Transformer Neural Networks

Christian Lessig
christian.lessig@ecmwf.int

ECMWF

Introduction

"Our work reinforces the bitter lesson. The most important factors determining the performance of a sensibly designed model are the compute and data available for training. […]"

Smith et al., ConvNets match Vision Transformers at Scale, https://arxiv.org/pdf/2310.16764.pdf

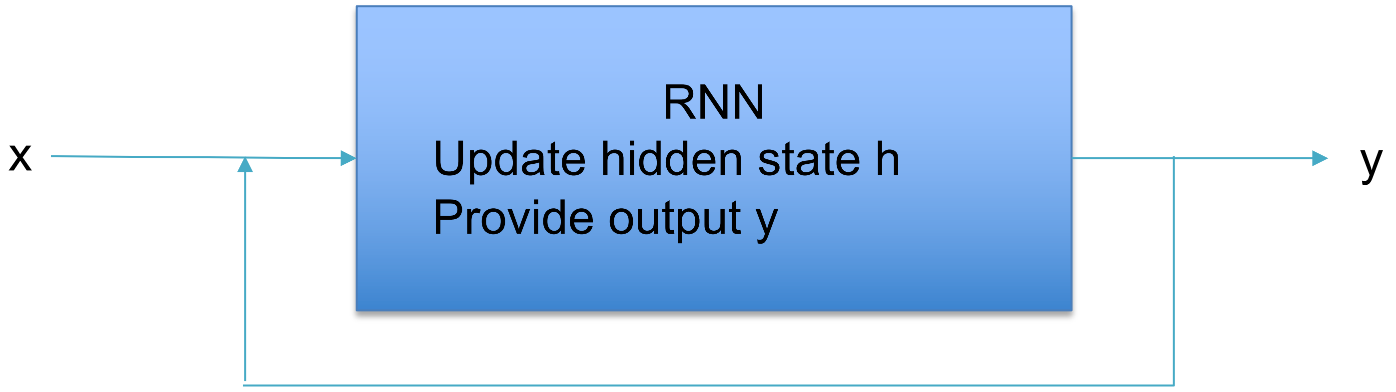
Introduction

"Our work reinforces the bitter lesson. The most important factors determining the performance of a sensibly designed model are the compute and data available for training. [...]"

Smith et al., ConvNets match Vision Transformers at Scale, https://arxiv.org/pdf/2310.16764.pdf

Motivation

- Recurrent neural networks
  - Standard for temporal sequence problems (e.g. in natural language processing up to 2018)



- Training is difficult to parallelize
- Implicit connection to past states

Motivation

- Architecture that can be parallelized more efficiently
- More direct interaction between information, in particular “far away” one

Attention Is All You Need

Ashish Vaswani* Google Brain avaswani@google.com

Noam Shazeer* Google Brain noam@google.com

Niki Parmar* Google Research nikip@google.com

Jakob Uszkoreit* Google Research usz@google.com

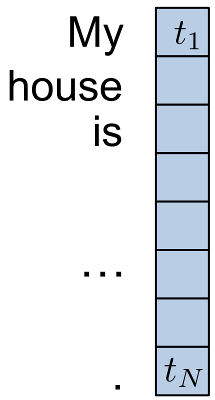
Llion Jones* Google Research

Aidan N. Gomez* † University of Toronto

Łukasz Kaiser* Google Brain

What is attention?

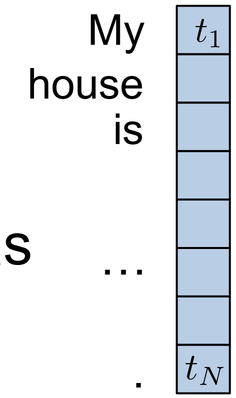
- Similarity measure between hidden/latent states $\{t_i\}_{i=1}^{N}$
  - Hidden/latent states are vectors in $\mathbb{R}^E$



What is attention?

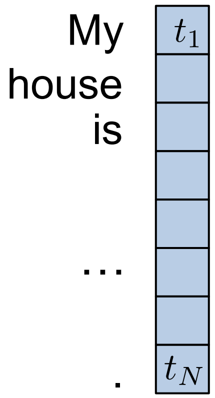
- Similarity measure between hidden/latent states $\{t_i\}_{i=1}^{N}$
  - Hidden/latent states are vectors in $\mathbb{R}^E$

t_i are known as **tokens**



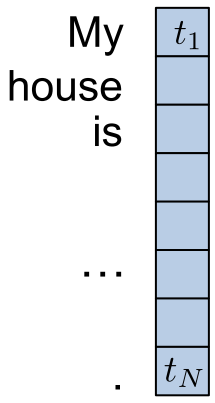
What is attention?

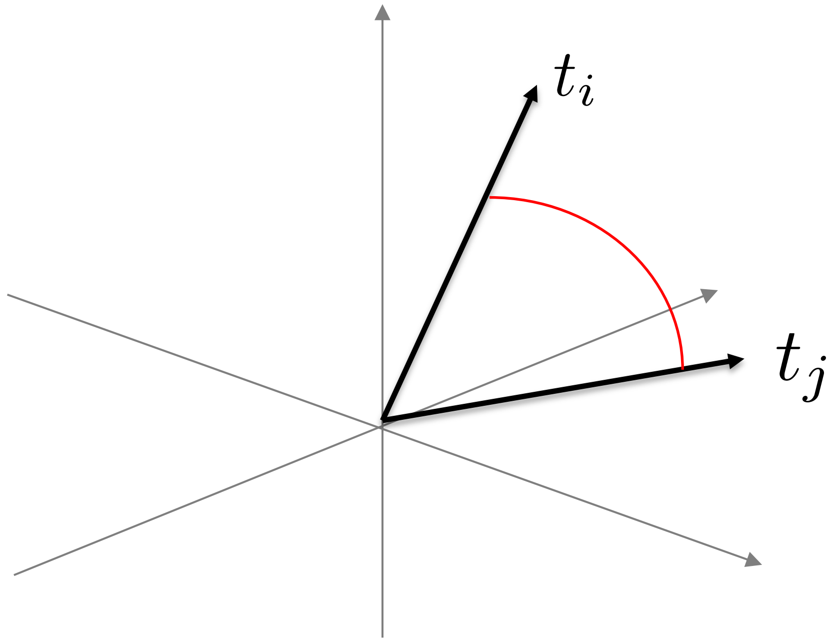
- Similarity measure between hidden/latent states $\{t_i\}_{i=1}^{N}$
  - Hidden/latent states are vectors in $\mathbb{R}^E$
  - Use usual dot product to measure similarity



What is attention?

- Similarity measure between hidden/latent states $\{t_i\}_{i=1}^N$
  - Hidden/latent states are vectors in $\mathbb{R}^E$
  - Use usual dot product to measure similarity





What is attention?

- Similarity measure between hidden/latent states $\{t_i\}_{i=1}^N$
  - Hidden/latent states are vectors in $\mathbb{R}^E$
  - Use usual dot product to measure similarity
  - Update hidden/latent state based on similarities

$$t_i \cdot t_j$$

What is attention?

- Similarity measure between hidden/latent states $\{t_i\}_{i=1}^{N}$
  - Hidden/latent states are vectors in $\mathbb{R}^E$
  - Use usual dot product to measure similarity
  - Update hidden/latent state based on similarities

$$\tilde{t}_i = \sum_{j=1}^{N} (t_i \cdot t_j) t_j$$

What is attention?

- Similarity measure between hidden/latent states $\{t_i\}_{i=1}^{N}$
  - Hidden/latent states are vectors in $\mathbb{R}^E$
  - Use usual dot product to measure similarity
  - Update hidden/latent state based on similarities

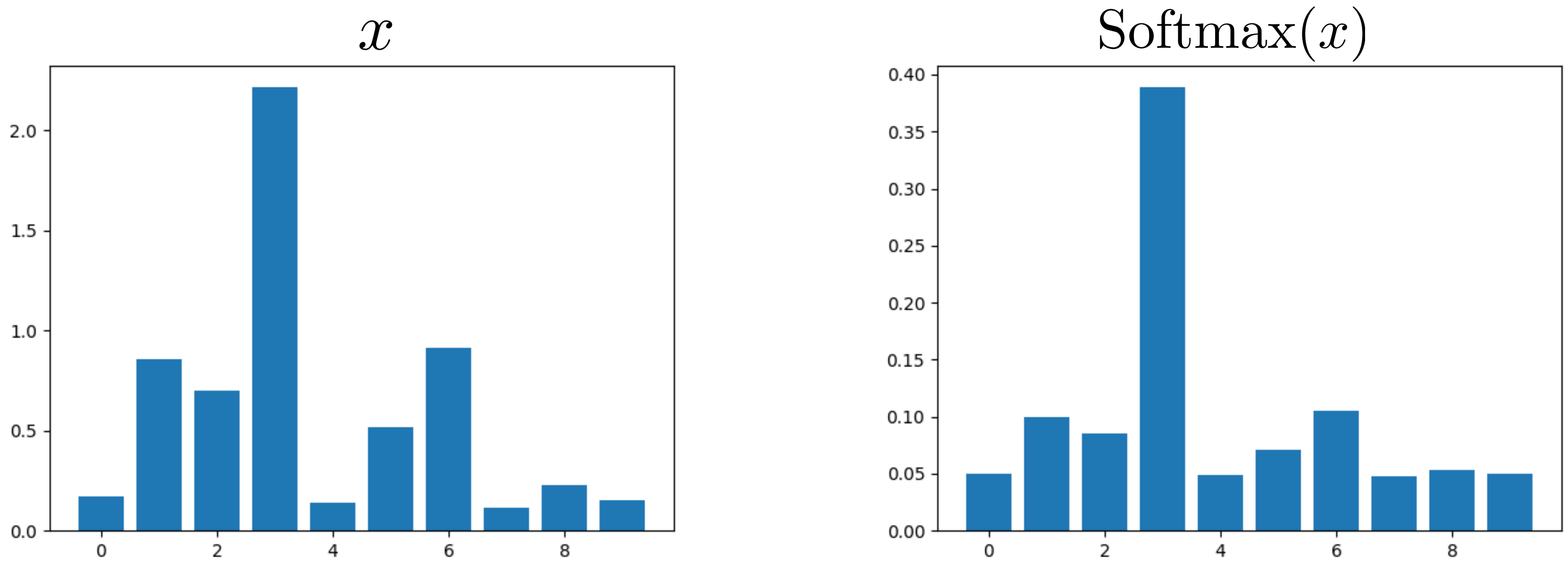
$$\tilde{t}_i = \sum_{j=1}^{N} \sigma(t_i \cdot t_j) t_j$$

softmax nonlinearity
(smoothed/differentiable version of argmax + normalization)

What is attention?

- Softmax

$$\mathrm{Softmax}(x)_i = \frac{e^{x_i}}{\sum_j e^{x_j}} \quad , \; x \in \mathbb{R}^n$$



What is attention?

- Similarity measure between hidden/latent states $\{t_i\}_{i=1}^N$
  - Hidden/latent states are vectors in $\mathbb{R}^E$
  - Use usual dot product to measure similarity
  - Update hidden/latent state based on similarities

$$\tilde{t}_i = \sum_{j=1}^{N} \sigma(t_i \cdot t_j) t_j$$

softmax nonlinearity
(smoothed/differentiable version of argmax + normalization)

What is attention?

- Similarity measure between hidden/latent states $\{t_i\}_{i=1}^N$
  - Hidden/latent states are vectors in $\mathbb{R}^E$
  - Use usual dot product to measure similarity
  - Update hidden/latent state based on similarities

$$\tilde{t}_i = \sum_{j=1}^{N} \sigma(t_i \cdot t_j) t_j$$

How to make this “learnable”?

softmax nonlinearity
(smoothed/differentiable version of argmax + normalization)

What is attention?

- Similarity measure between hidden/latent states $\{t_i\}_{i=1}^N$
  - Hidden/latent states are vectors in $\mathbb{R}^E$
  - Use usual dot product to measure similarity
  - Update hidden/latent state based on similarities

$$q_i = P_q t_i$$
$$k_i = P_k t_i$$
$$v_i = P_v t_i$$

$$\tilde{t}_i = \sum_{j=1}^{N} \sigma(t_i \cdot t_j) t_j$$

How to make this "learnable"?

softmax nonlinearity
(smoothed/differentiable version of argmax + normalization)

What is attention?

- Similarity measure between hidden/latent states $\{t_i\}_{i=1}^{N}$
  - Hidden/latent states are vectors in $\mathbb{R}^E$
  - Use usual dot product to measure similarity
  - Update hidden/latent state based on similarities

$$q_i = P_q t_i$$
$$k_i = P_k t_i$$
$$v_i = P_v t_i$$

$$\tilde{t}_i = \sum_{j=1}^{N} \sigma(q_i \cdot k_j) v_j$$

softmax nonlinearity
(smoothed/differentiable version of argmax + normalization)

How to make this "learnable"?

What is attention?

- Similarity measure between hidden/latent states $\{t_i\}_{i=1}^{N}$
  - Hidden/latent states are vectors in $\mathbb{R}^E$
  - Use usual dot product to measure similarity
  - Update hidden/latent state based on similarities

$$q_i = P_q t_i$$
$$k_i = P_k t_i$$
$$v_i = P_v t_i$$

$$\tilde{t}_i = \sum_{j=1}^{N} \sigma(q_i \cdot k_j) v_j$$

Learnable attention module

softmax nonlinearity
(smoothed/differentiable version of argmax + normalization)

What is attention?

- Similarity measure between hidden/latent states $\{t_i\}_{i=1}^N$
  - Hidden/latent states are vectors in $\mathbb{R}^E$
  - Use usual dot product to measure similarity
  - Update hidden/latent state based on similarities

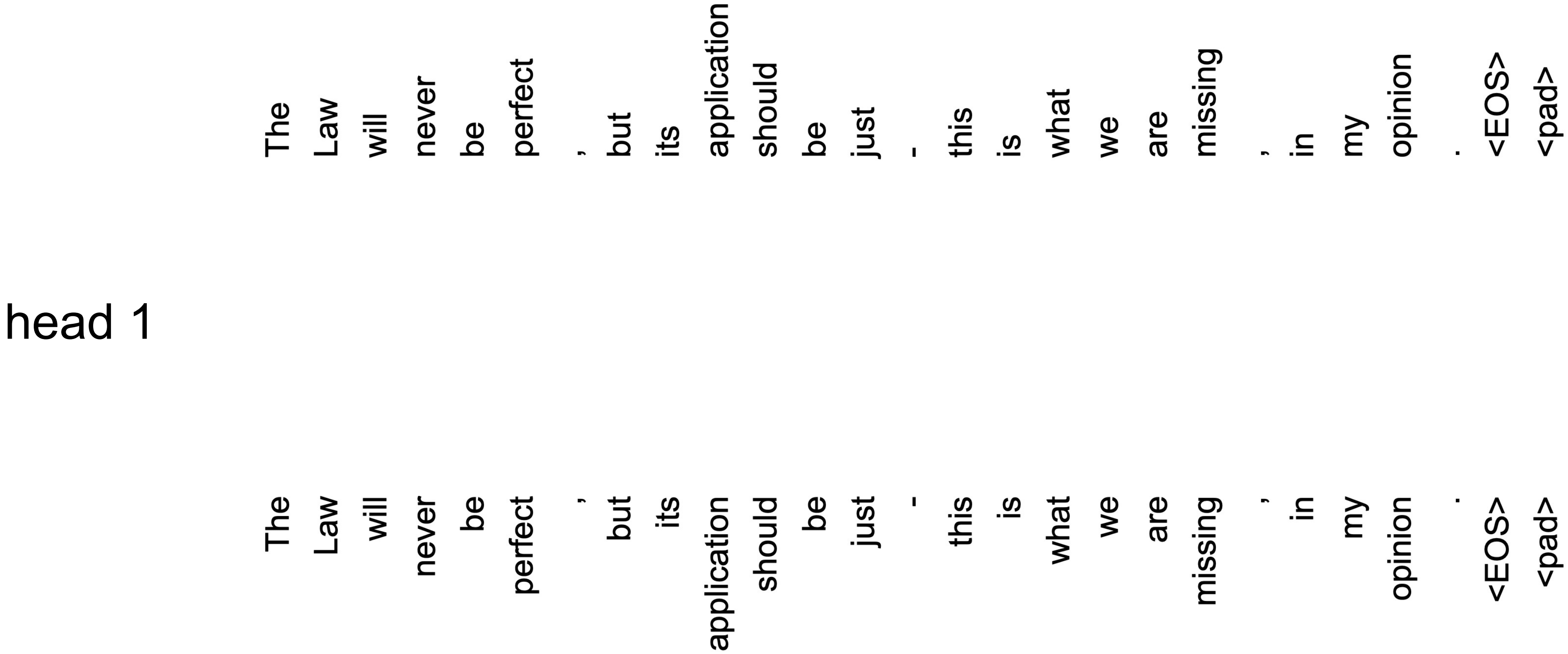
$$q_i^h = P_q^h t_i$$
$$k_i^h = P_k^h t_i$$
$$v_i^h = P_v^h t_i$$

$$\tilde{t}_i = \sum_{j=1}^{N} \sigma(q_i \cdot k_j) v_j$$

softmax nonlinearity
(smoothed/differentiable version of argmax + normalization)

Learnable attention module with multiple heads

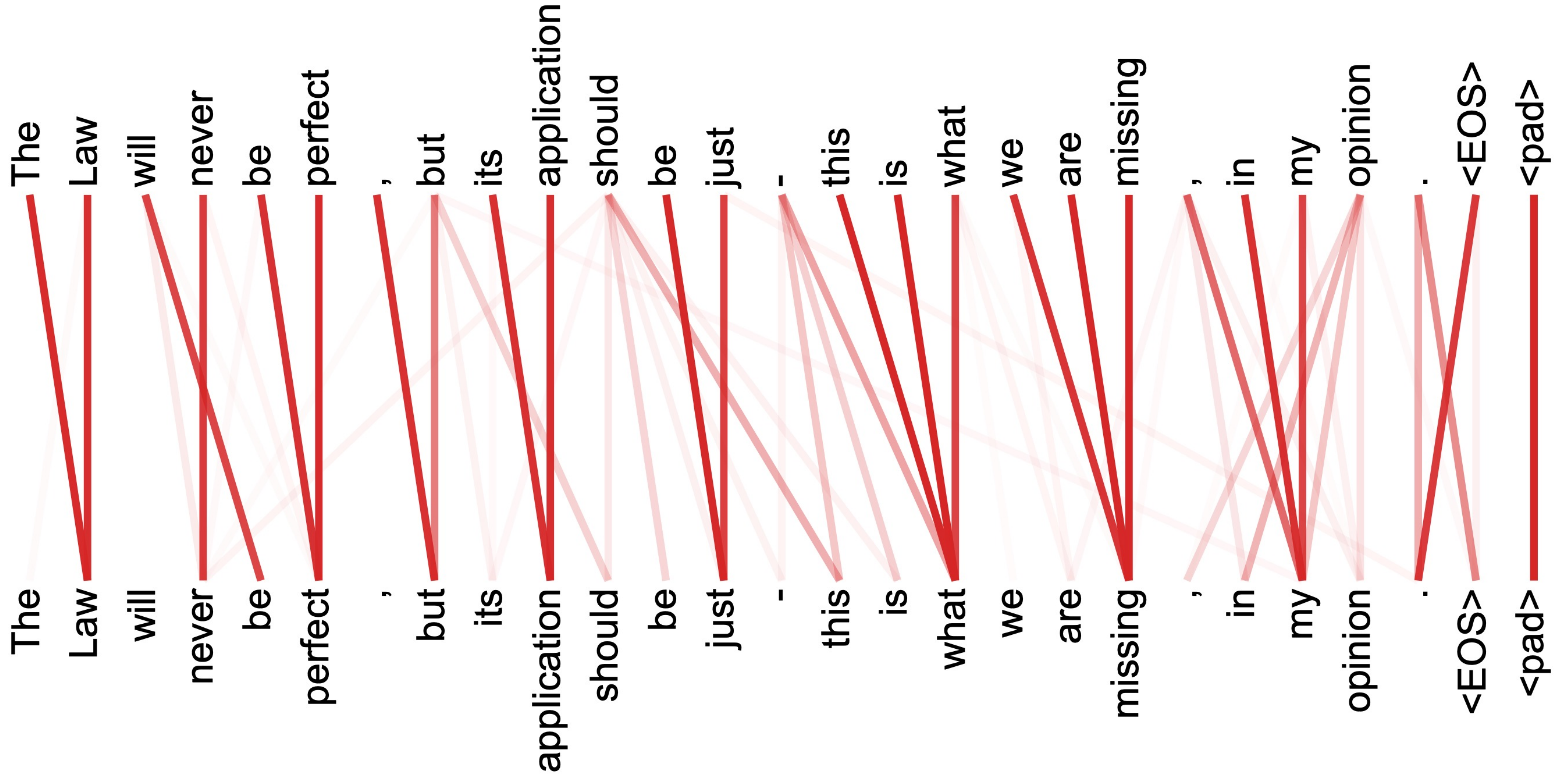
What is attention?



Vaswani et al., Attention is all you need, 2017, https://arxiv.org/pdf/2310.16764.pdf

What is attention?

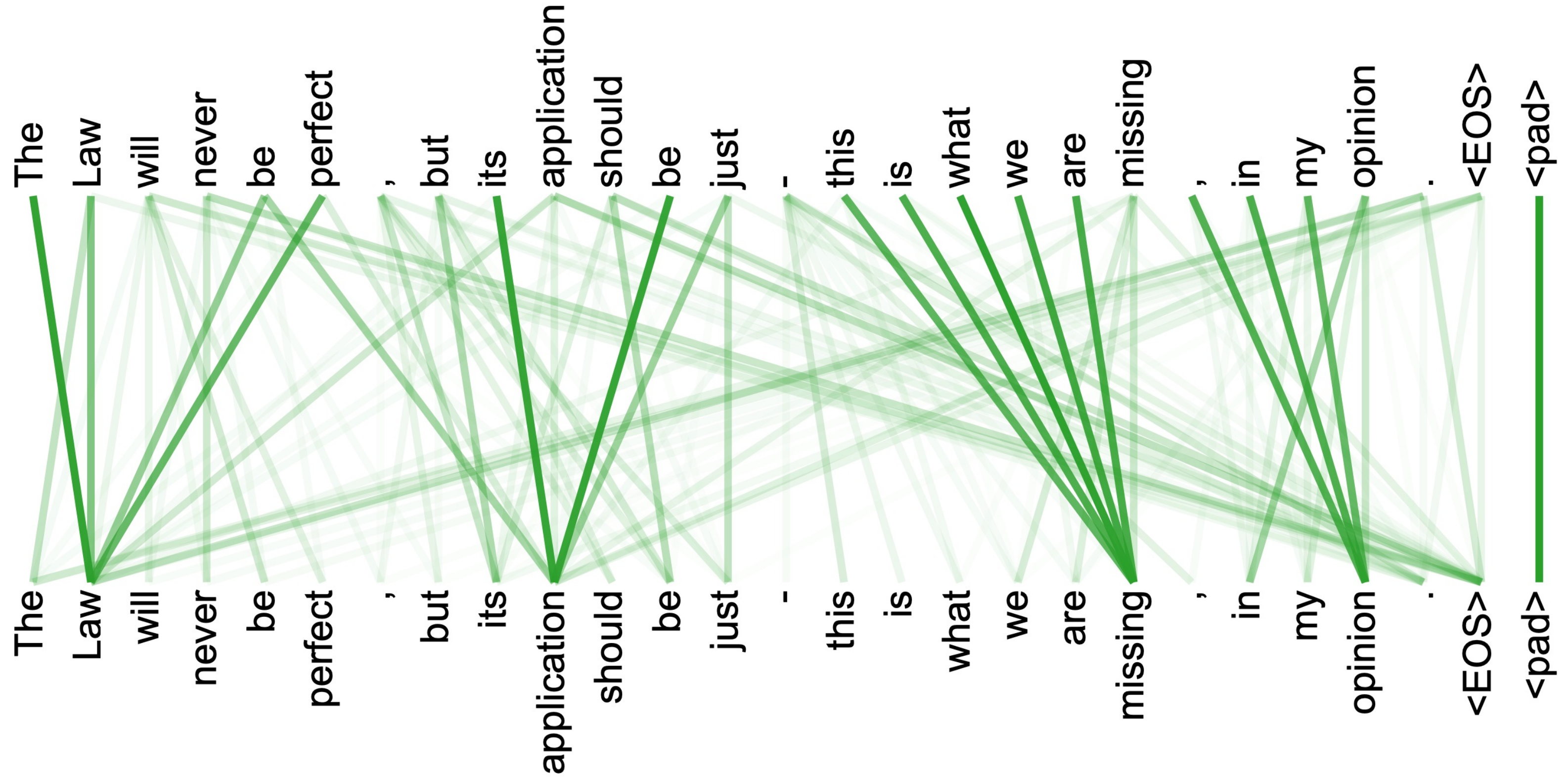
head 1



Vaswani et al., Attention is all you need, 2017, https://arxiv.org/pdf/2310.16764.pdf

What is attention?

head 2



Vaswani et al., Attention is all you need, 2017, https://arxiv.org/pdf/2310.16764.pdf

What is attention?

- Similarity measure between hidden/latent states $\{t_i\}_{i=1}^N$
  - Hidden/latent states are vectors in $\mathbb{R}^E$
  - Use usual dot product to measure similarity
  - Update hidden/latent state based on similarities

$$q_i^h = P_q^h t_i$$
$$k_i^h = P_k^h t_i$$
$$v_i^h = P_v^h t_i$$

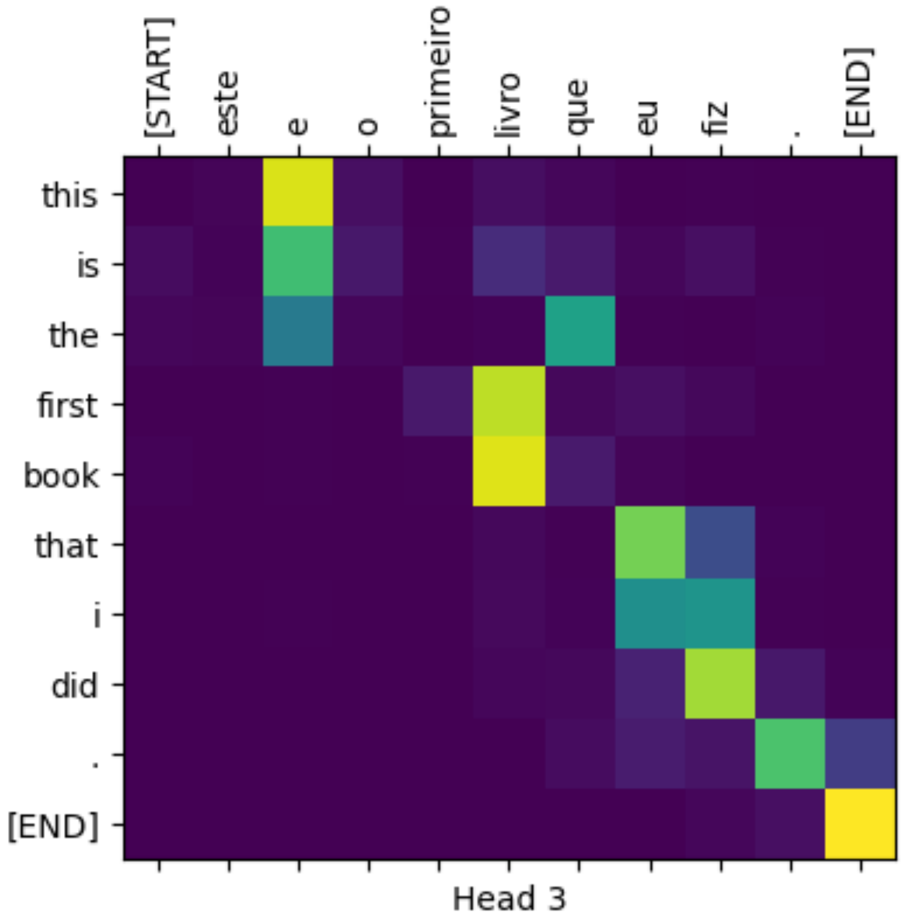
attention matrix

$$\tilde{t}_i = \sum_{j=1}^{N} \sigma(q_i \cdot k_j) v_j$$

softmax nonlinearity
(smoothed/differentiable version of argmax + normalization)

Learnable attention module with multiple heads

What is attention?



What is attention?

- Similarity measure between hidden/latent states $\{t_i\}_{i=1}^{N}$
  - Hidden/latent states are vectors in $\mathbb{R}^E$
  - Use usual dot product to measure similarity
  - Update hidden/latent state based on similarities

$$q_i^h = P_q^h t_i$$
$$k_i^h = P_k^h t_i$$
$$v_i^h = P_v^h t_i$$

attention matrix

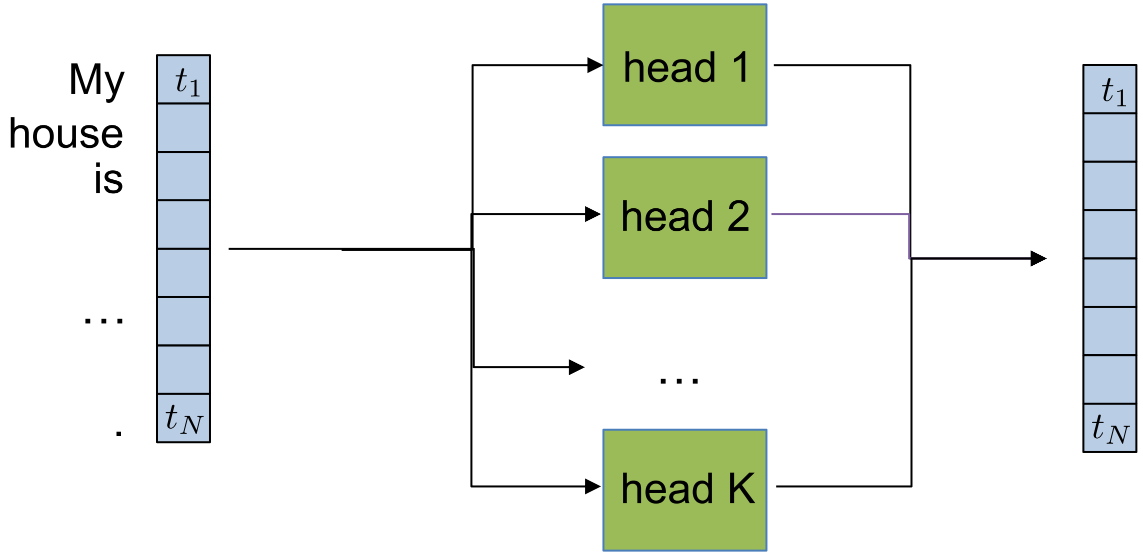
$$\tilde{t}_i = \sum_{j=1}^{N} \sigma(q_i \cdot k_j) v_j$$

softmax nonlinearity
(smoothed/differentiable version of argmax + normalization)

Learnable attention module with multiple heads

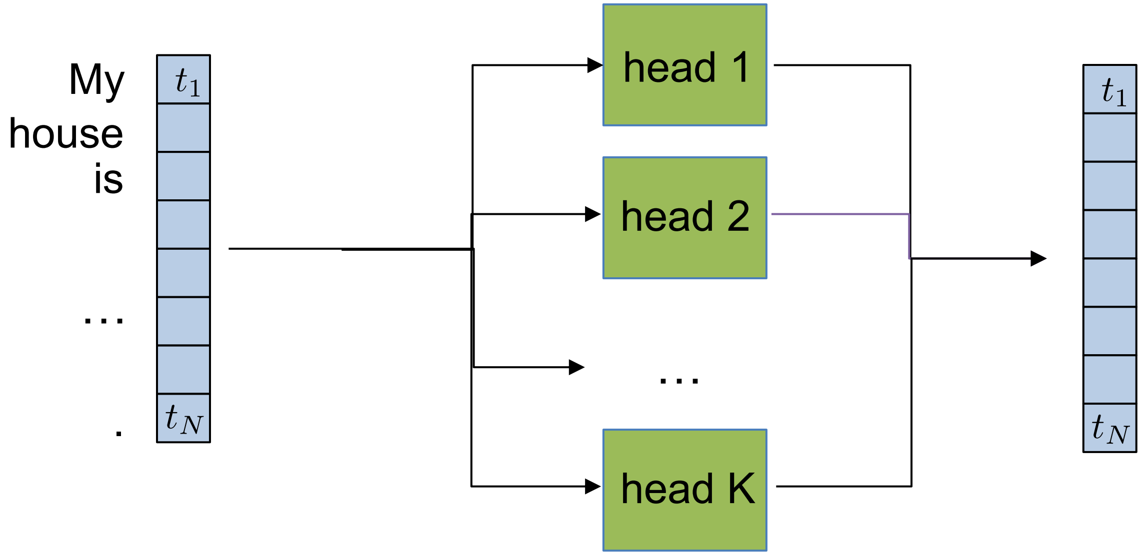
What is attention?

- Similarity measure between hidden/latent states $\{t_i\}_{i=1}^{N}$
  - Hidden/latent states are vectors in $\mathbb{R}^E$
  - Use usual dot product to measure similarity
  - Update hidden/latent state based on similarities



What is attention?

- Similarity measure between hidden/latent states $\{t_i\}_{i=1}^{N}$
  - Hidden/latent states are vectors in $\mathbb{R}^E$
  - Use usual dot product to measure similarity
  - Update hidden/latent state based on similarities



Common choice:

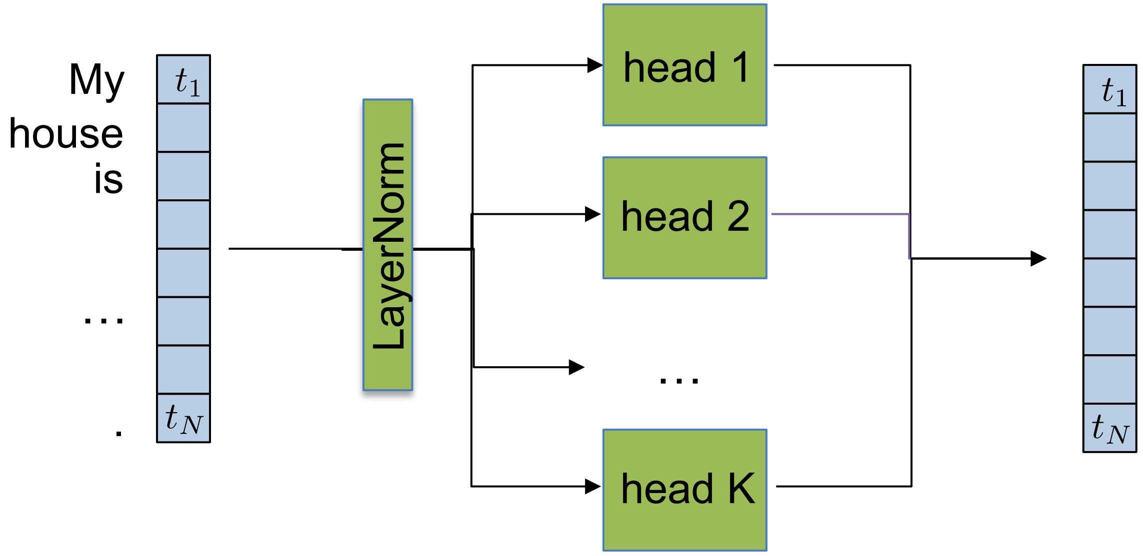
$$t_i \in \mathbb{R}^E$$

$$P_*^h : \mathbb{R}^E \to \mathbb{R}^{E/H}$$

$$\tilde{t}_i = \tilde{P}(\tilde{t}_i^1, \cdots, \tilde{t}_i^H)$$

What is attention?

- Similarity measure between hidden/latent states $\{t_i\}_{i=1}^N$
  - Hidden/latent states are vectors in $\mathbb{R}^E$
  - Use usual dot product to measure similarity
  - Update hidden/latent state based on similarities



Common choice:

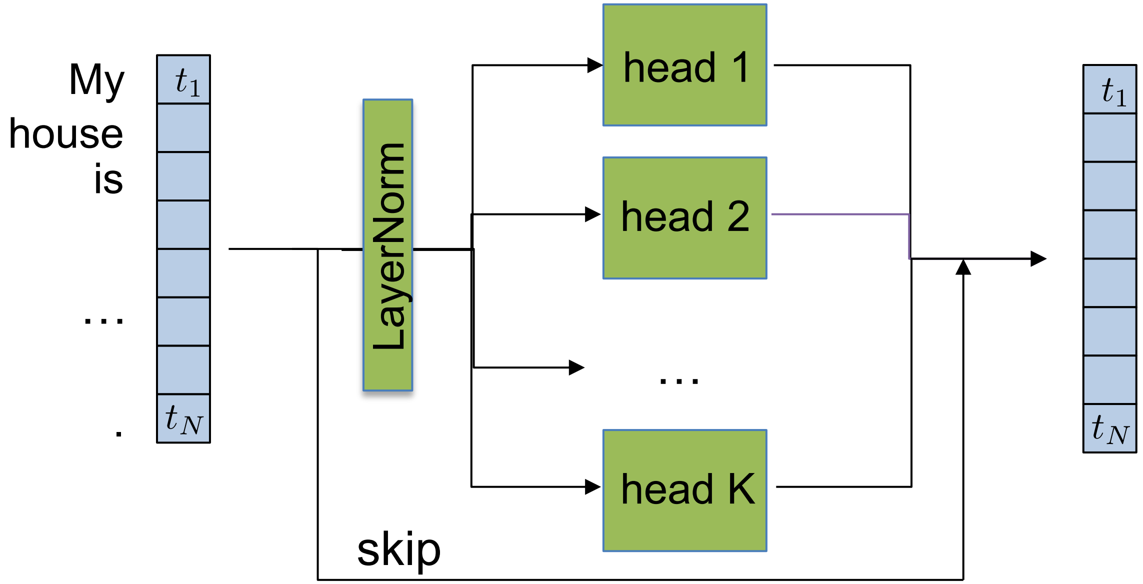
$$t_i \in \mathbb{R}^E$$

$$P_*^h : \mathbb{R}^E \to \mathbb{R}^{E/H}$$

$$\tilde{t}_i = \tilde{P}(\tilde{t}_i^1, \cdots, \tilde{t}_i^H)$$

What is attention?

- Similarity measure between hidden/latent states $\{t_i\}_{i=1}^N$
  - Hidden/latent states are vectors in $\mathbb{R}^E$
  - Use usual dot product to measure similarity
  - Update hidden/latent state based on similarities



Common choice:

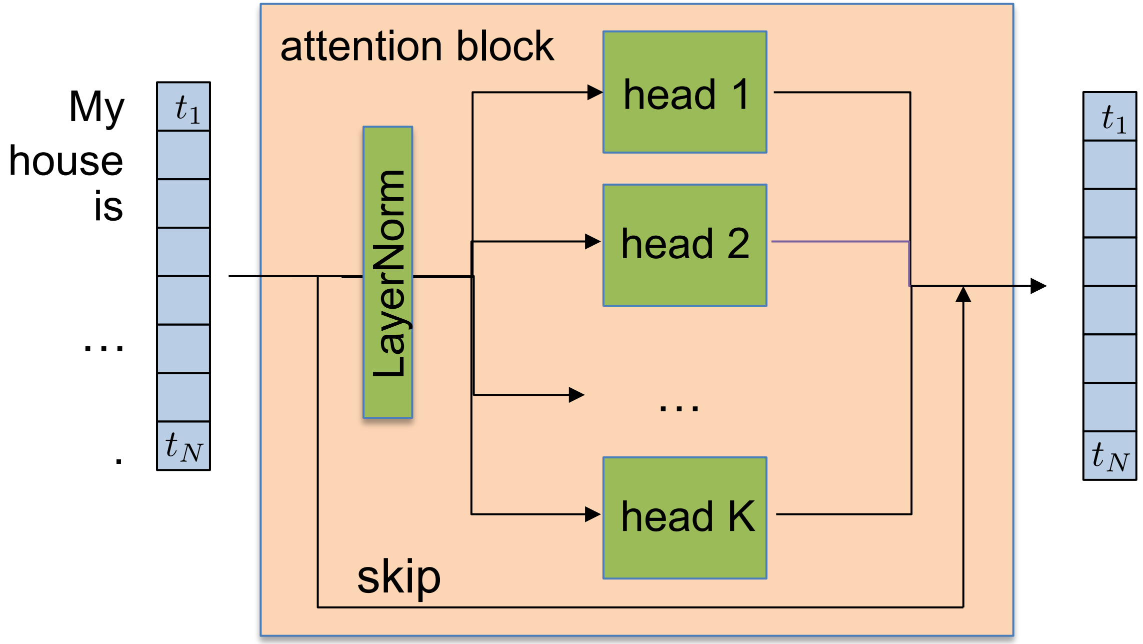
$$t_i \in \mathbb{R}^E$$

$$P_*^h : \mathbb{R}^E \to \mathbb{R}^{E/H}$$

$$\tilde{t}_i = \tilde{P}(\tilde{t}_i^1, \cdots, \tilde{t}_i^H)$$

What is attention?

- Similarity measure between hidden/latent states $\{t_i\}_{i=1}^N$
  - Hidden/latent states are vectors in $\mathbb{R}^E$
  - Use usual dot product to measure similarity
  - Update hidden/latent state based on similarities



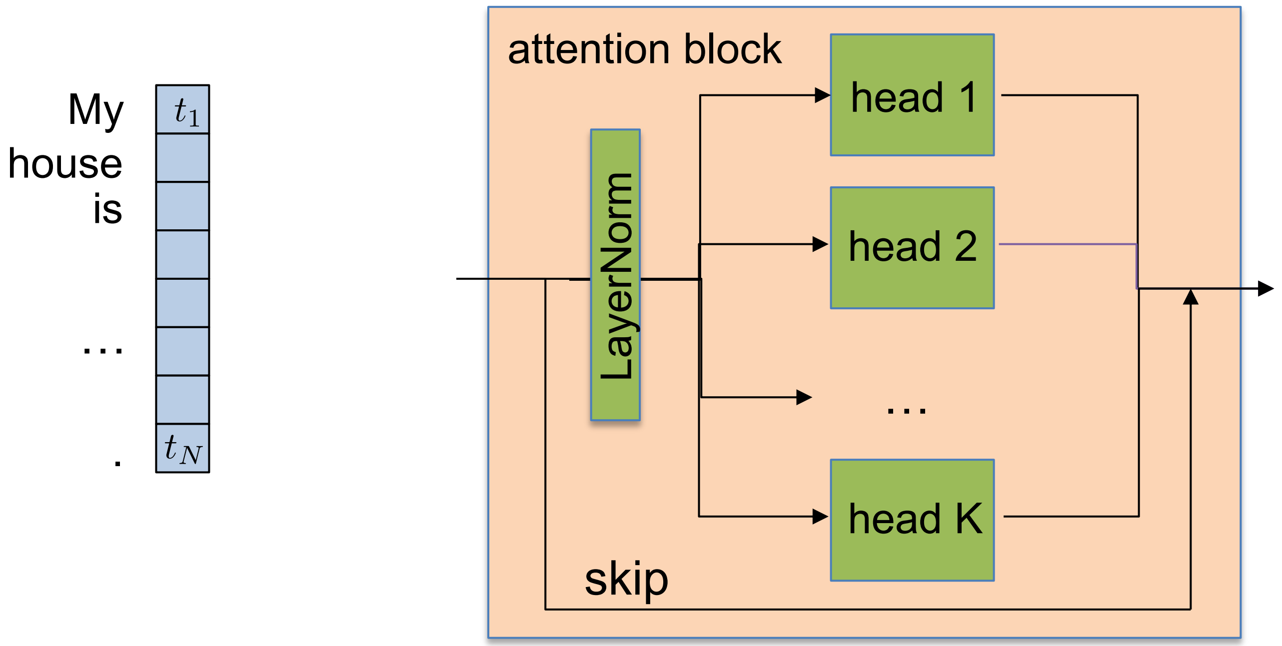
Common choice:

$$t_i \in \mathbb{R}^E$$

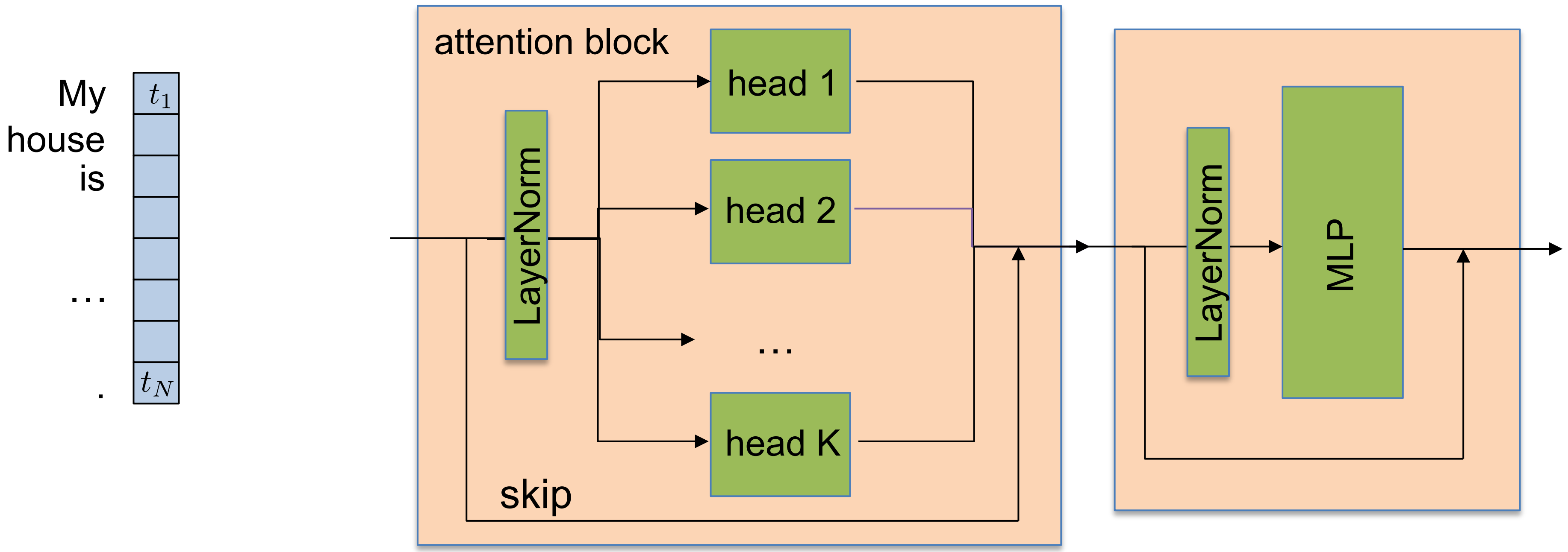
$$P_*^h : \mathbb{R}^E \to \mathbb{R}^{E/H}$$

$$\tilde{t}_i = \tilde{P}(\tilde{t}_i^1, \cdots, \tilde{t}_i^H)$$

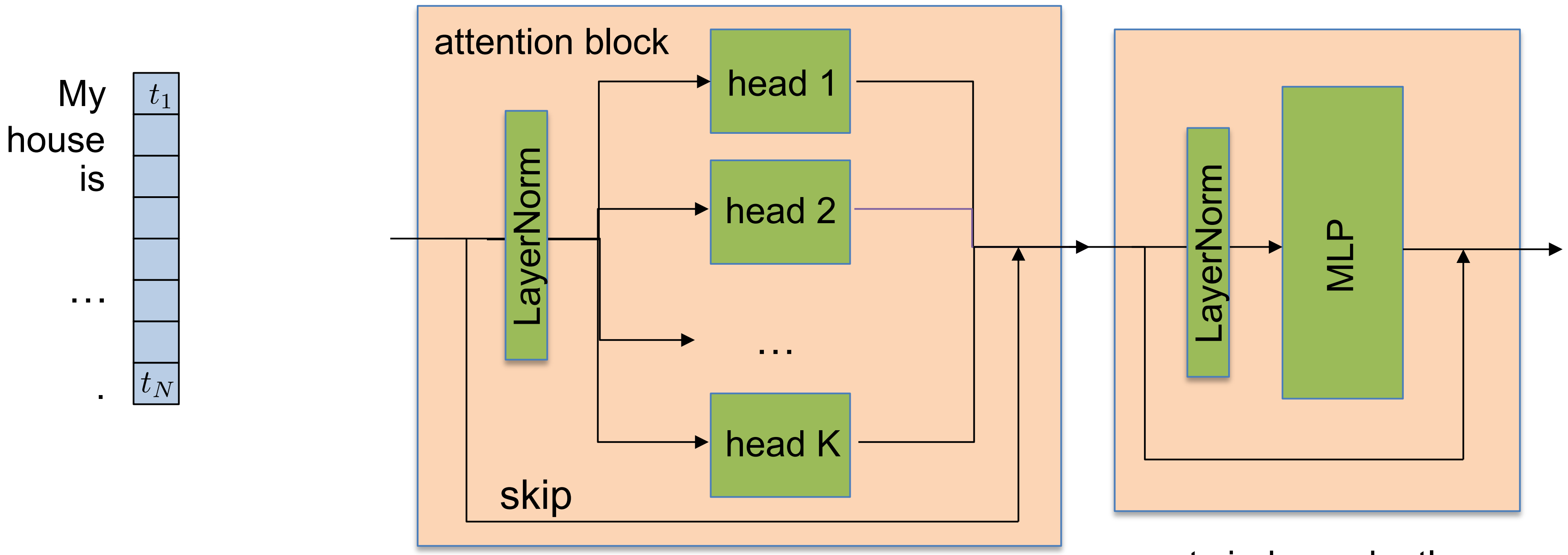
Transformer encoder



Transformer encoder

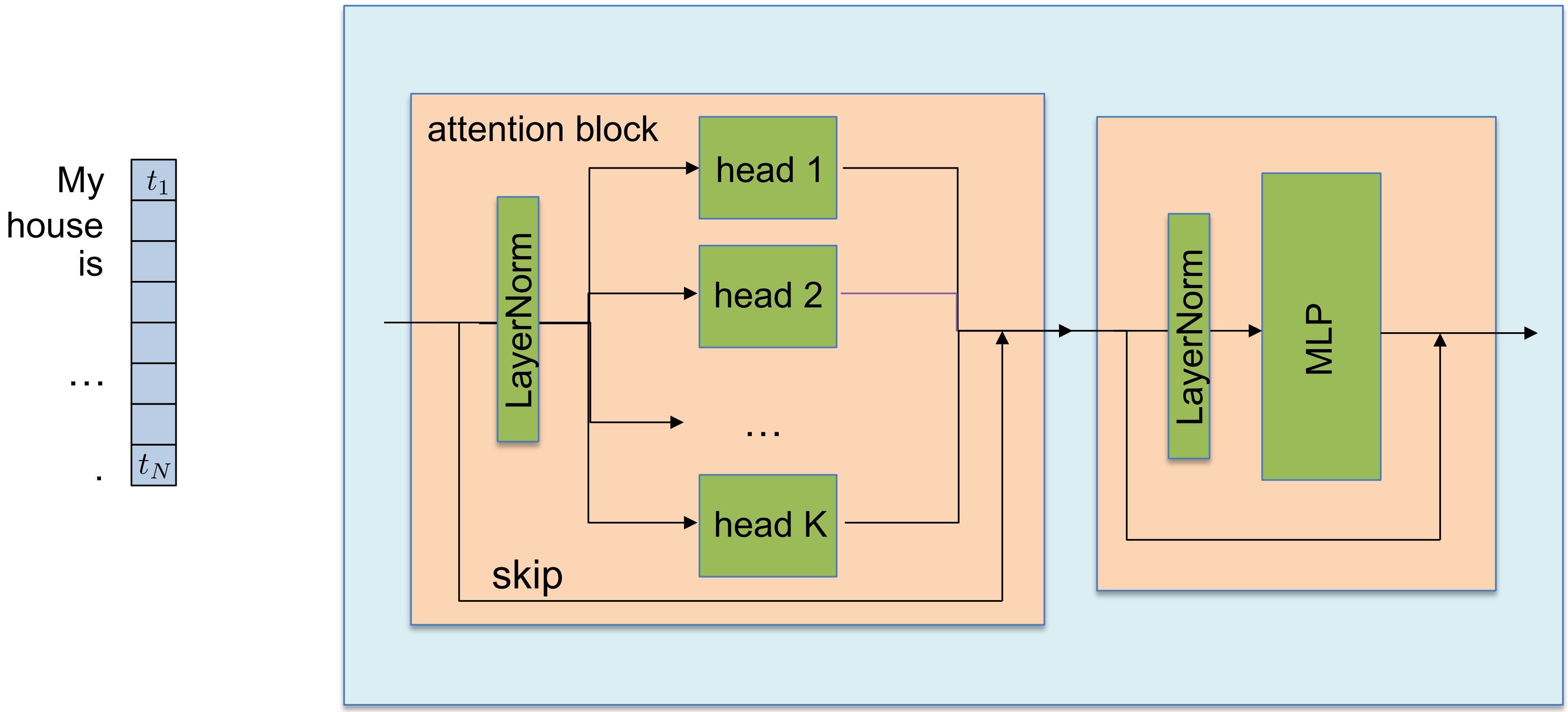


Transformer encoder

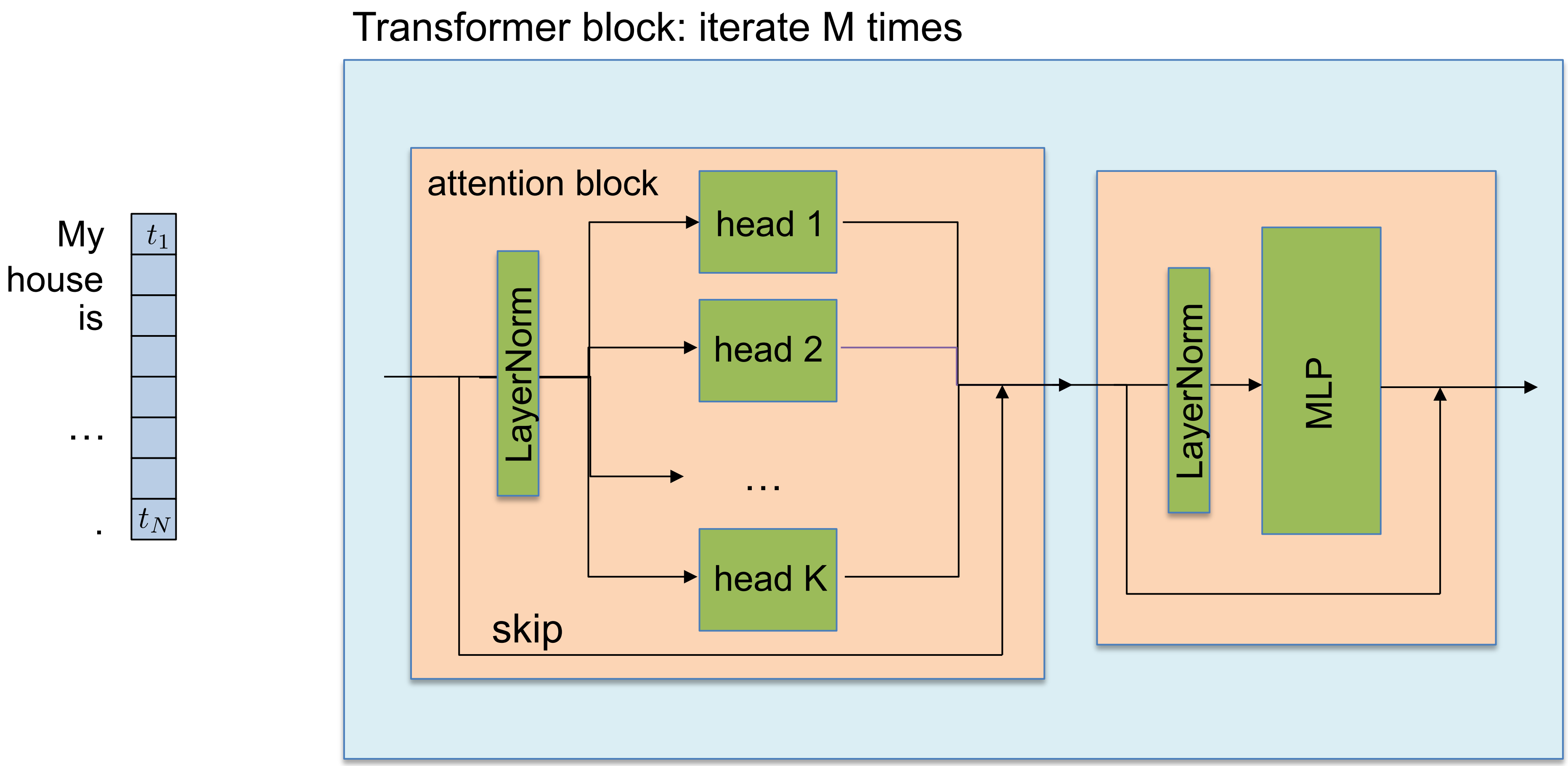


acts independently on each token

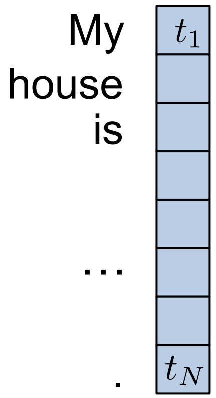
Transformer encoder

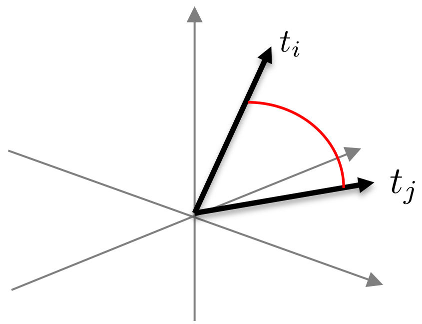


Transformer encoder



Transformer encoder

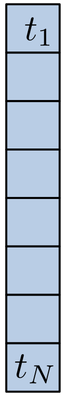


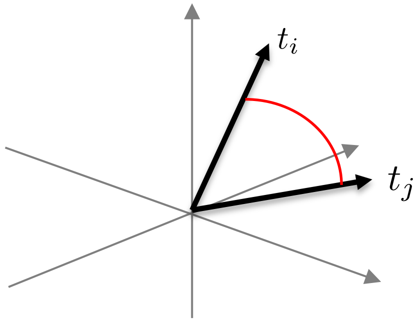


attention block

Transformer encoder

My
house
is
…
.

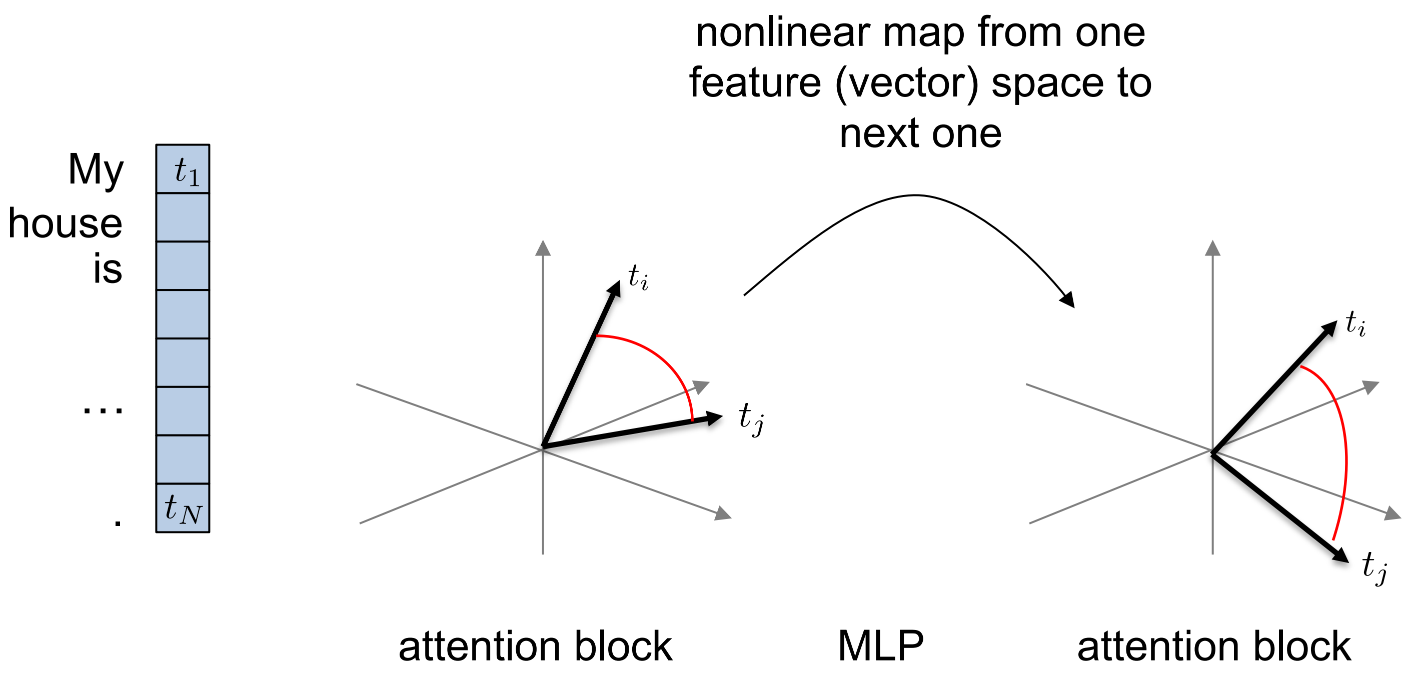




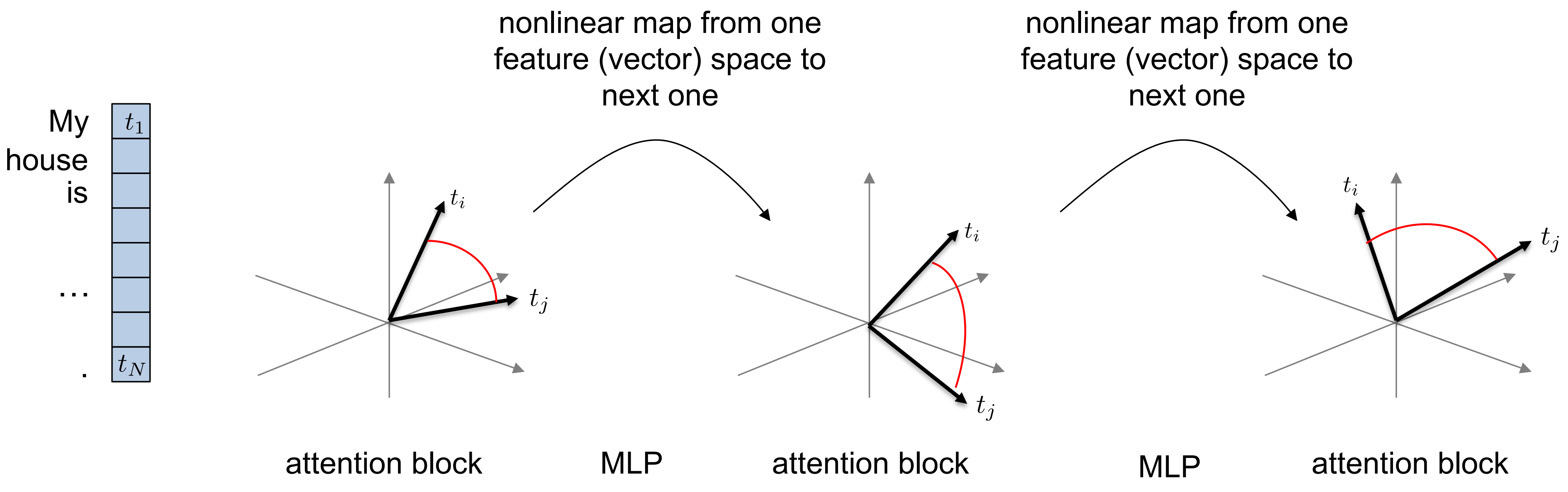
attention block

MLP

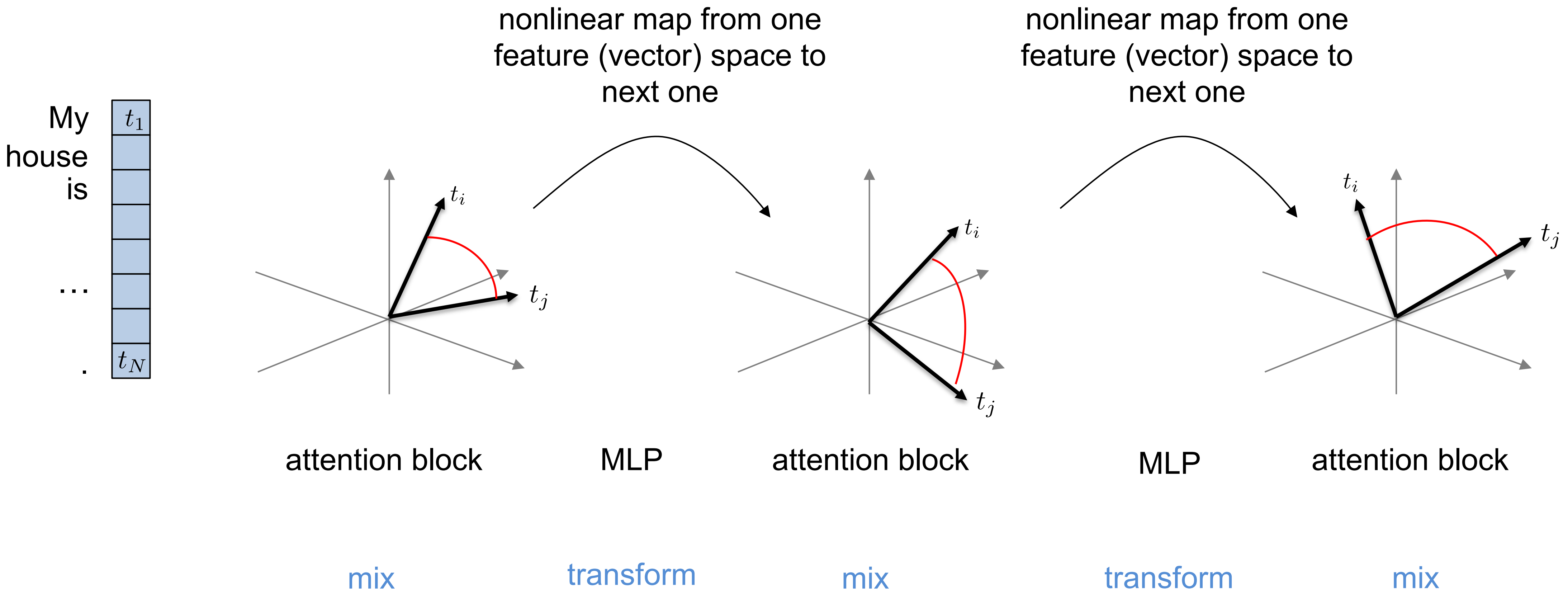
Transformer encoder



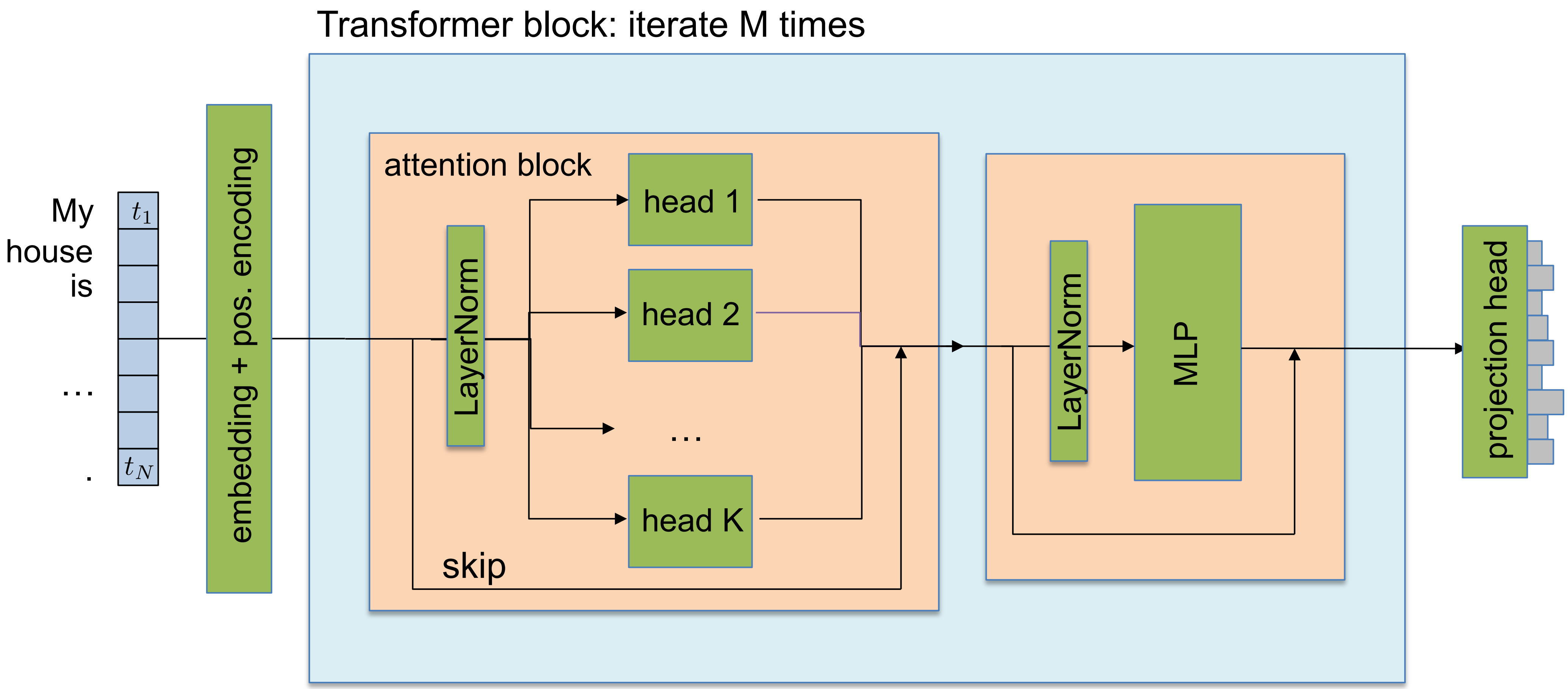
Transformer encoder



Transformer encoder

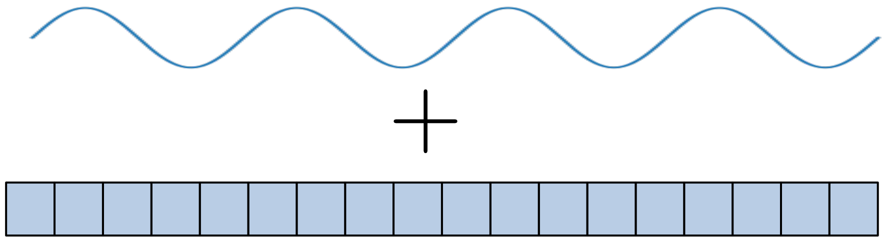


Transformer encoder



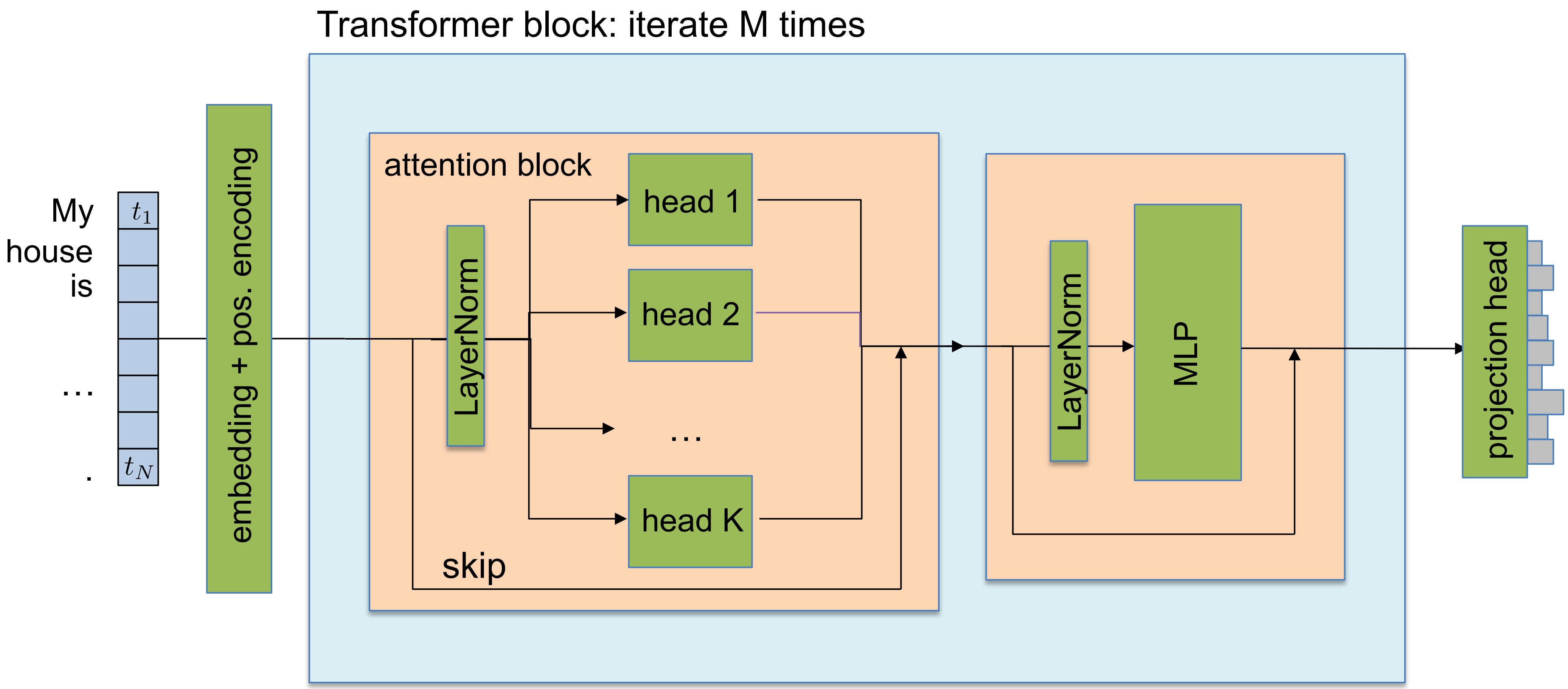
Transformer encoder

- Positional encoding
  - Tokens after embedding form a set without ordering
  - Structure/relationship between tokens needs to be encoded separately
  - Encoding can be fixed or learned
  - Classical approach: harmonic positional encoding that overlays sine/cosine oscillations with frequency that depends on position

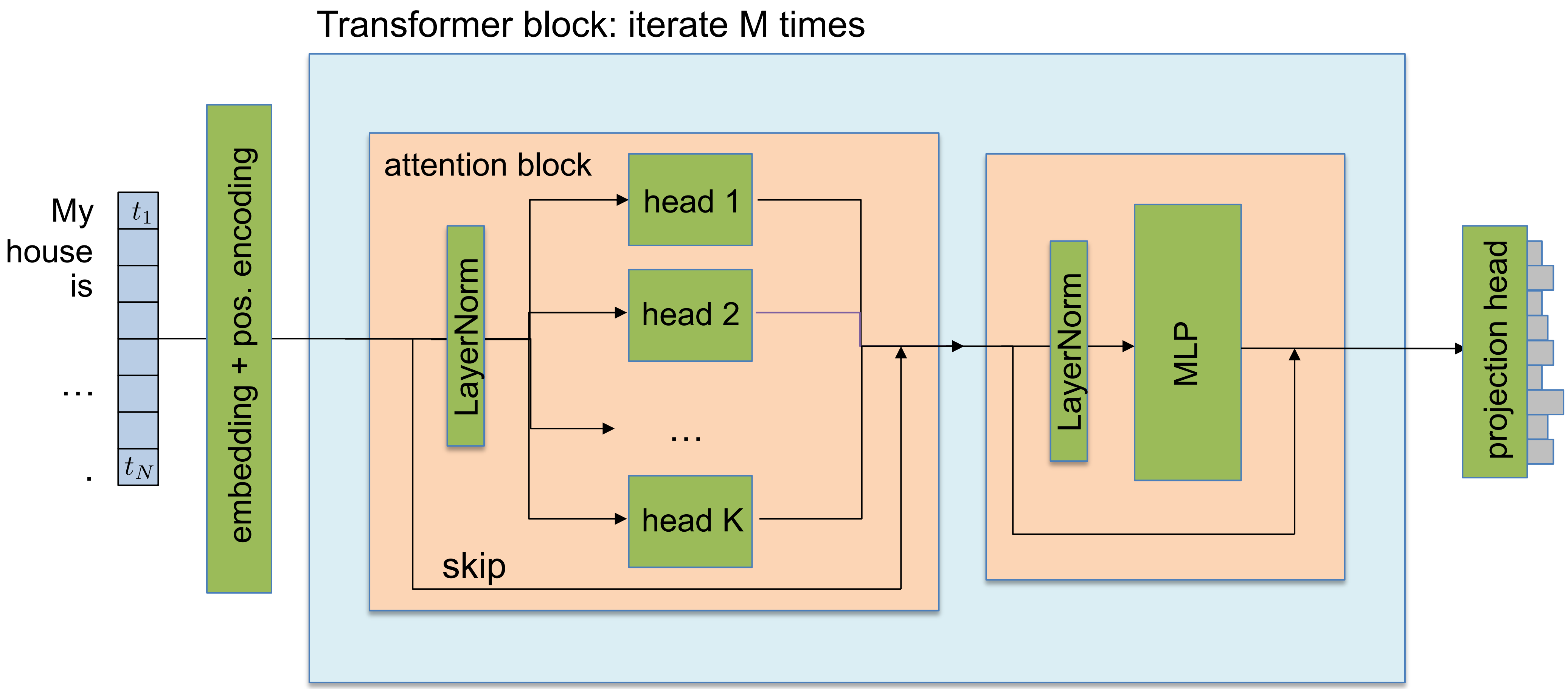


$t_i \in \mathbb{R}^E$

Transformer encoder



Vision transformer (encoder)



How to define a token when one has a pixel image instead of discrete words?

Vision transformer (encoder)

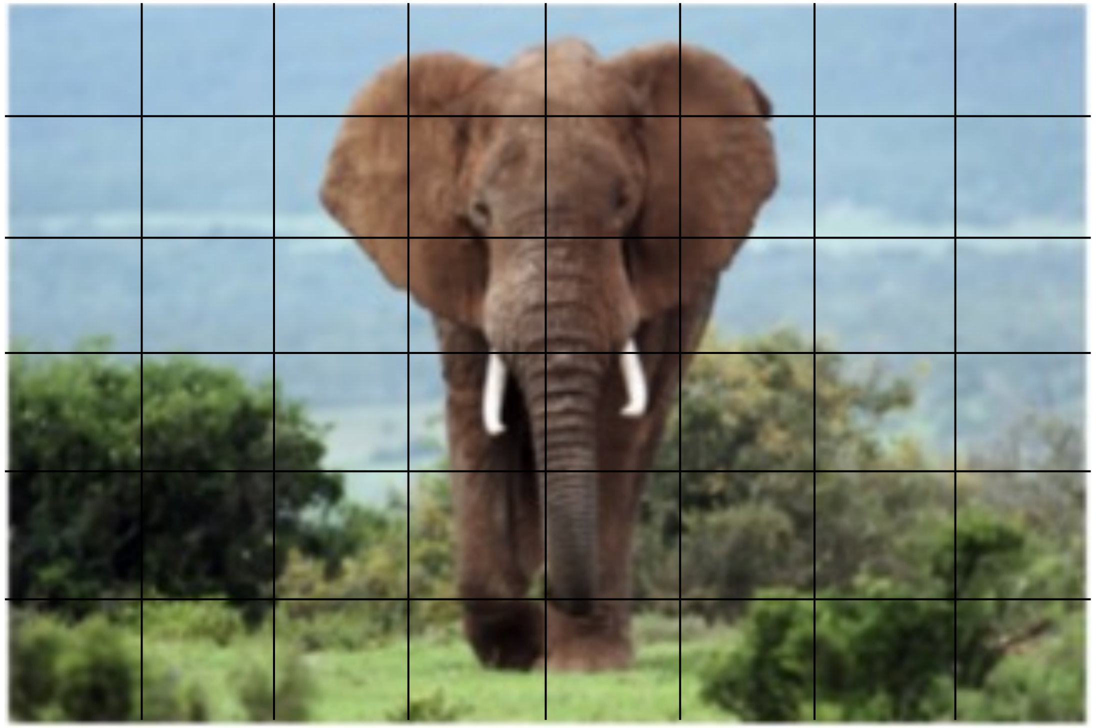
.

How to define a token when one has a pixel image instead of discrete words?

Dosovitskiy et al., An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, 2021, https://arxiv.org/abs/2010.11929

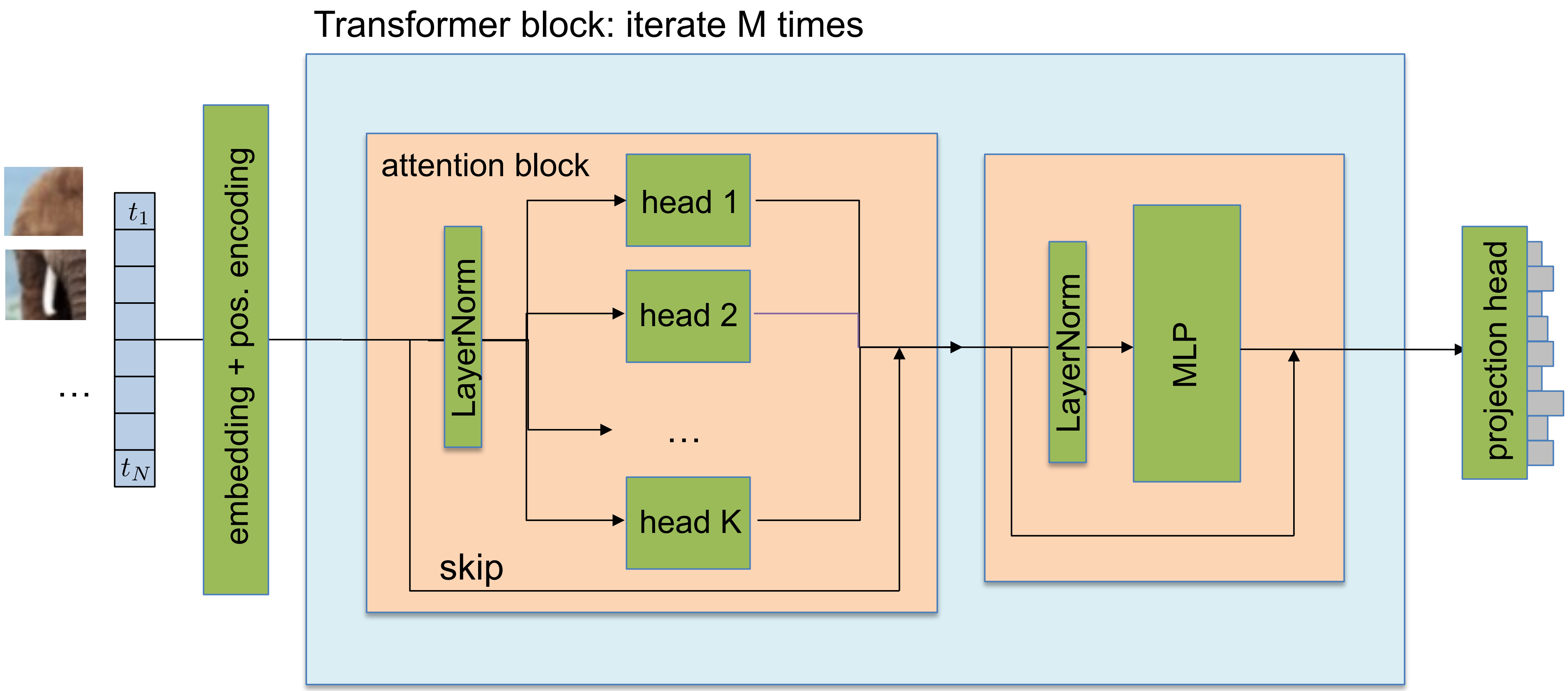
Vision transformer (encoder)

token is small image patch
- enough structure for dot product to make sense
- small enough attention to provide rich structure



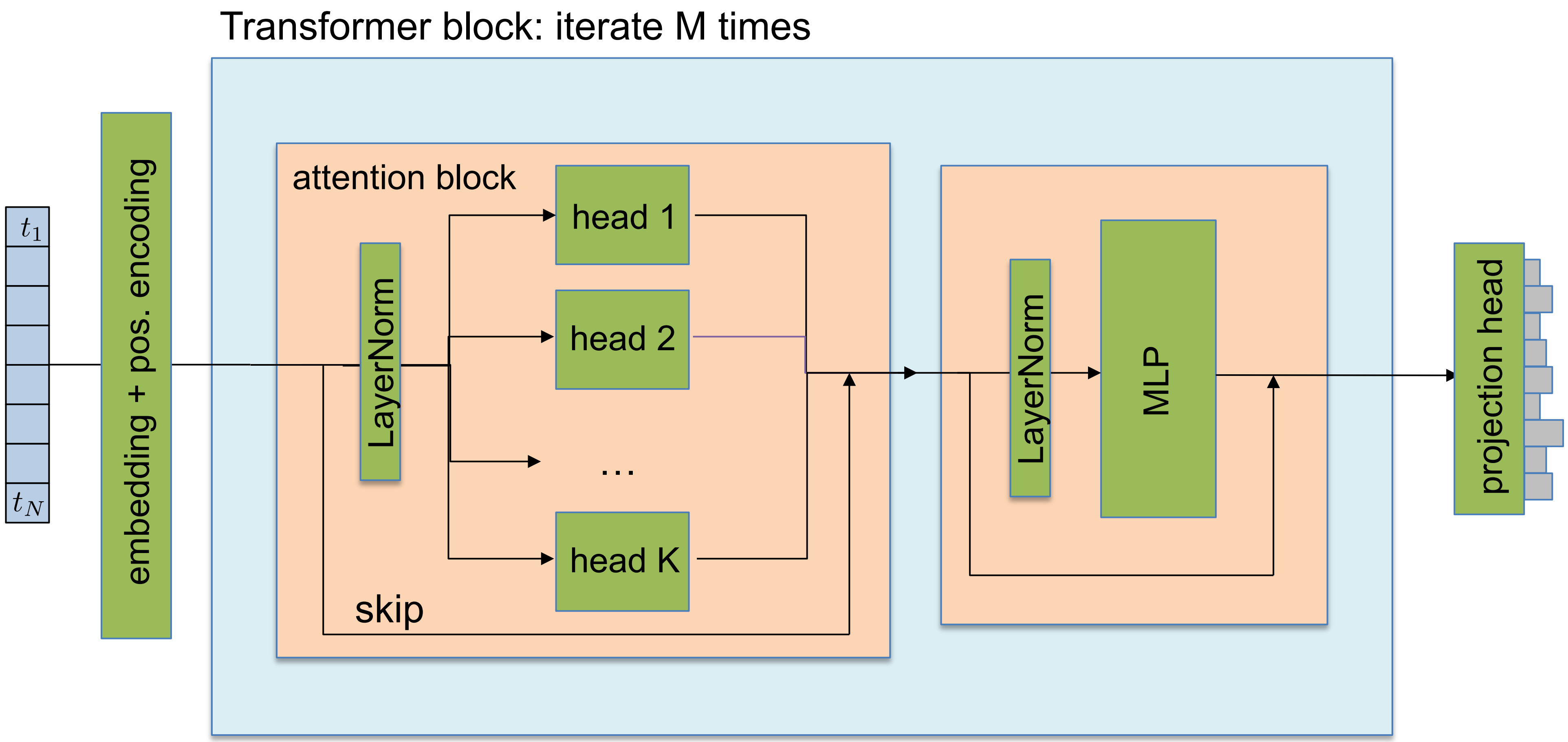
How to define a token when one has a pixel image instead of discrete words?

Vision transformer (encoder)



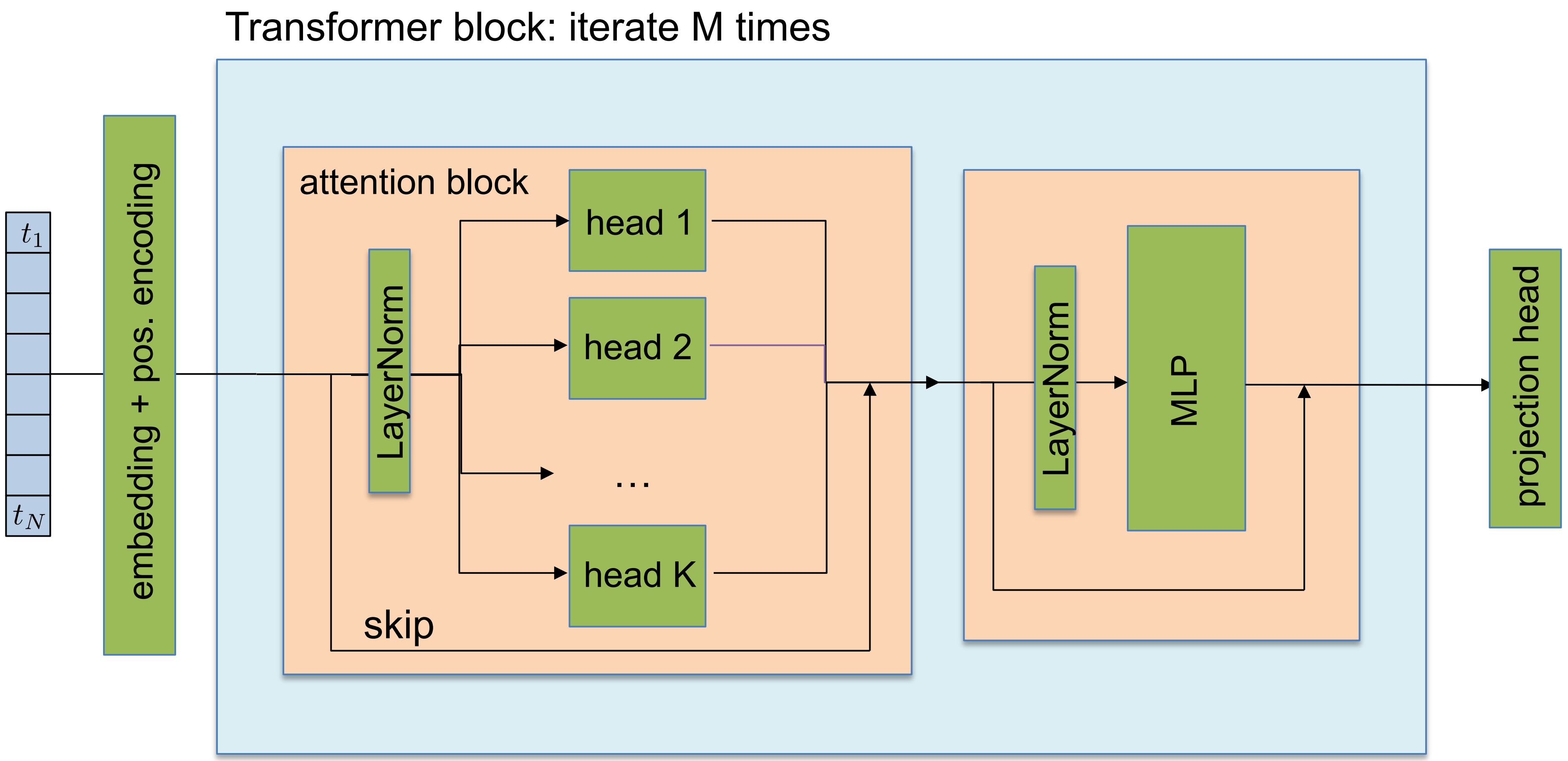
Vision transformer: token is small but non-trivial image patch

X transformer (encoder)



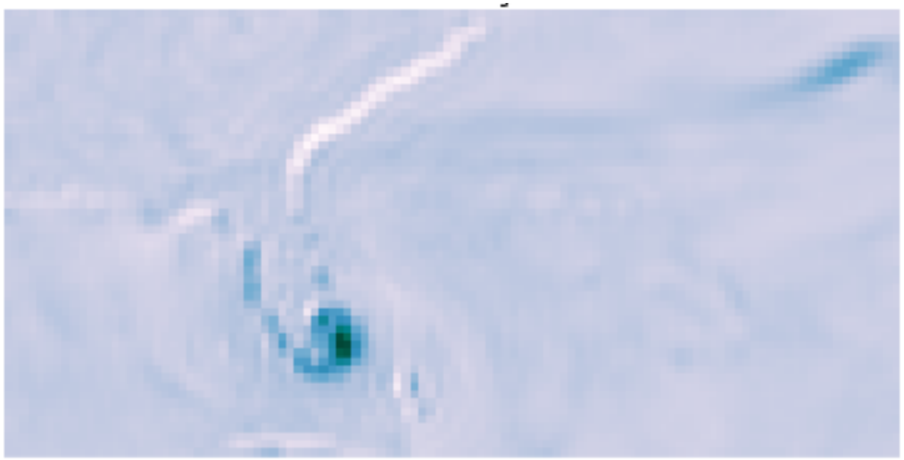
Central (only) question: what is a token, i.e. what is small information "nugget"?

X transformer (encoder)



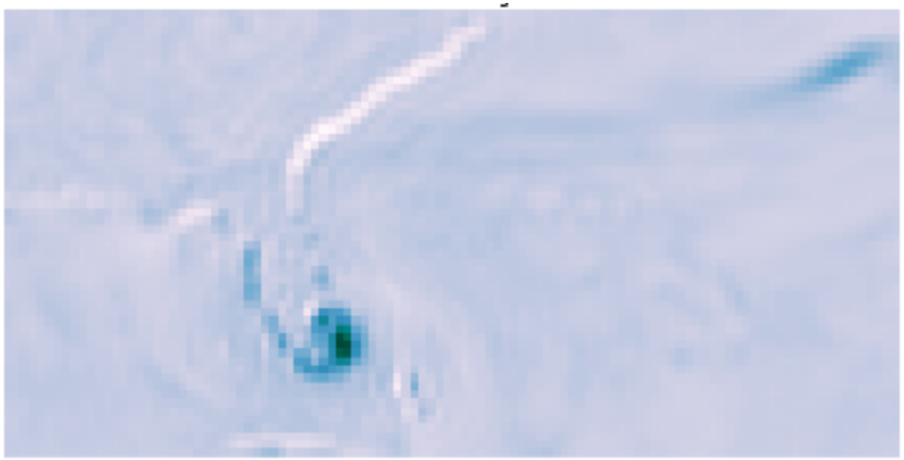
Atmosphere: token as information from small space-time neighborhood

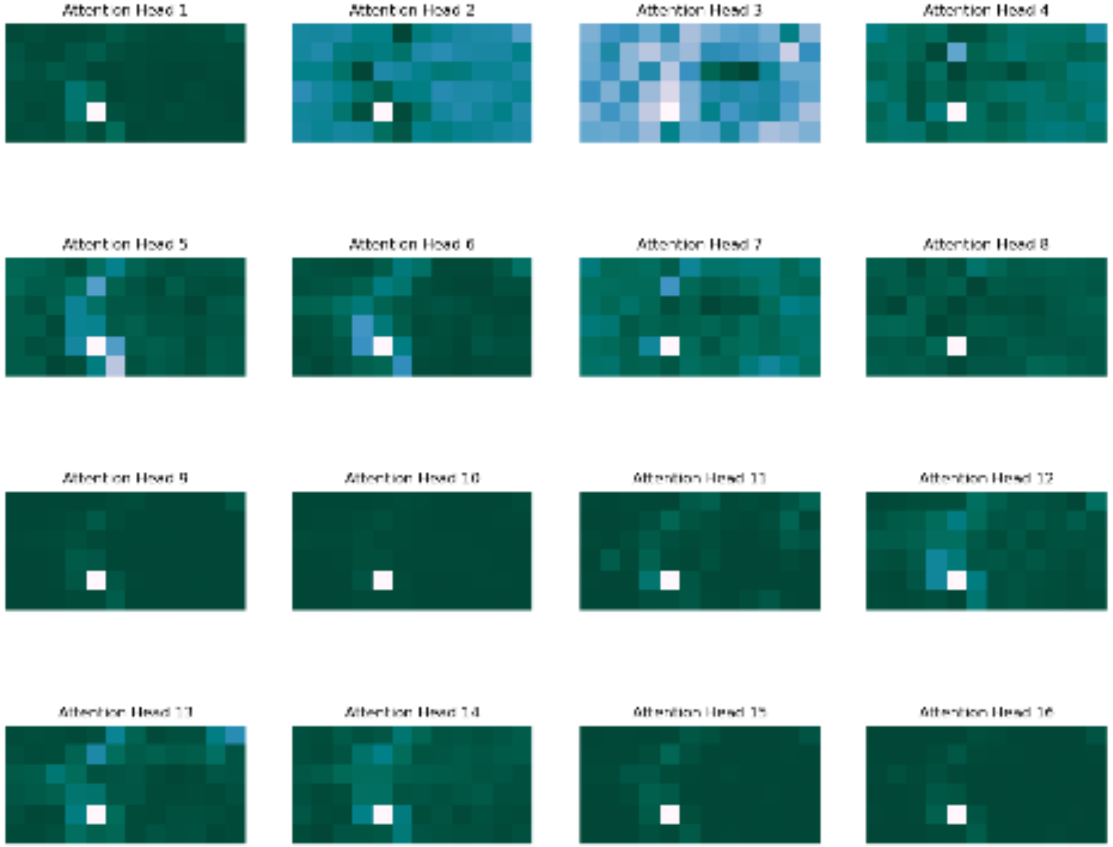
What is attention?



Lessig et al., AtmoRep, 2023, https://arxiv.org/abs/2308.13280

What is attention?



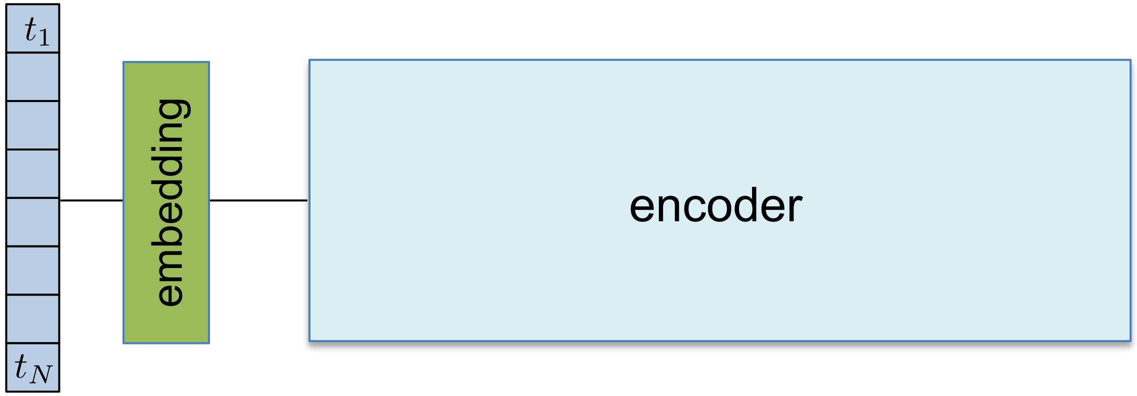


Lessig et al., AtmoRep, 2023, https://arxiv.org/abs/2308.13280

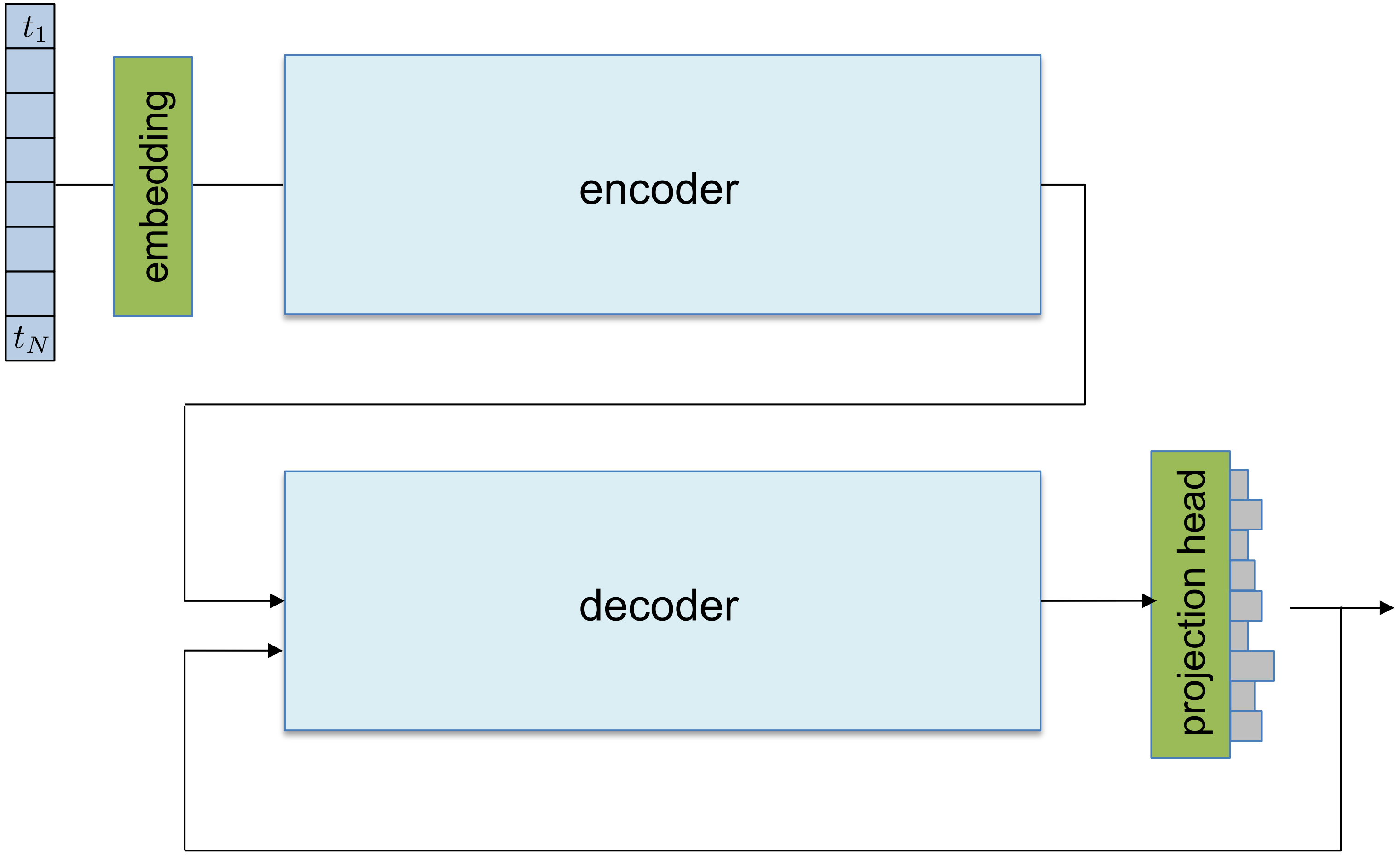
Transformer encoder-decoder

- Introduced in the original transformer paper (Vaswani et al., 2017) for translation tasks
  - Encode: input and encode language A
  - Decoder: decode and output language B

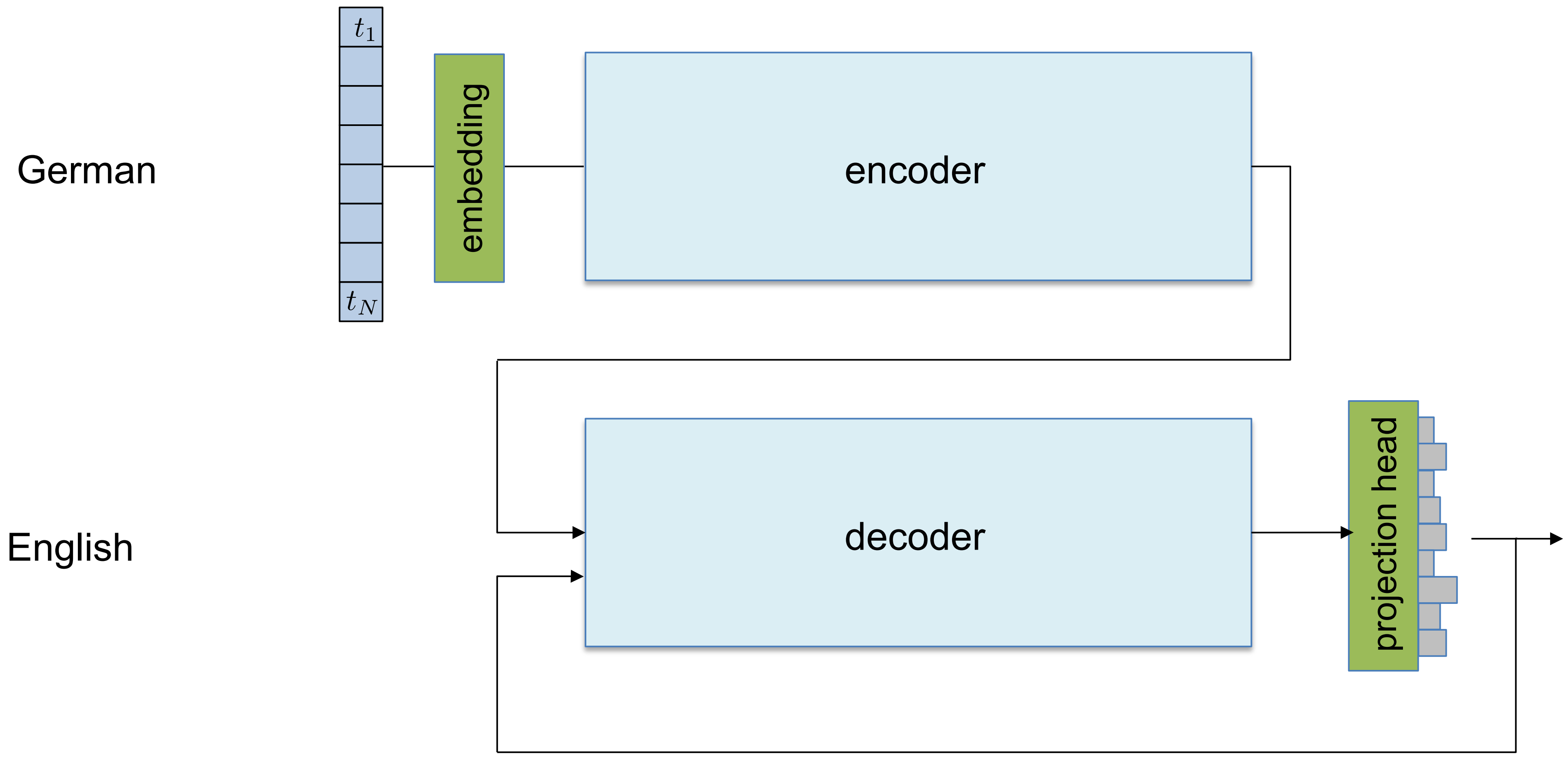
Transformer encoder-decoder



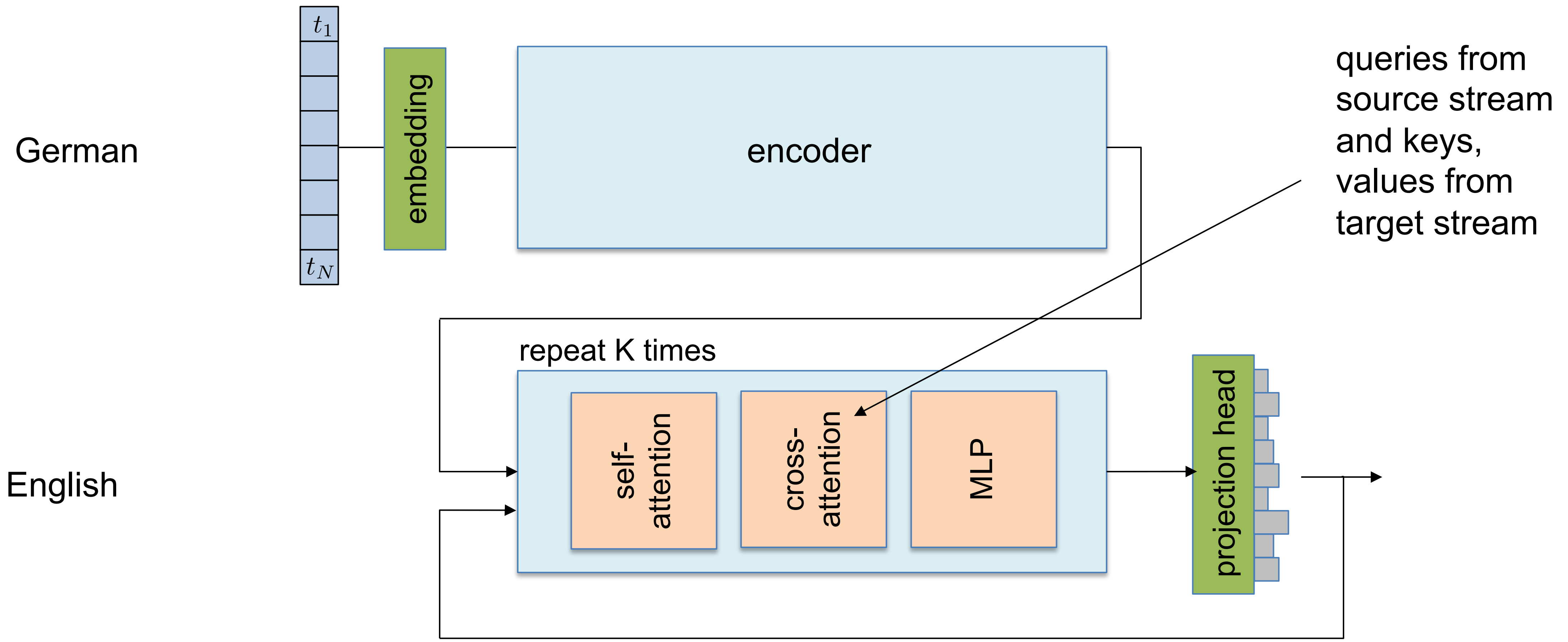
Transformer encoder-decoder



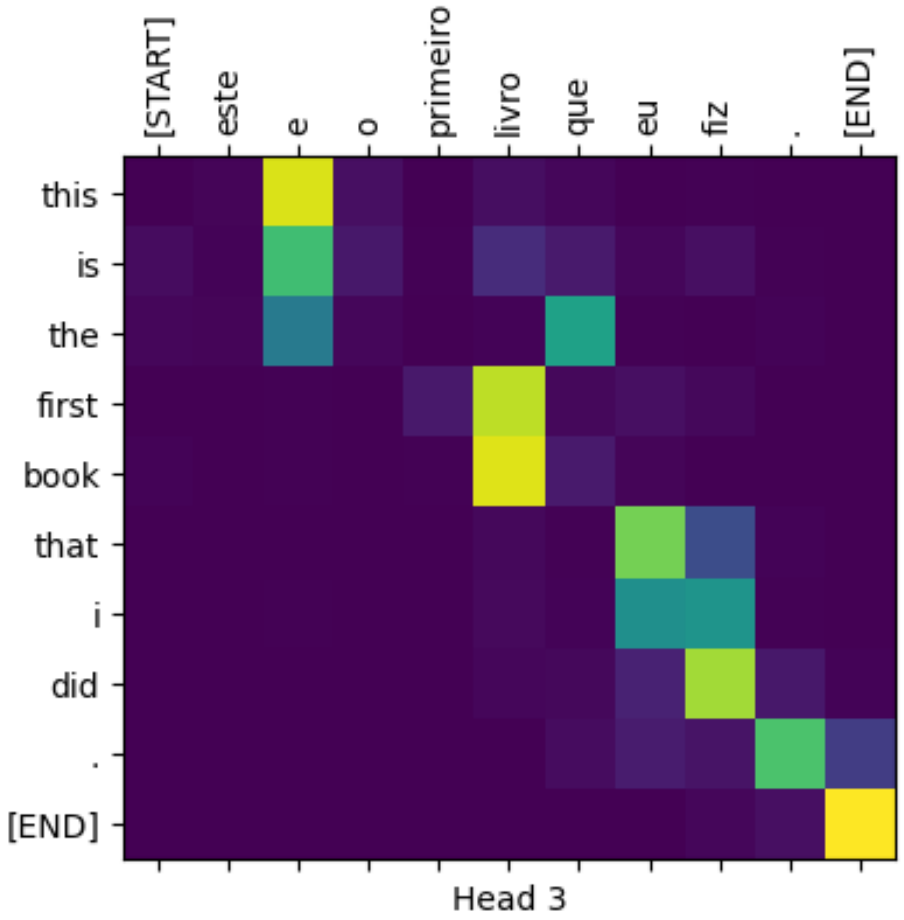
Transformer encoder-decoder



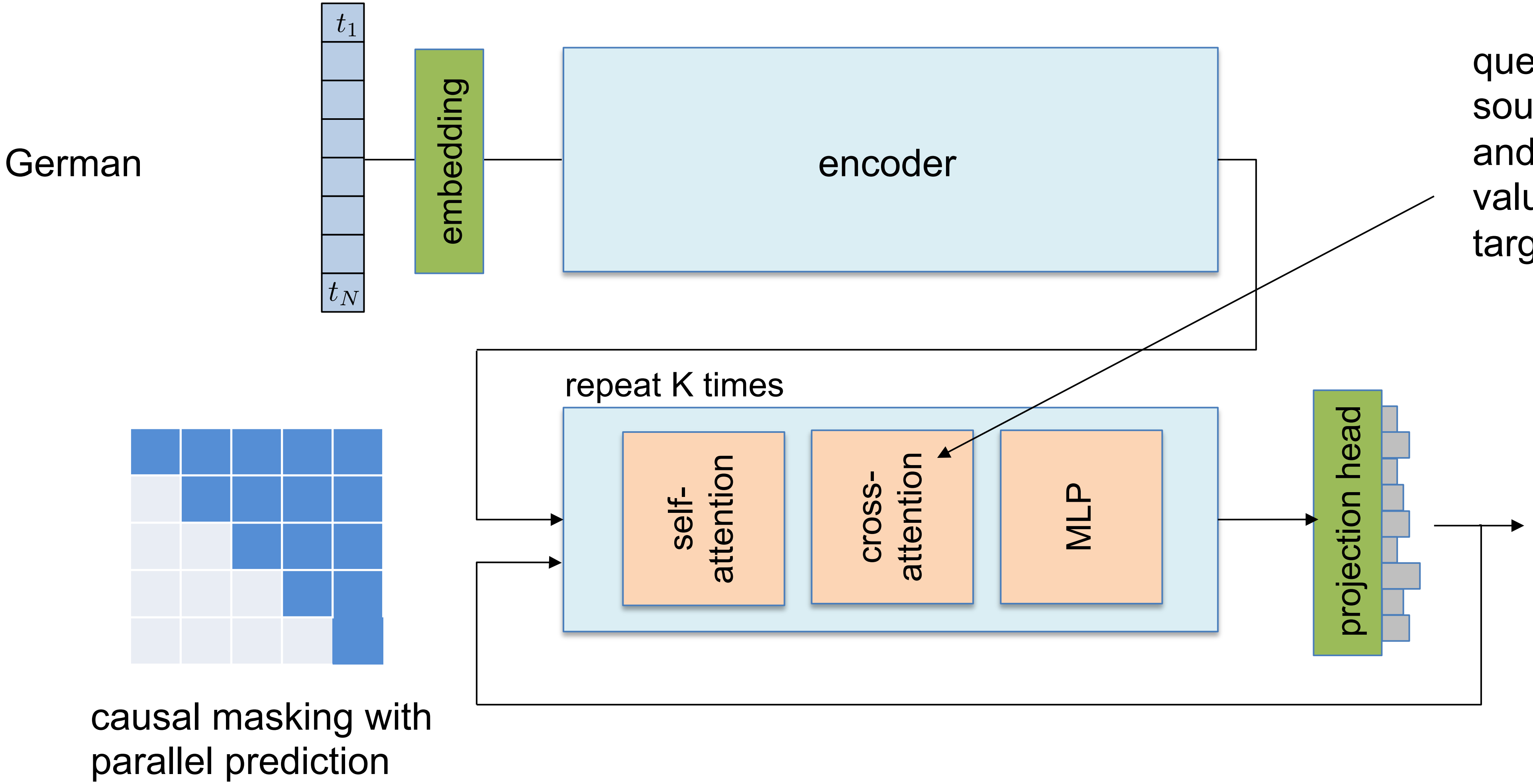
Transformer encoder-decoder



Transformer encoder-decoder



Transformer encoder-decoder



Extensions

- Sparse attention
  - Attention is quadratic which quickly limits the number of tokens one can process
  - Sparse attention computes attention only between “likely” relevant tokens, e.g. nearby ones in a temporal stream
- Shared kv’s between heads
- qk LayerNorm
  - Additional layer norm in attention head (important for large models)

Summary

- Transformer are standard neural network in many applications
- Sequence-to-sequence model operating on a set/sequence of tokens
- Versatile since only the definition of a token and the embedding network is domain specific
- Computationally highly efficient since software and hardware is optimized for them
    - But sparse attention required in general