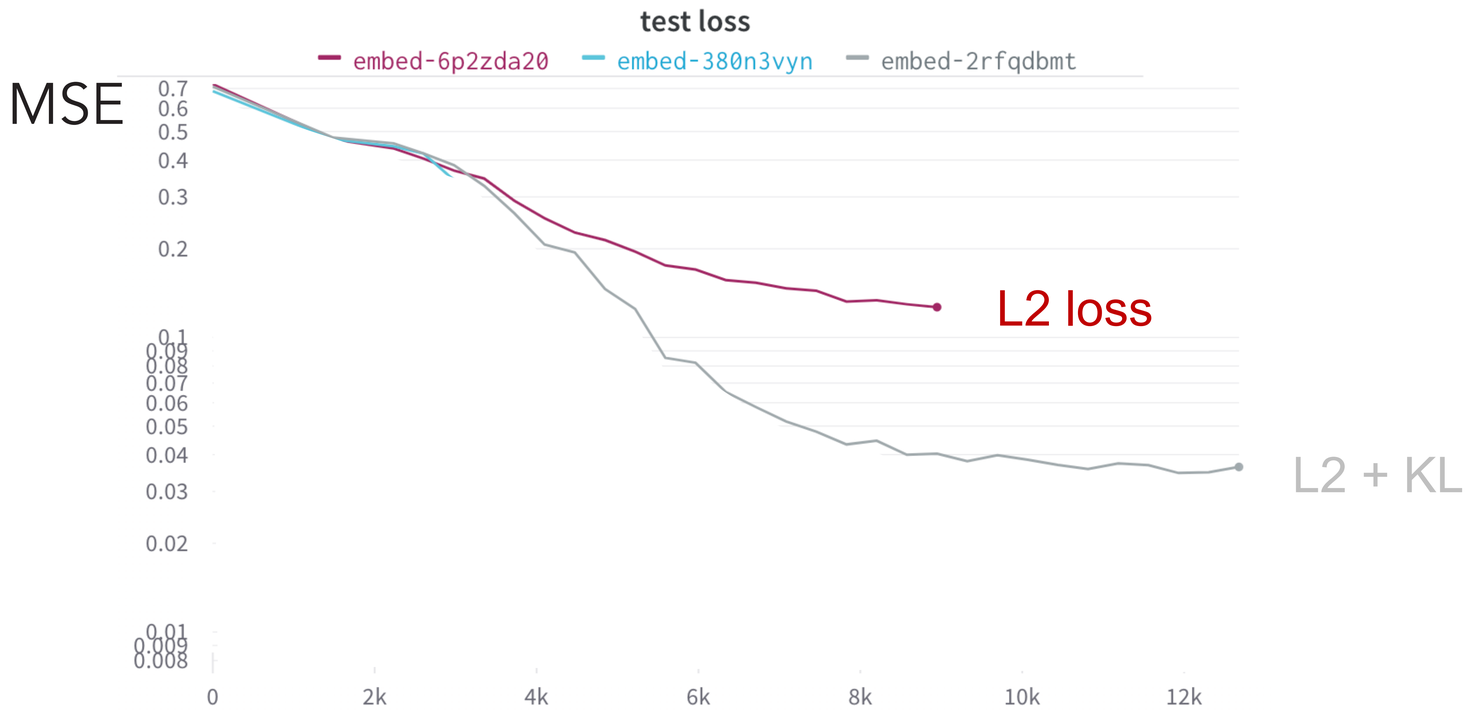
Statistical loss: respect the stochasticity



end-to-end training with ensemble facilitates statistical representation

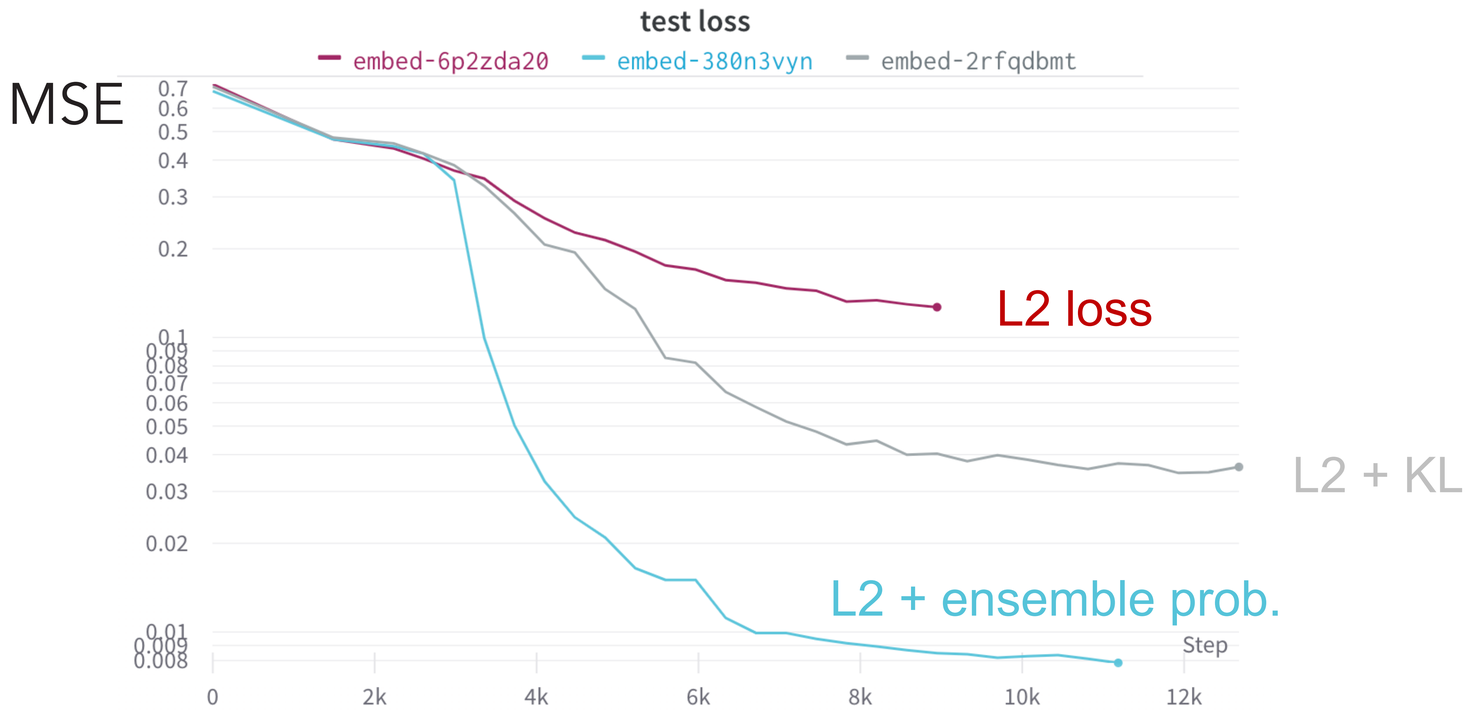
Statistical loss: respect the stochasticity

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

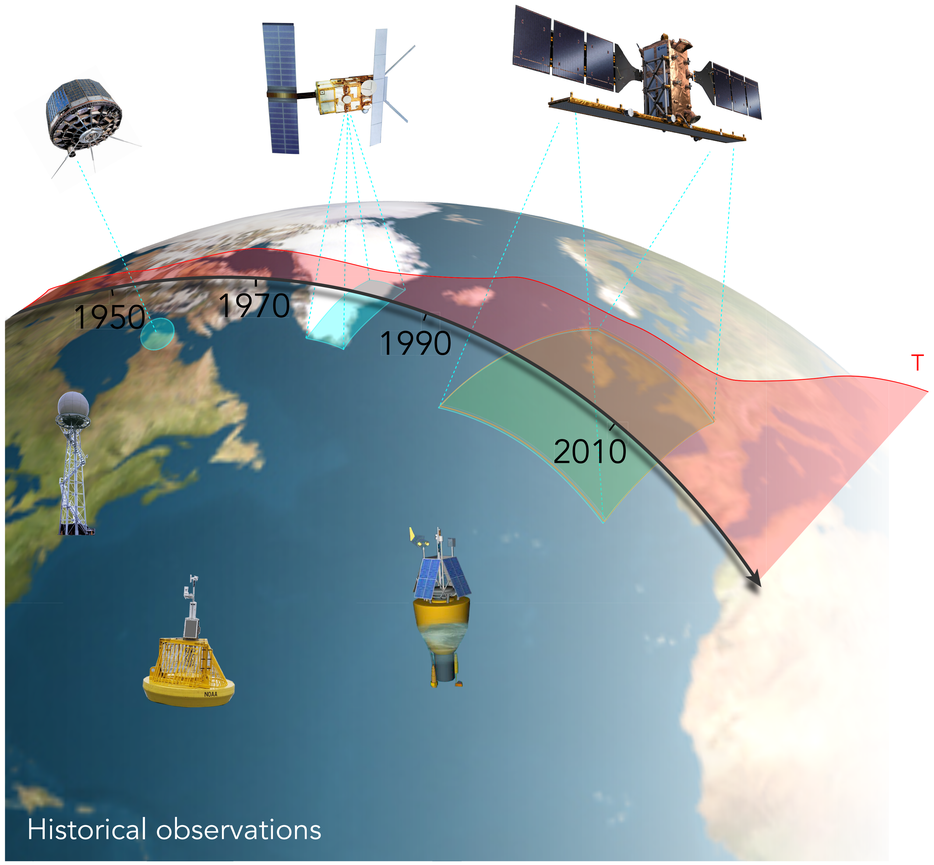


Statistical loss: respect the stochasticity

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



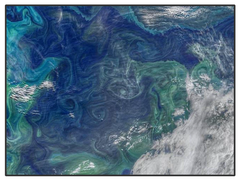
AtmoRep

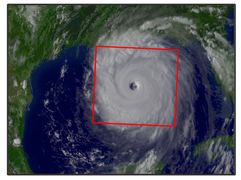


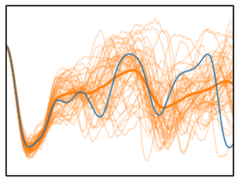
ERA5 reanalysis

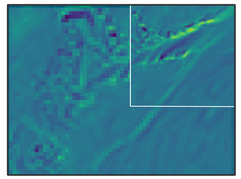
large scale machine learning

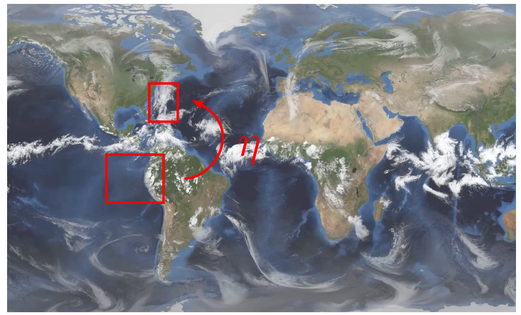
applications


Forecasting


Impact analysis


Climate projections

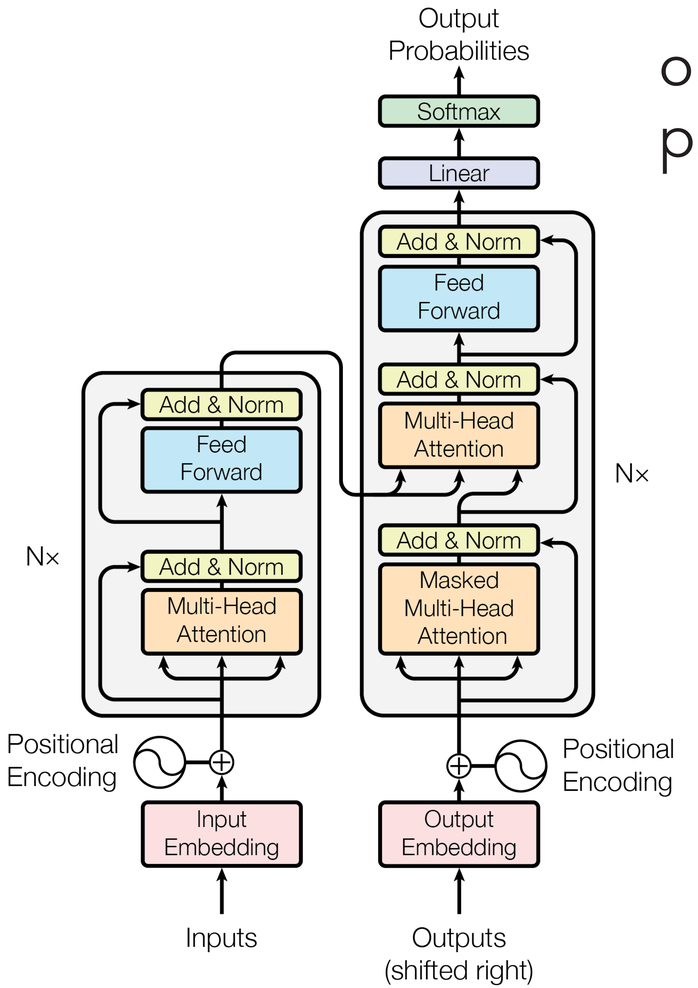

Downscaling



scientific insight

AtmoRep: forecasting and climate projections

AtmoRep: forecasting and climate projections



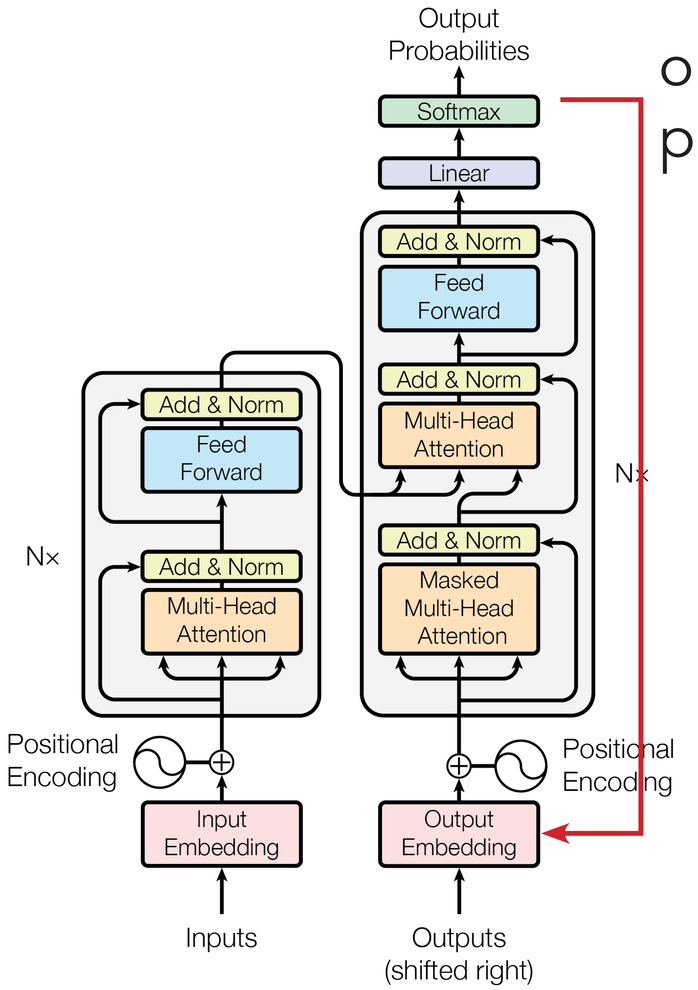
output stream of predicted words

encoder

decoder

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. *Attention is all you need*. In NEURIPS 2017, 2017.

AtmoRep: forecasting and climate projections



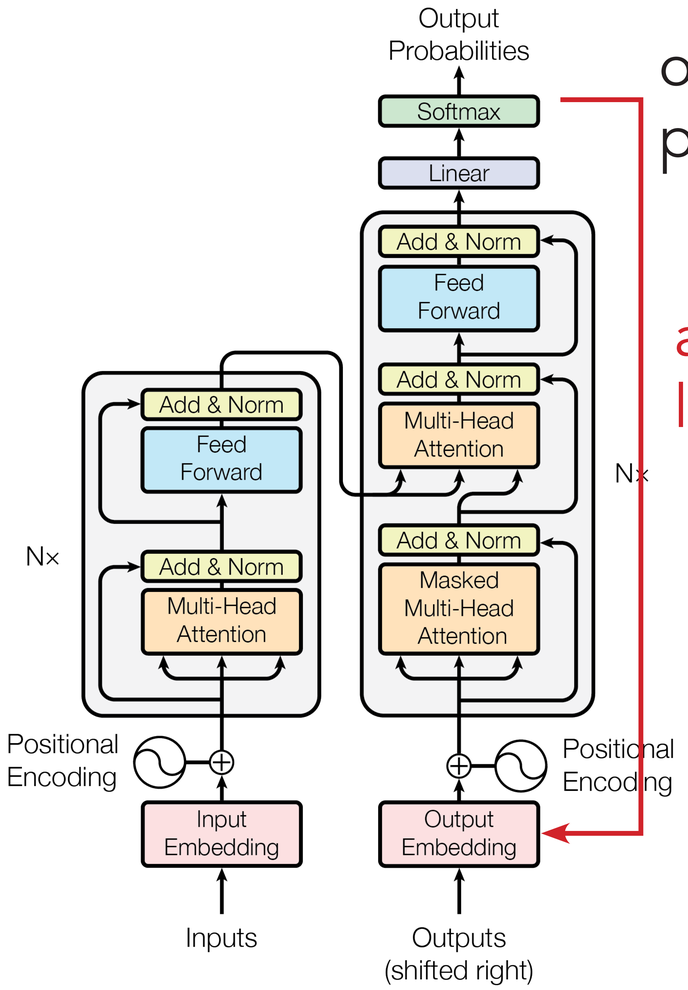
output stream of predictions

encoder

decoder

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. *Attention is all you need*. In NEURIPS 2017, 2017.

AtmoRep: forecasting and climate projections



encoder

decoder

output stream of predictions

akin to time stepping loop in simulations

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. *Attention is all you need*. In NEURIPS 2017, 2017.

Physics and Learning?

Physics and Learning?

Scientific machine learning: use constraints from known models (e.g. symmetries) in the machine learning model

Physics and Learning

Scientific machine learning: use constraints from known models (e.g. symmetries) in the machine learning model

- Hard constraints typically severly degrade the efficiency of the learning
- Respect the physics with architectural choices without being too rigid?

Observational data: avoid the (inductive) biases and constraints we have in analytic models in the learned model

Physics and Learning

Scientific machine learning: use constraints from known models (e.g. symmetries) in the machine learning model

- Hard constraints typically severly degrade the efficiency of the learning
- Respect the physics with architectural choices without being too rigid?

Observational data: avoid the (inductive) biases and constraints we have in analytic models in the learned model

- Use neural networks where reductionist models are no longer effectiv
- Provide new understanding of physics?

Summary

- Self-supervised representation learning has (in my opinion) great, untapped potential
  - Large amounts of unlabeled data (and fast growing)
  - Labeled data is scarce and difficult to obtain

Summary

- Self-supervised representation learning has (in my opinion) great, untapped potential
  - Large amounts of unlabeled data (and fast growing)
  - Labeled data is scarce and difficult to obtain
- Transformers are a versatile and powerful architecture
  - Largely unexplored for science and engineering
  - Fits natural with scientific computing

Self-supervised representation learning

- DINO[1]
  - Self-supervised representation learning for computer vision tasks
  - Vision transformer as neural network
  - Training with unlabeled ImageNet dataset
  - Student-teacher training with virtual prediction task

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

Self-supervised representation learning

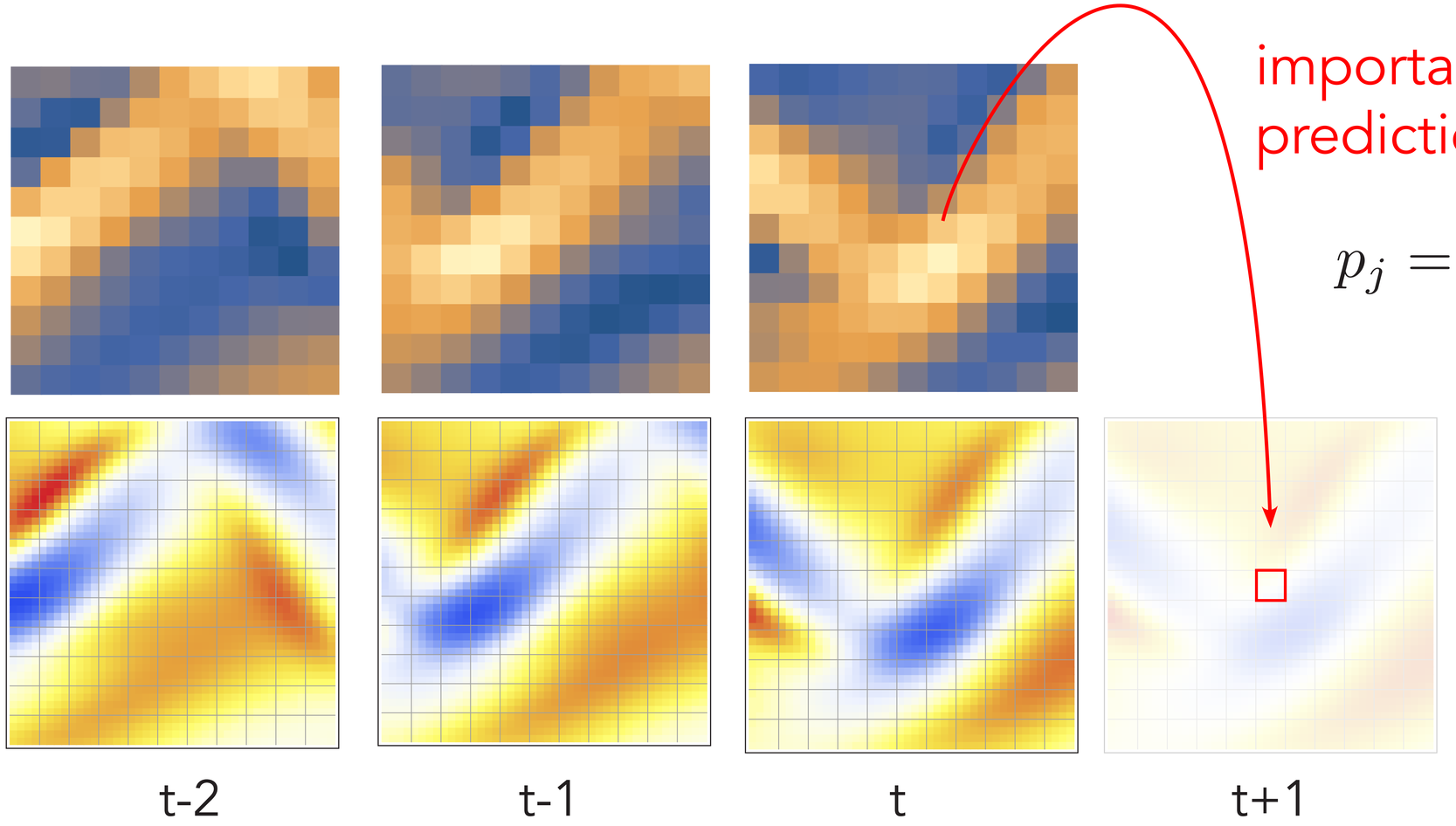
- DINO[1]

	Cifar_{10}	Cifar_{100}	INat_{18}	INat_{19}	Flwrs	Cars	INet
ViT-S/16							
Sup. [69]	**99.0**	89.5	70.7	76.6	98.2	92.1	79.9
DINO	**99.0**	**90.5**	**72.0**	**78.2**	**98.5**	**93.0**	**81.5**
ViT-B/16							
Sup. [69]	99.0	90.8	**73.2**	77.7	98.4	92.1	81.8
DINO	**99.1**	**91.7**	72.6	**78.6**	**98.8**	**93.0**	**82.8**

Performance of fine-tuned model on classification

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

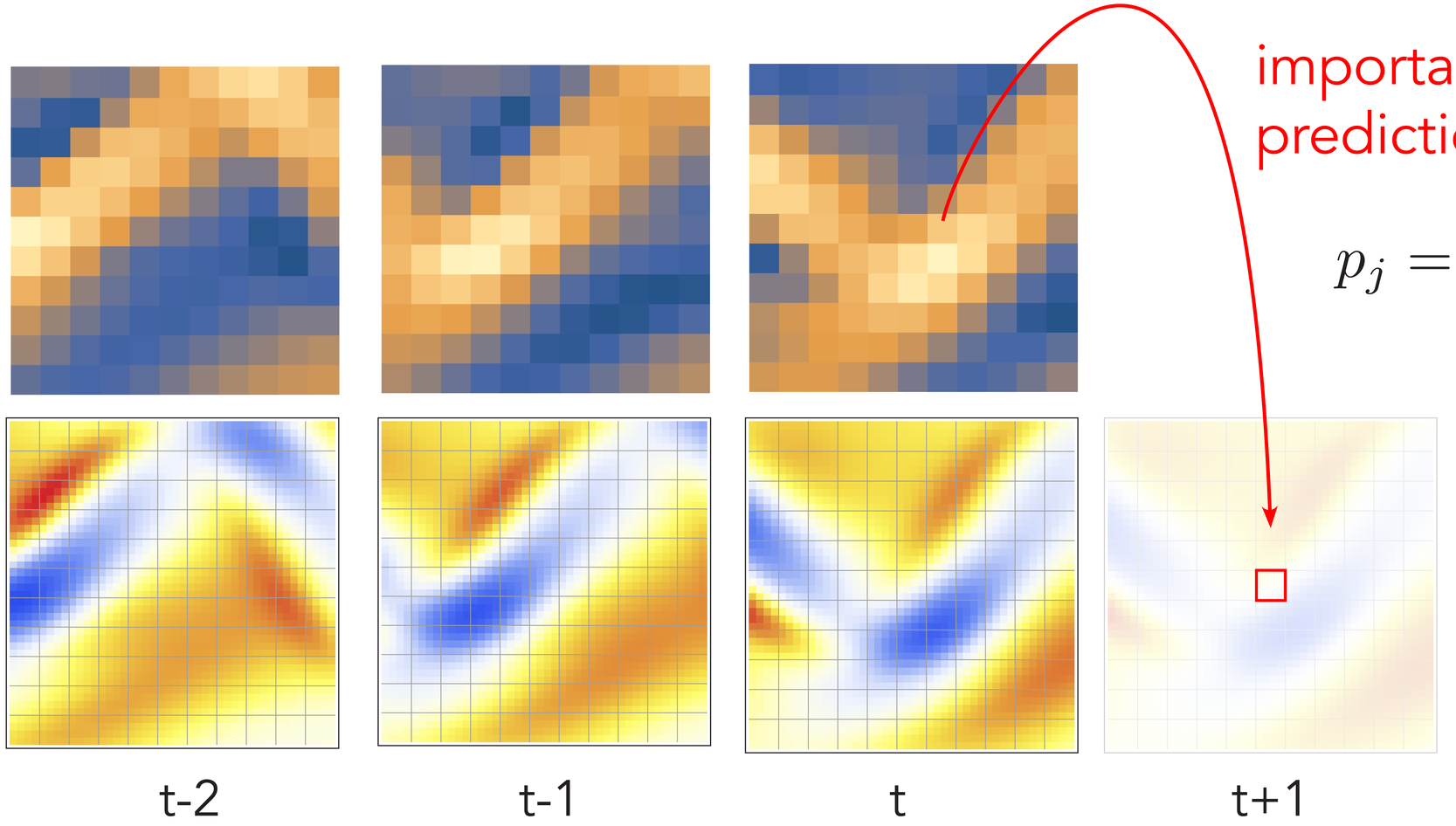
Fluid flow



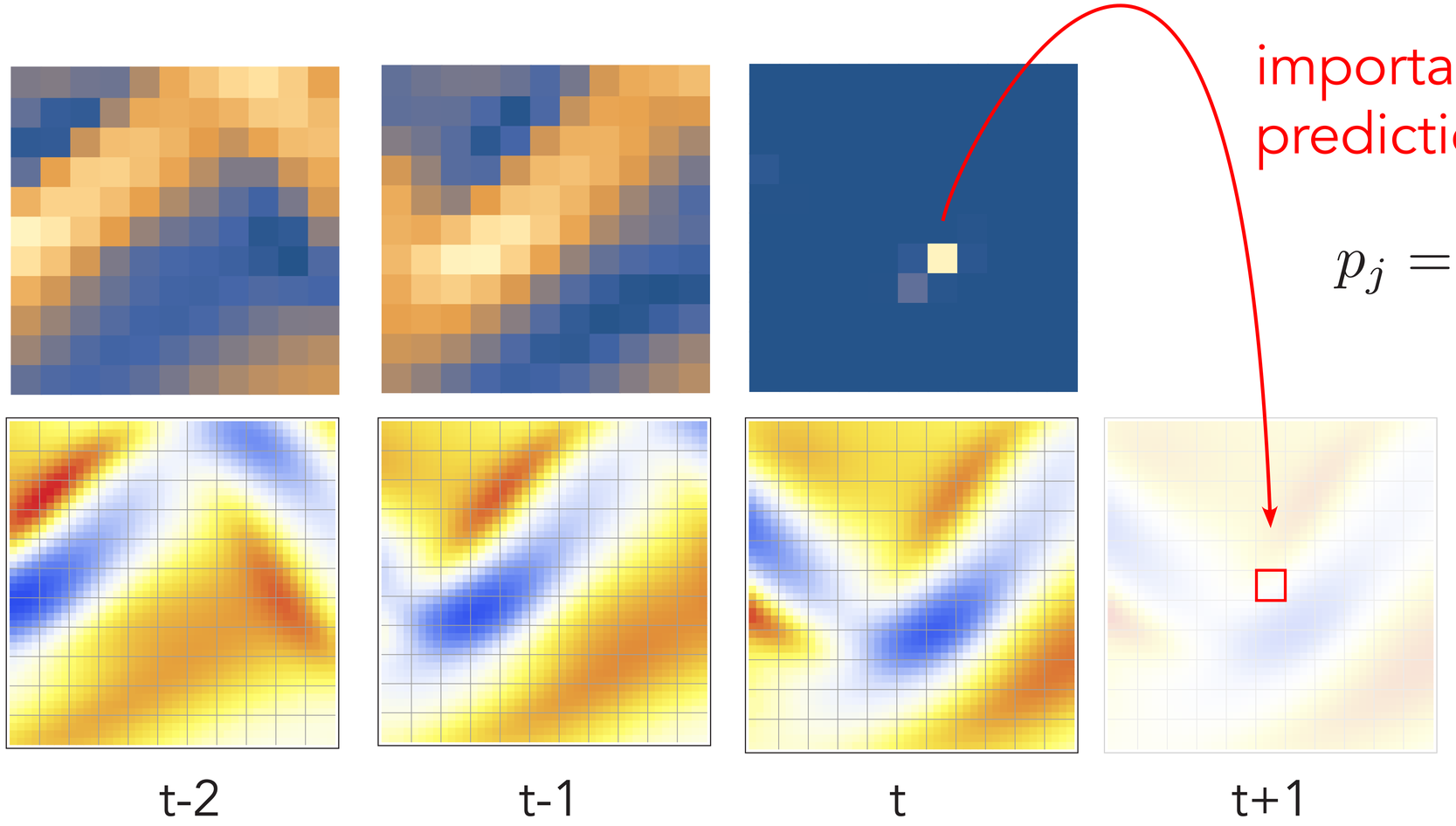
important for prediction

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

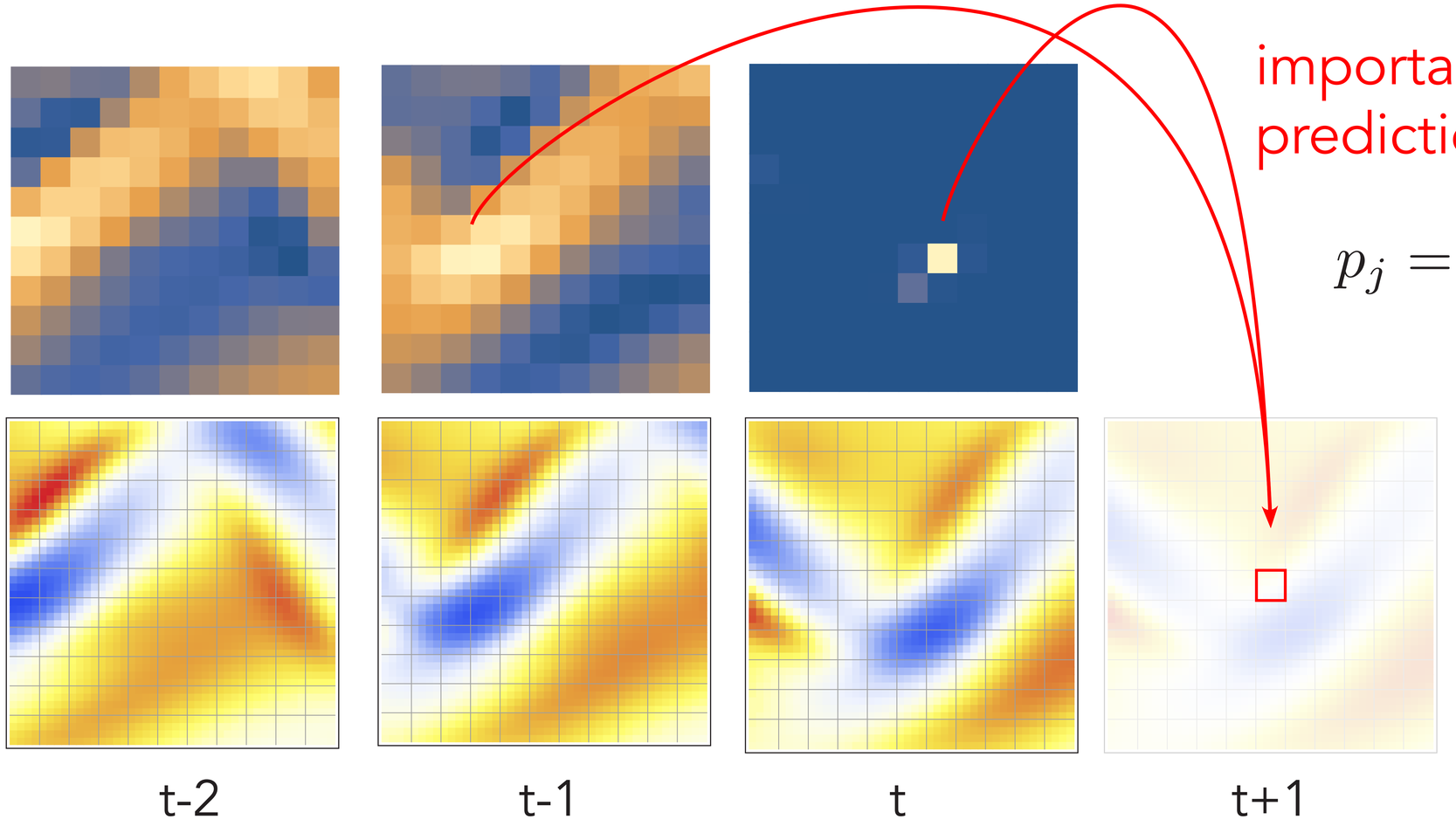
Fluid flow



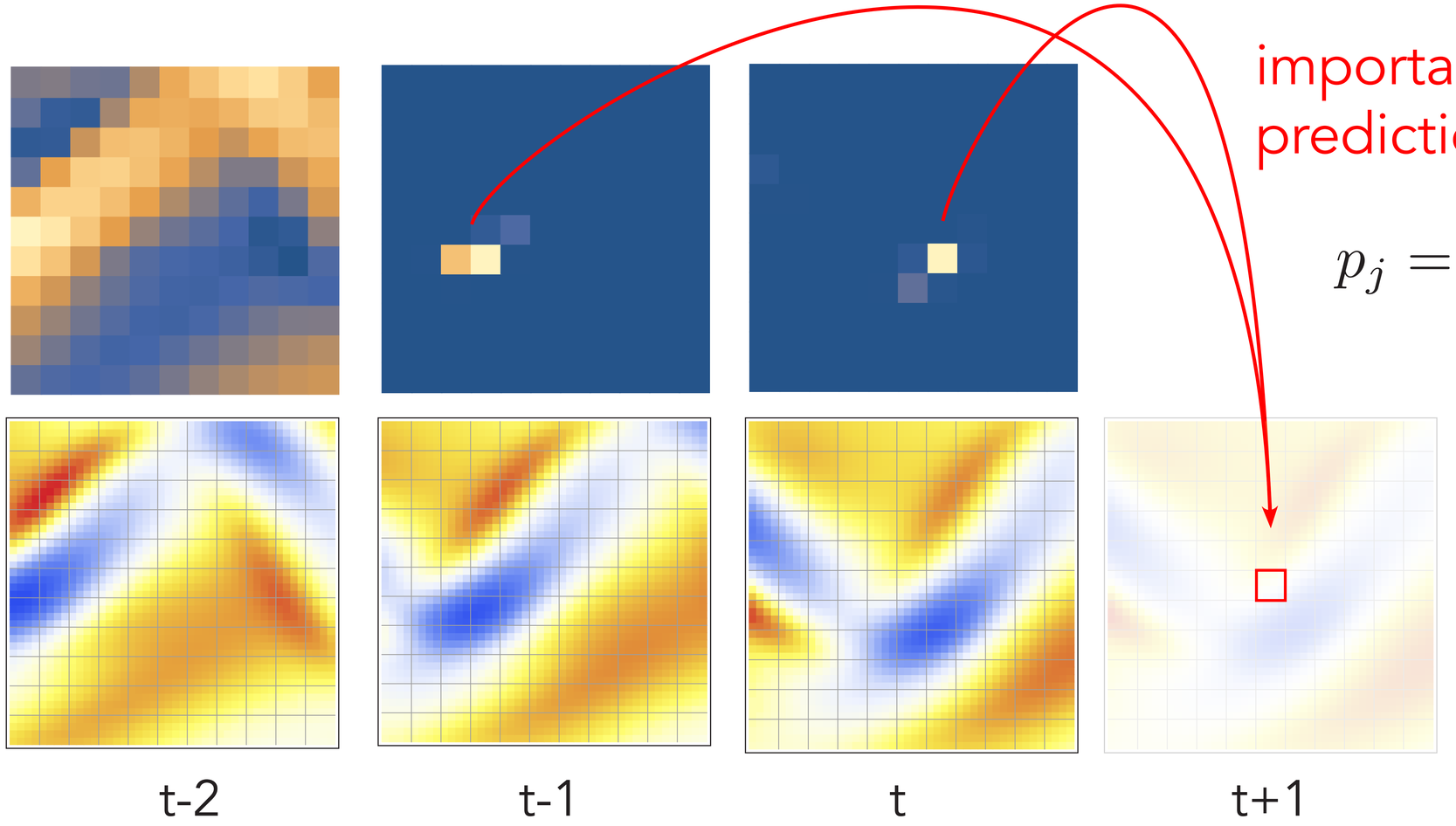
Fluid flow



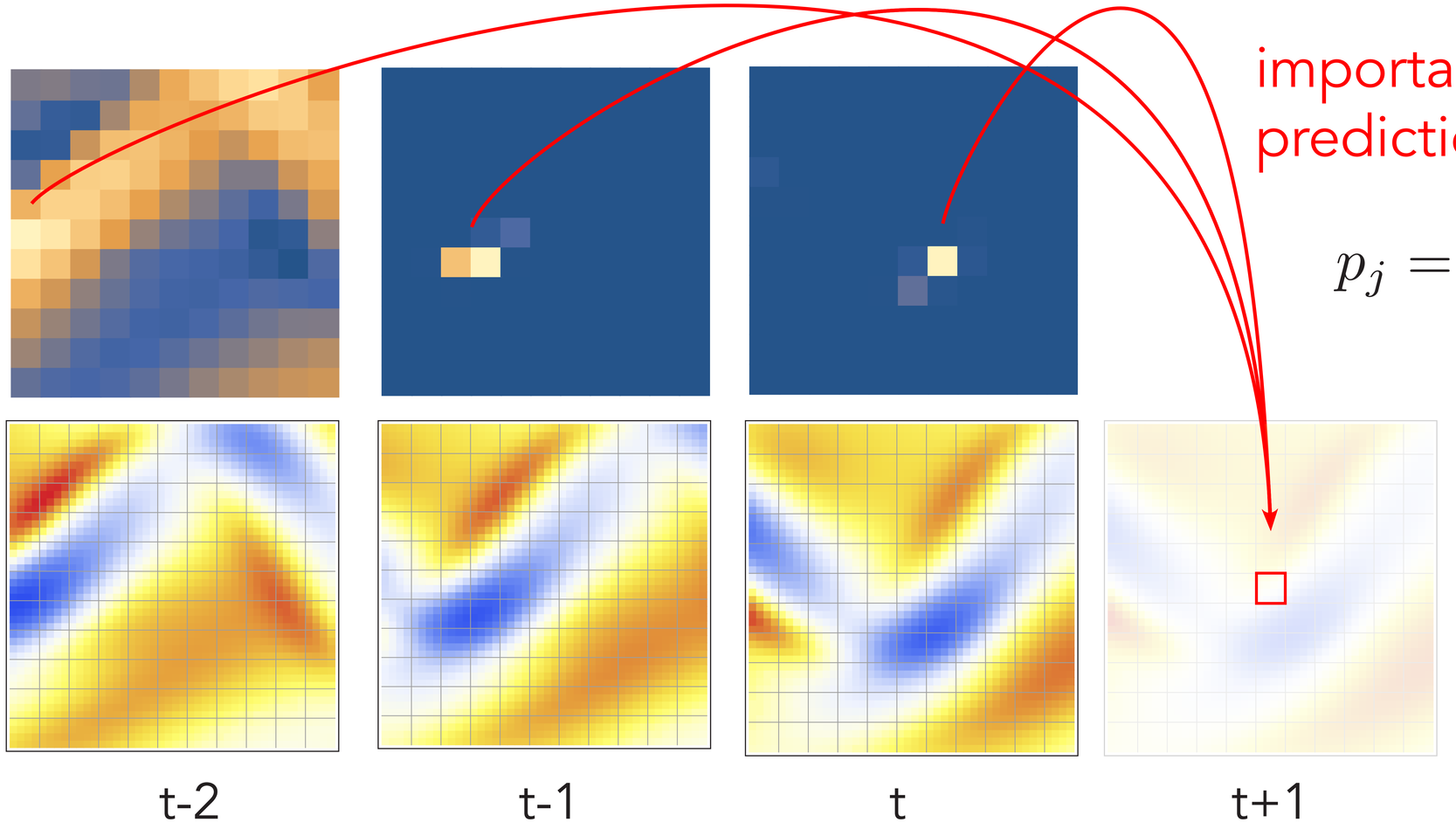
Fluid flow



Fluid flow



Fluid flow



important for prediction

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

attention map values

Fluid flow