The Transformer architecture

Marco Kuhlmann
Department of Computer and Information Science

LINKÖPING UNIVERSITY

Attention is all you need

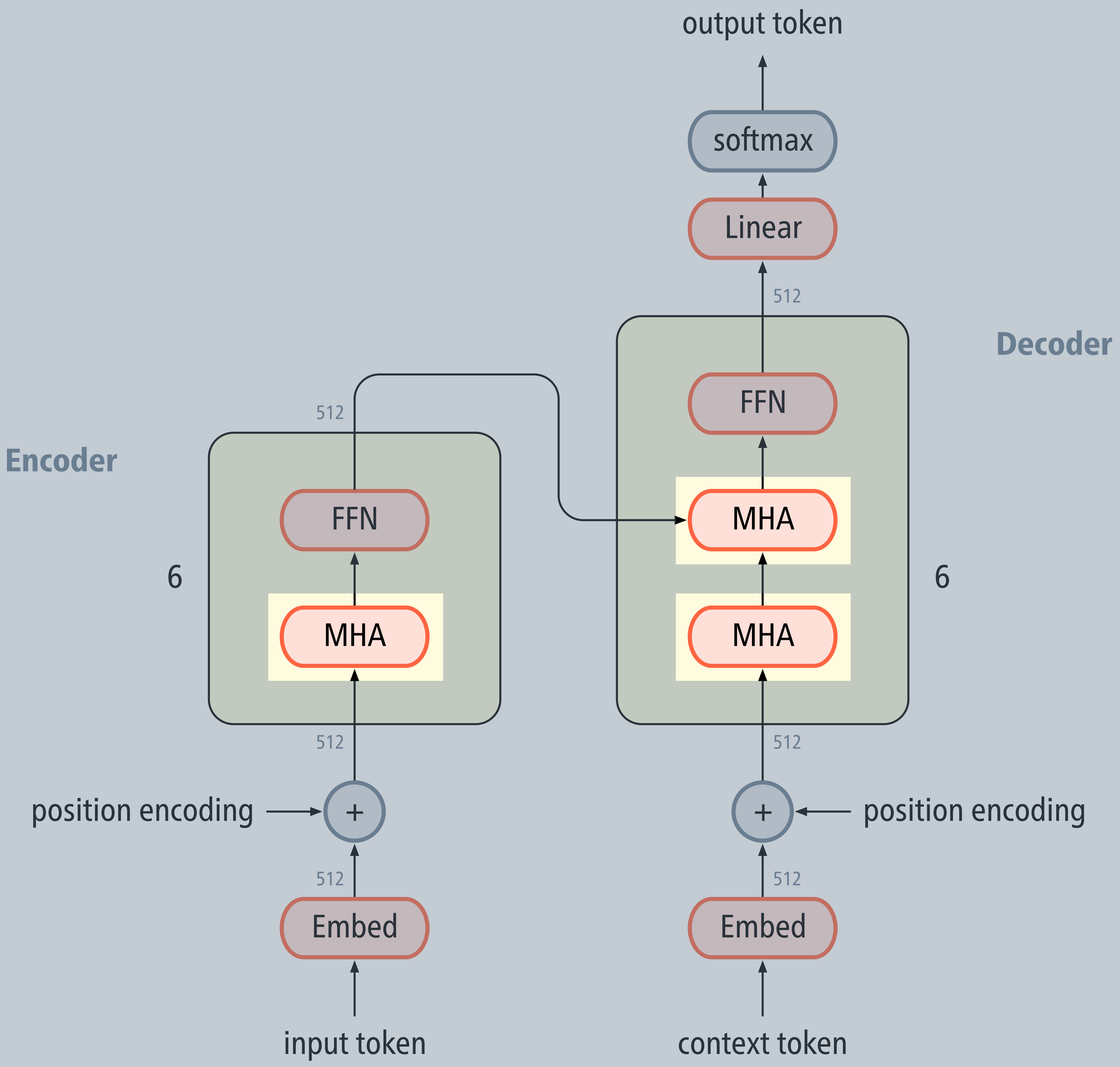
- Recurrent neural networks implement a sequential model of computation in that it processes sequence elements one by one.

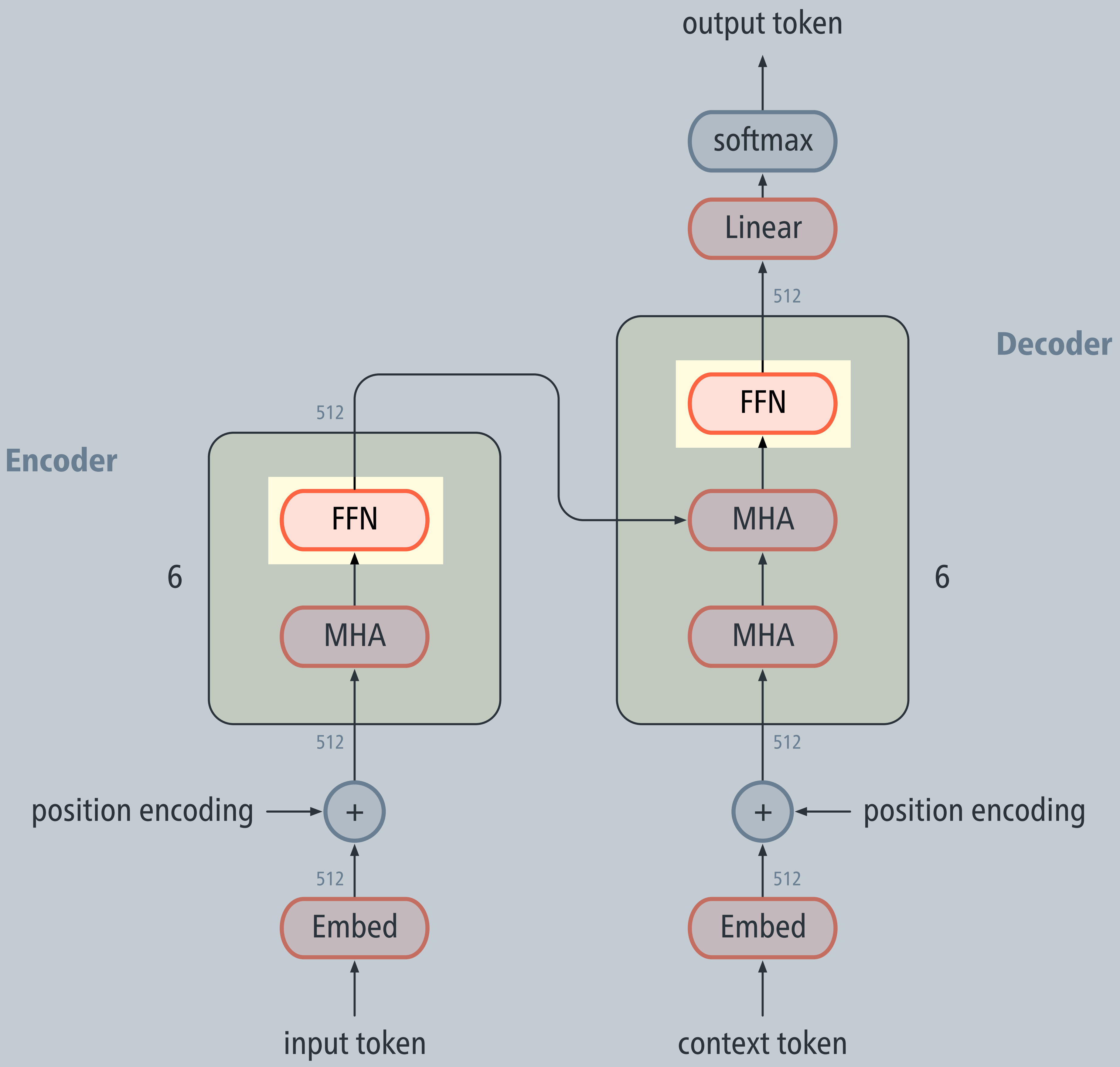
- In contrast, attention facilitates direct access to all elements, independently of sequence length.

- The **Transformer** is an encoder–decoder architecture that drops recurrent neural networks and exclusively uses attention.

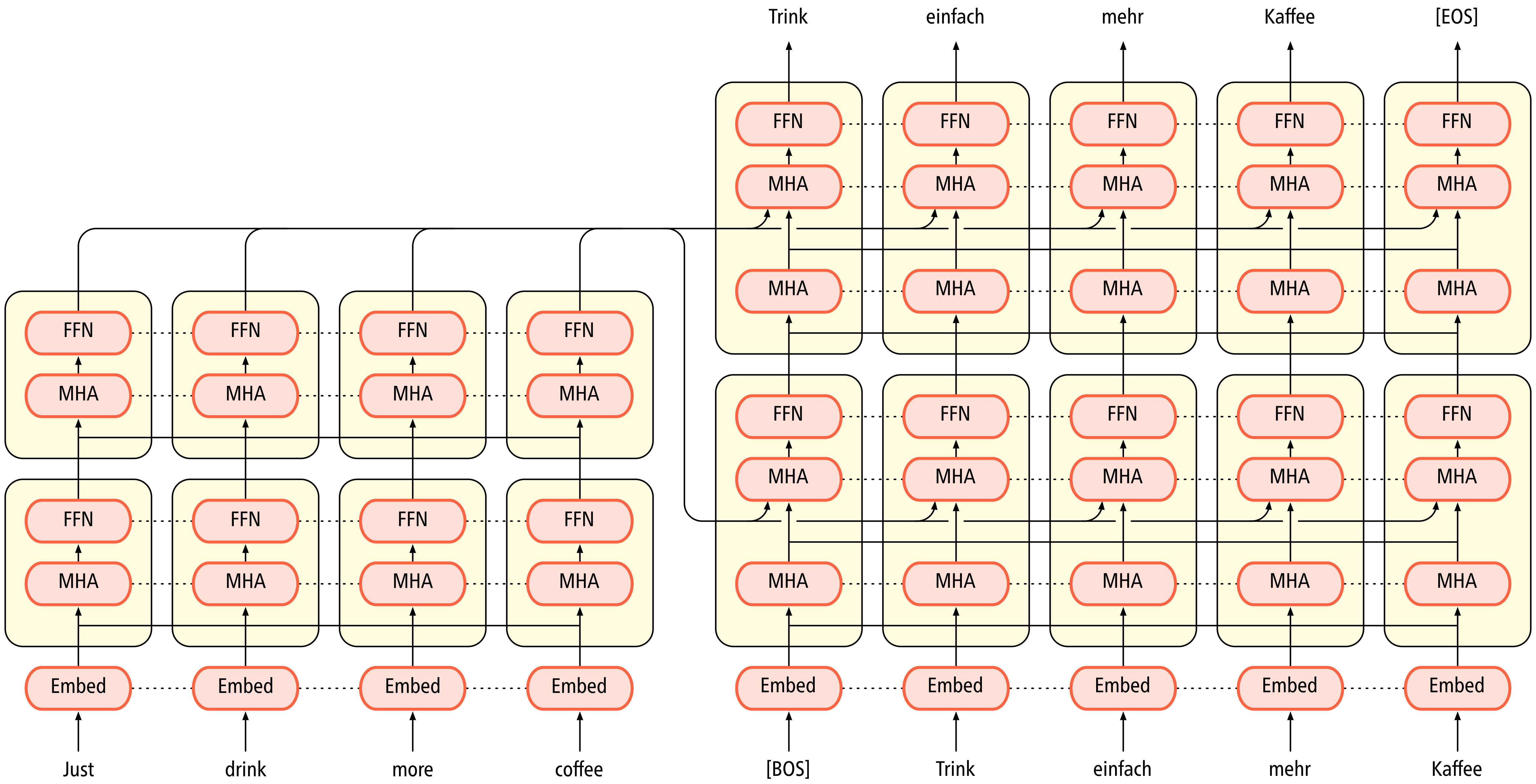
  can be parallelised

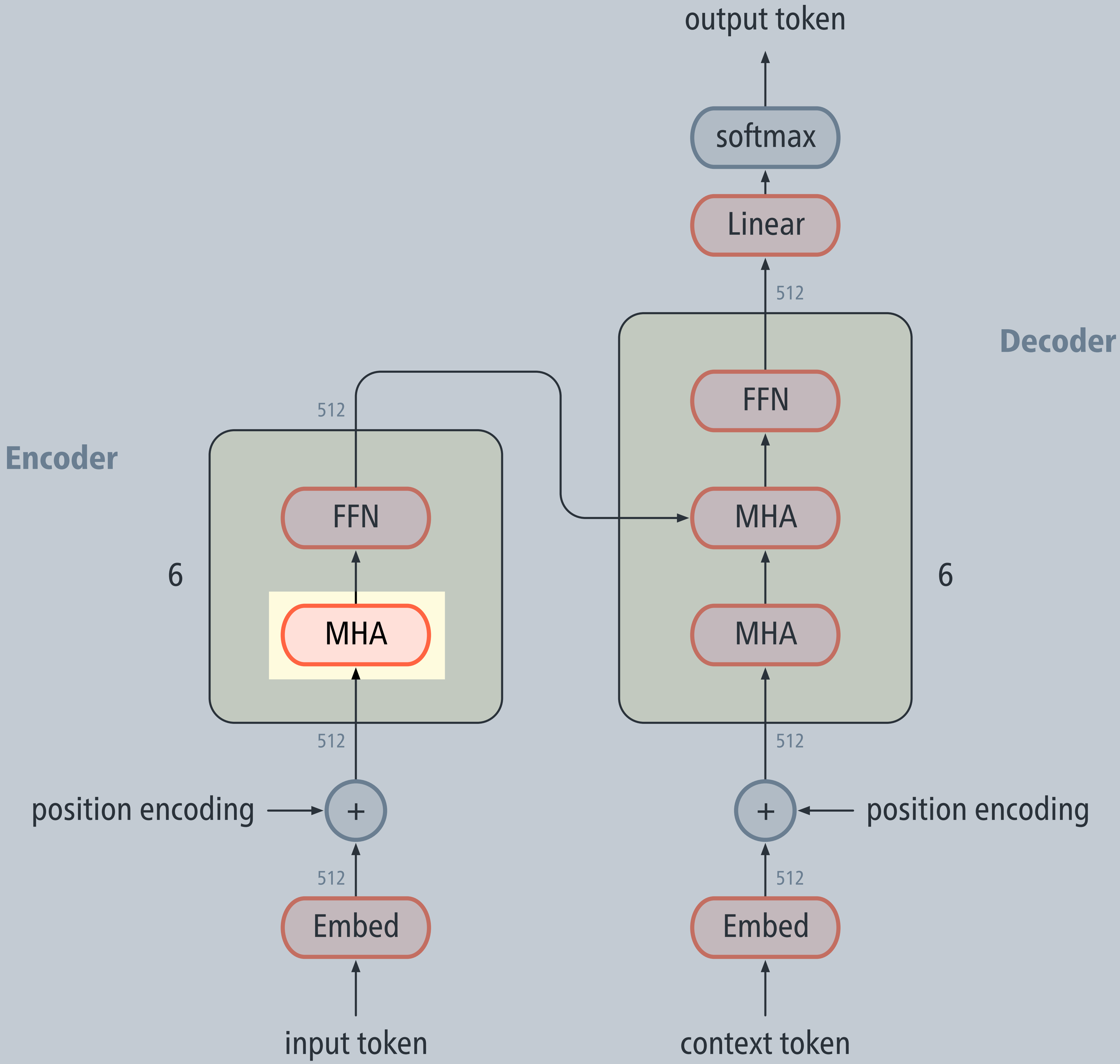
Vaswani et al. (2017)


output token
softmax
Linear
512
Decoder
FFN
MHA
MHA
6
512
Encoder
FFN
MHA
6
512
position encoding
+
512
Embed
input token
512
+
position encoding
512
Embed
context token

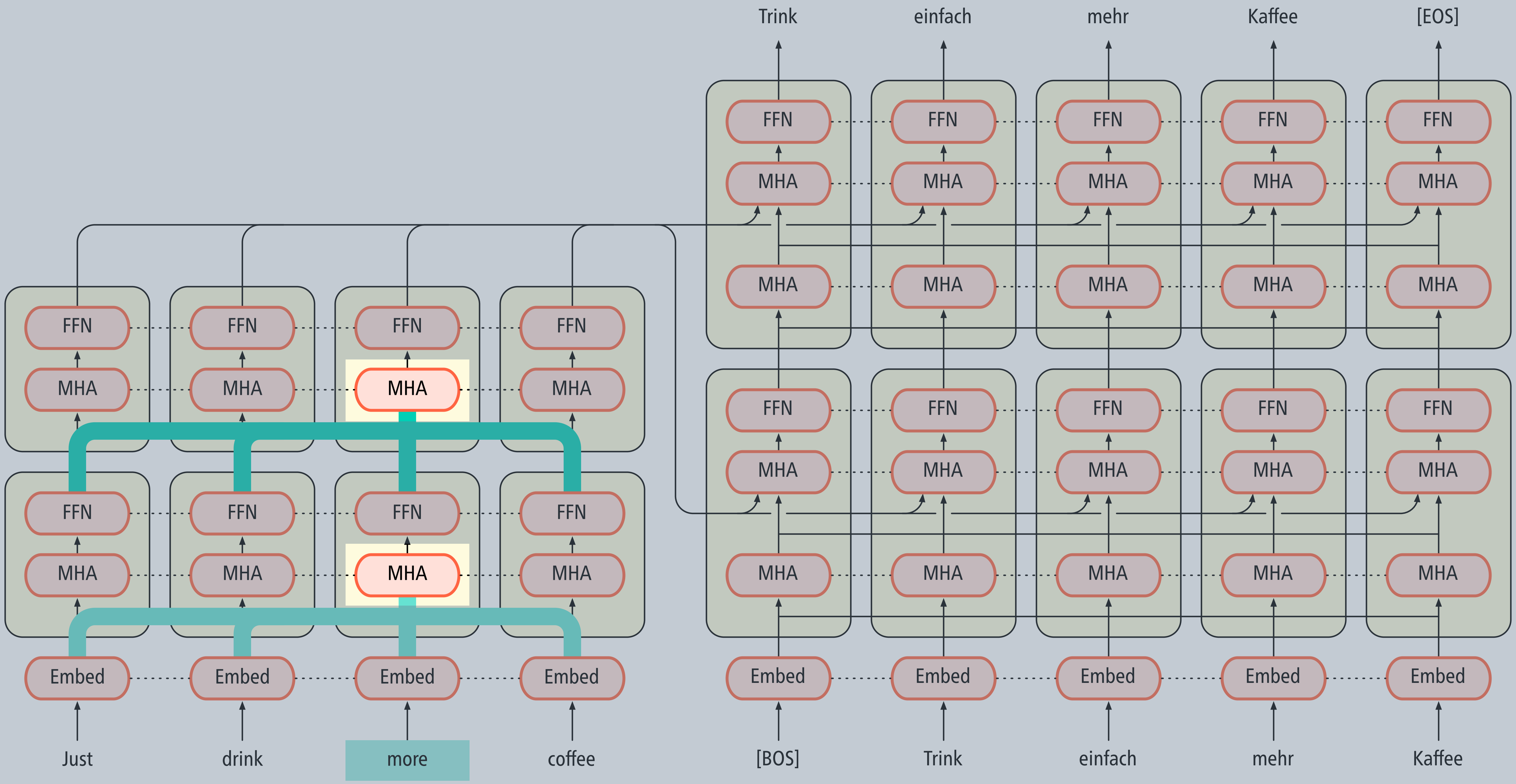

output token
softmax
Linear
512
Decoder
FFN
512
Encoder
FFN
MHA
MHA
6
MHA
6
512
512
position encoding
+
+
position encoding
512
512
Embed
Embed
input token
context token

Example translation

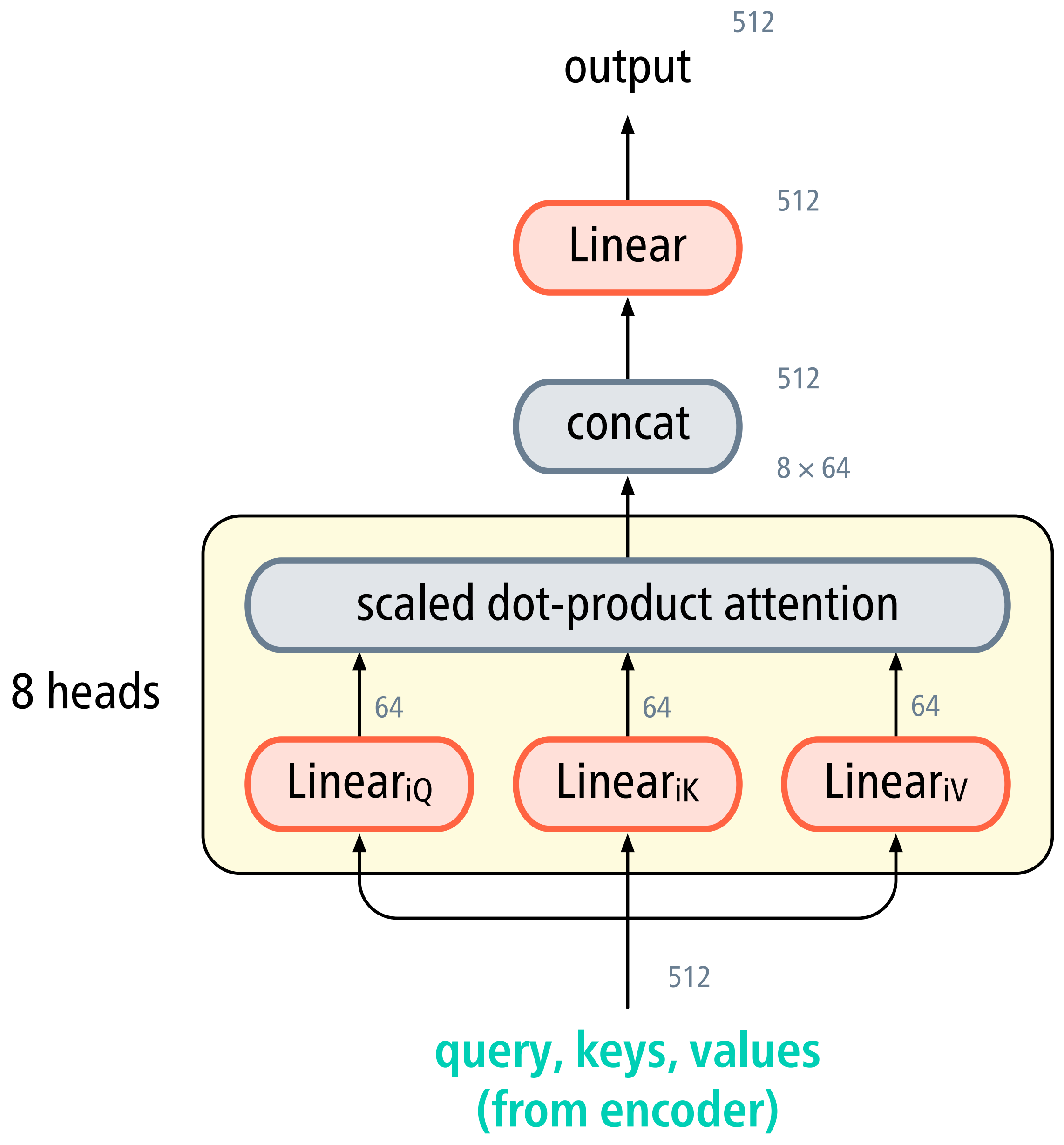


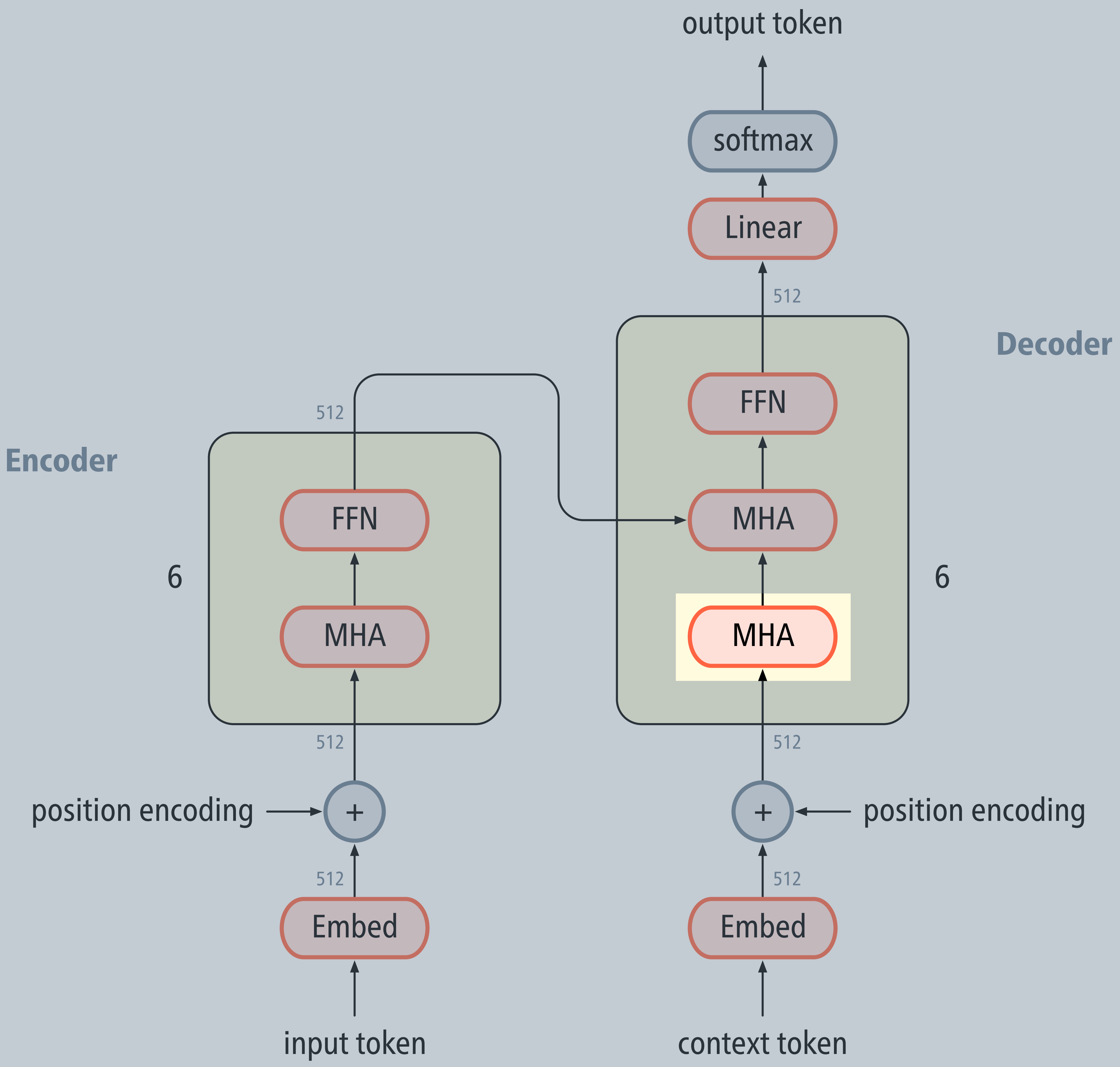

output token
softmax
Linear
512
Decoder
FFN
MHA
MHA
6
512
Encoder
FFN
MHA
6
512
position encoding
+
512
Embed
input token
512
+
position encoding
512
Embed
context token

Multi-head attention in the encoder

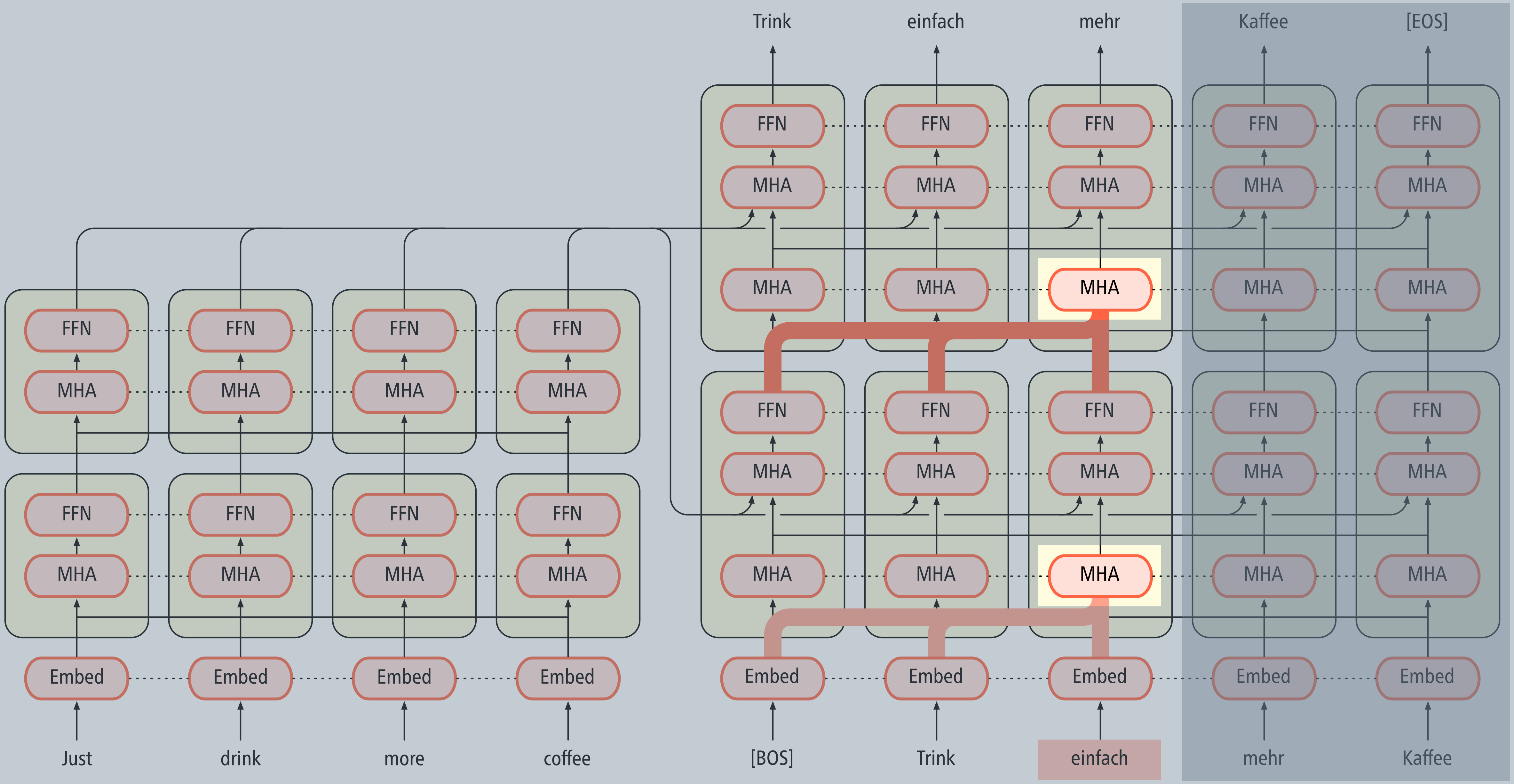


Multi-head attention in the encoder

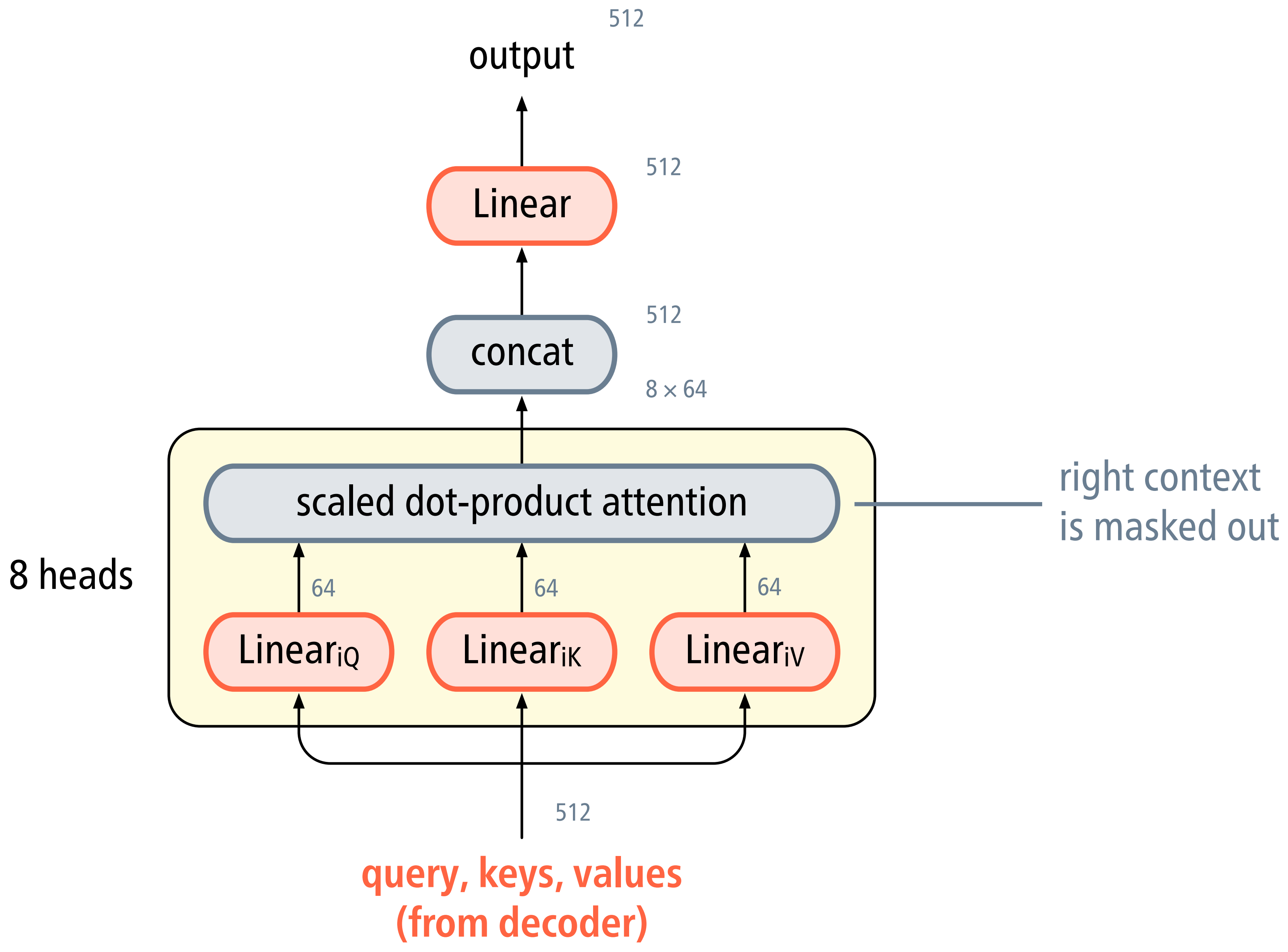



output token
softmax
Linear
512
Decoder
512
FFN
Encoder
FFN
MHA
6
MHA
6
MHA
512
512
position encoding
+
+
position encoding
512
512
Embed
Embed
input token
context token

Multi-head attention in the decoder



Multi-head attention in the decoder



512
output

512
Linear

512
concat
8 × 64

scaled dot-product attention

right context
is masked out

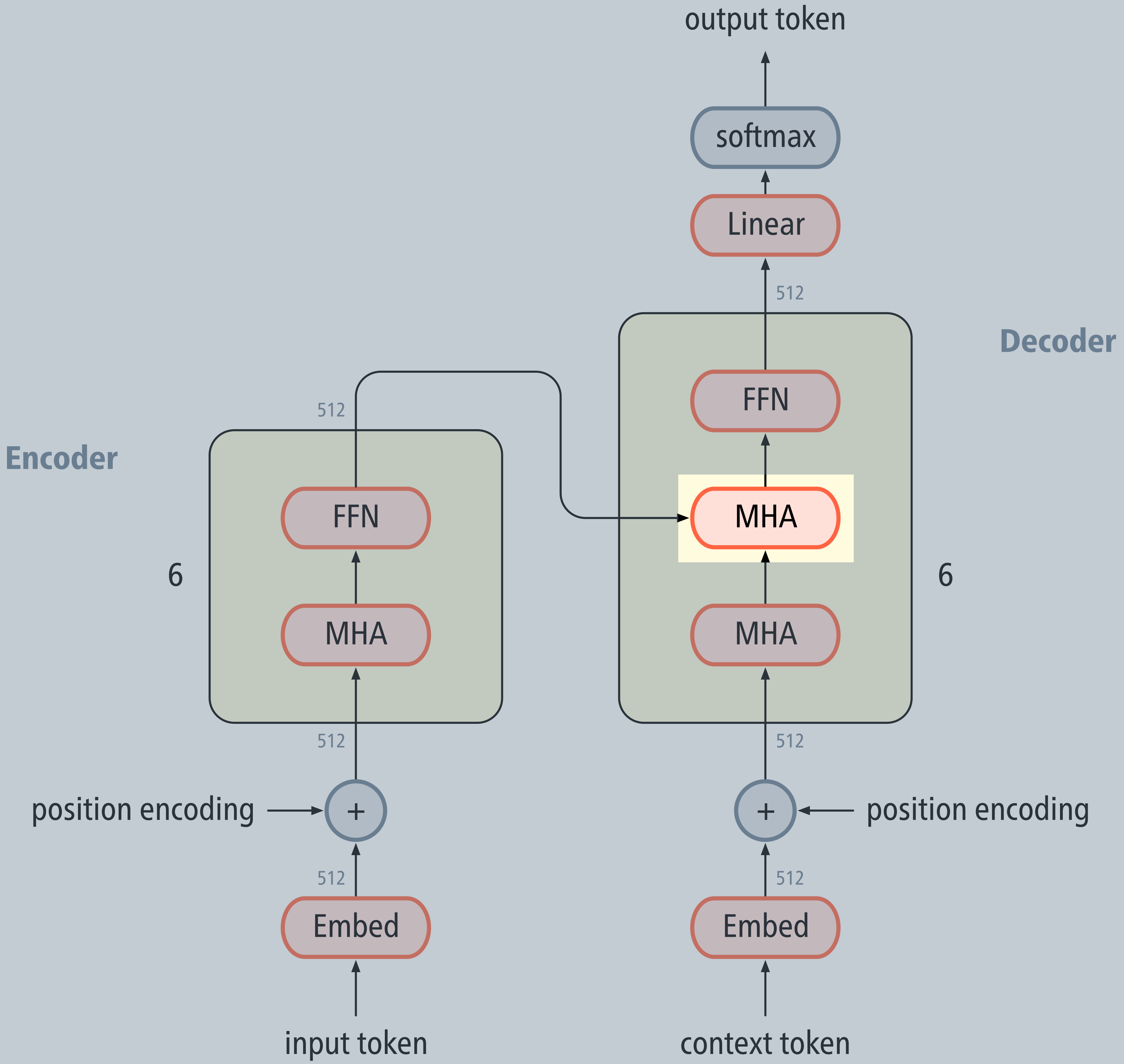
8 heads

64
64
64

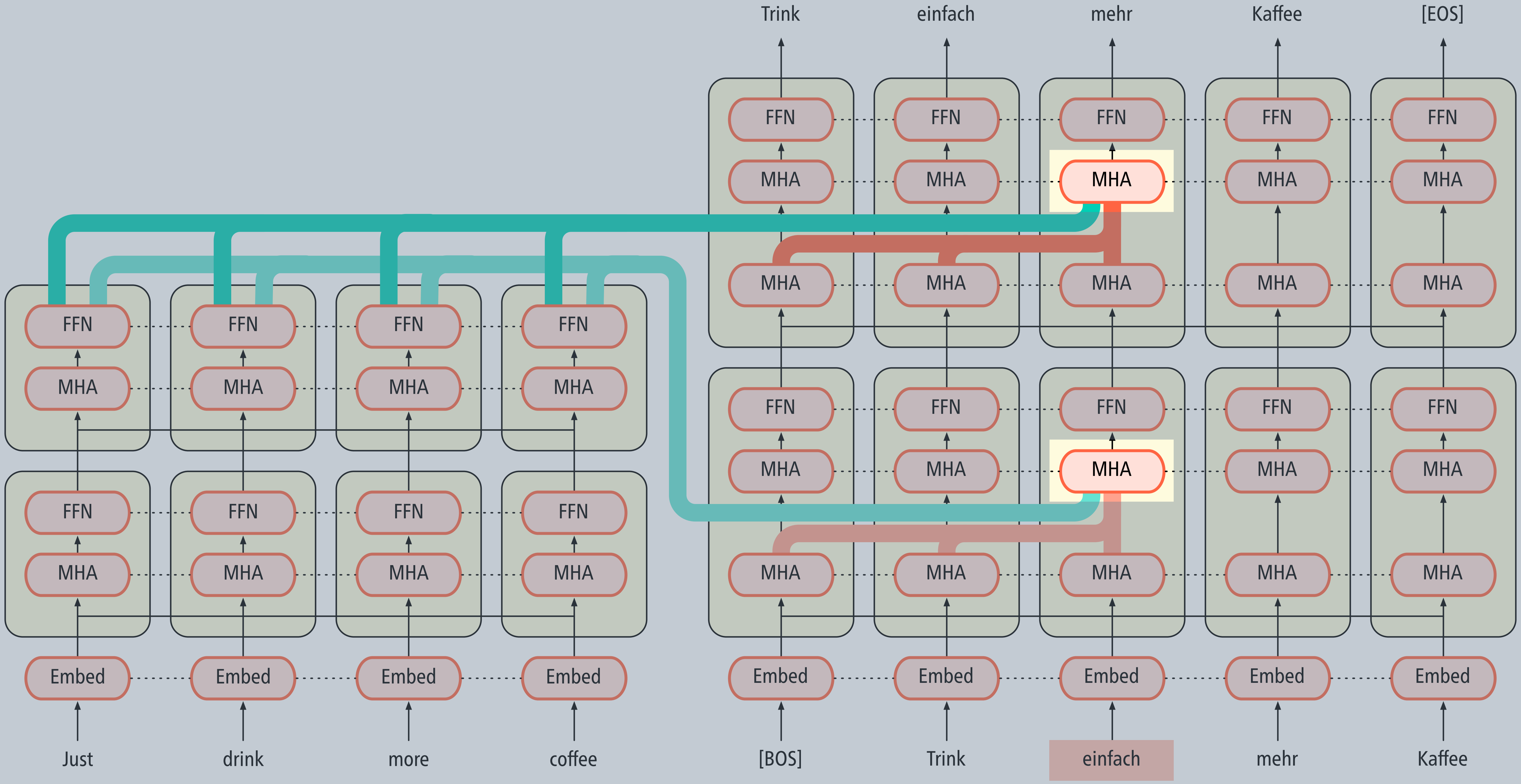
Linear$_{iQ}$
Linear$_{iK}$
Linear$_{iV}$

512

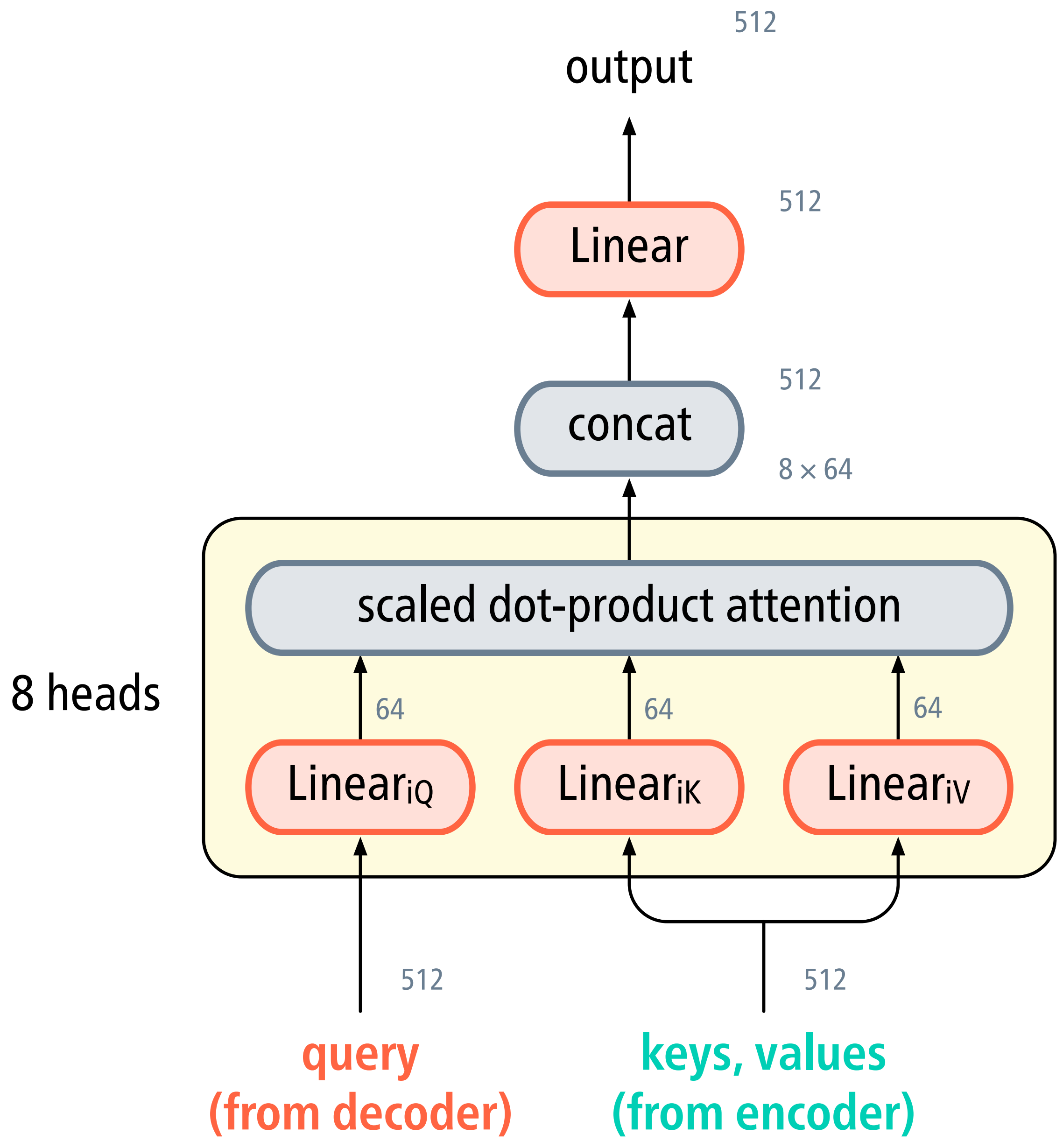
query, keys, values
(from decoder)

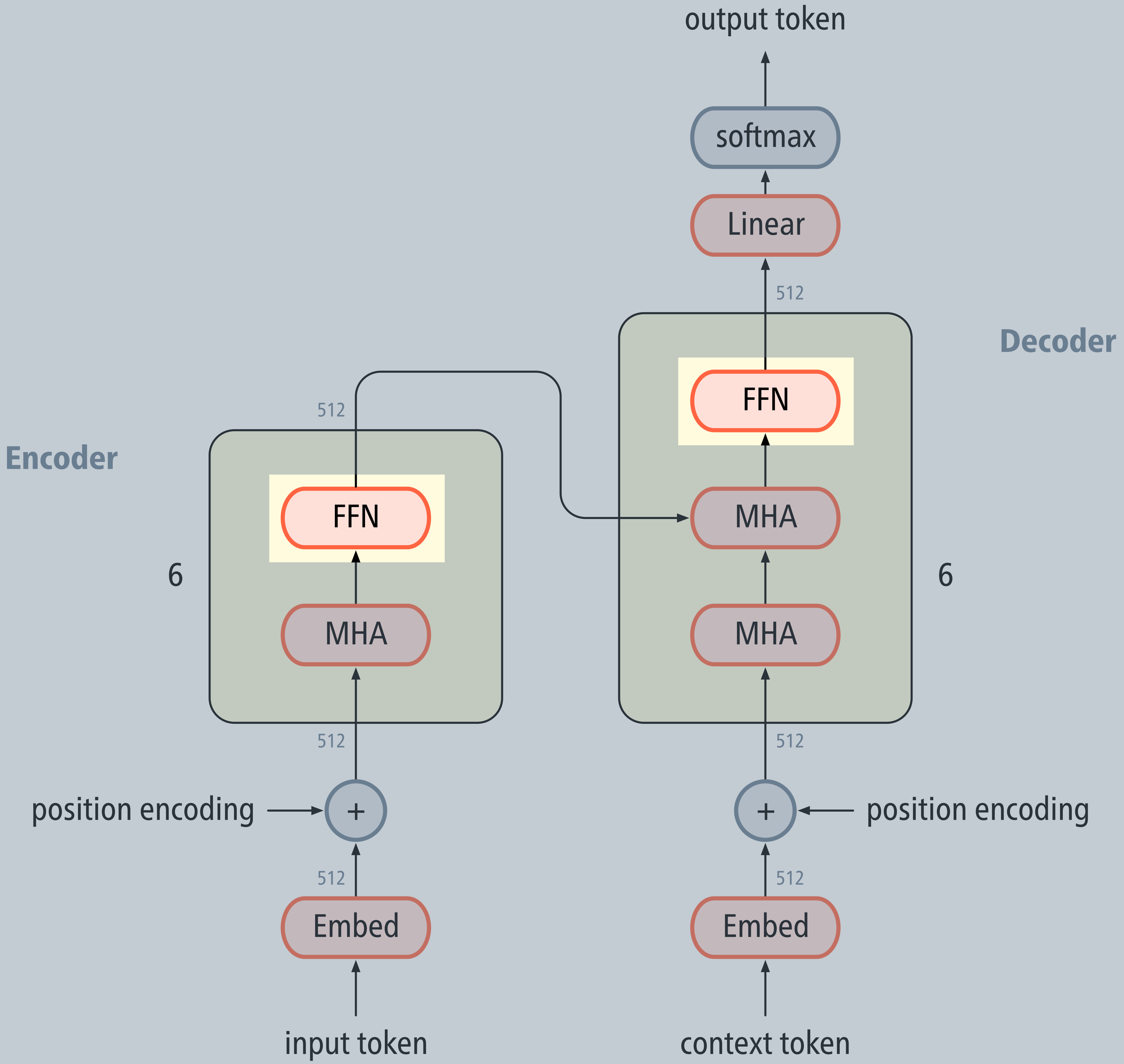

output token
softmax
Linear
512
Decoder
FFN
MHA
MHA
6
512
Encoder
FFN
MHA
6
512
position encoding
+
512
Embed
input token
512
+
position encoding
512
Embed
context token

Cross-attention

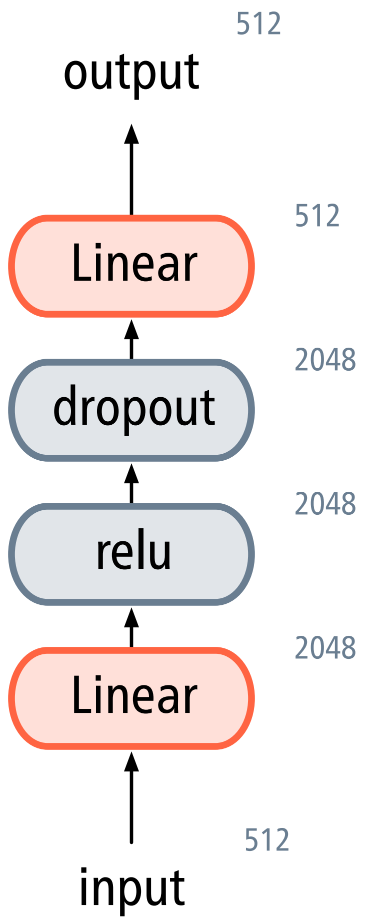


Cross-attention

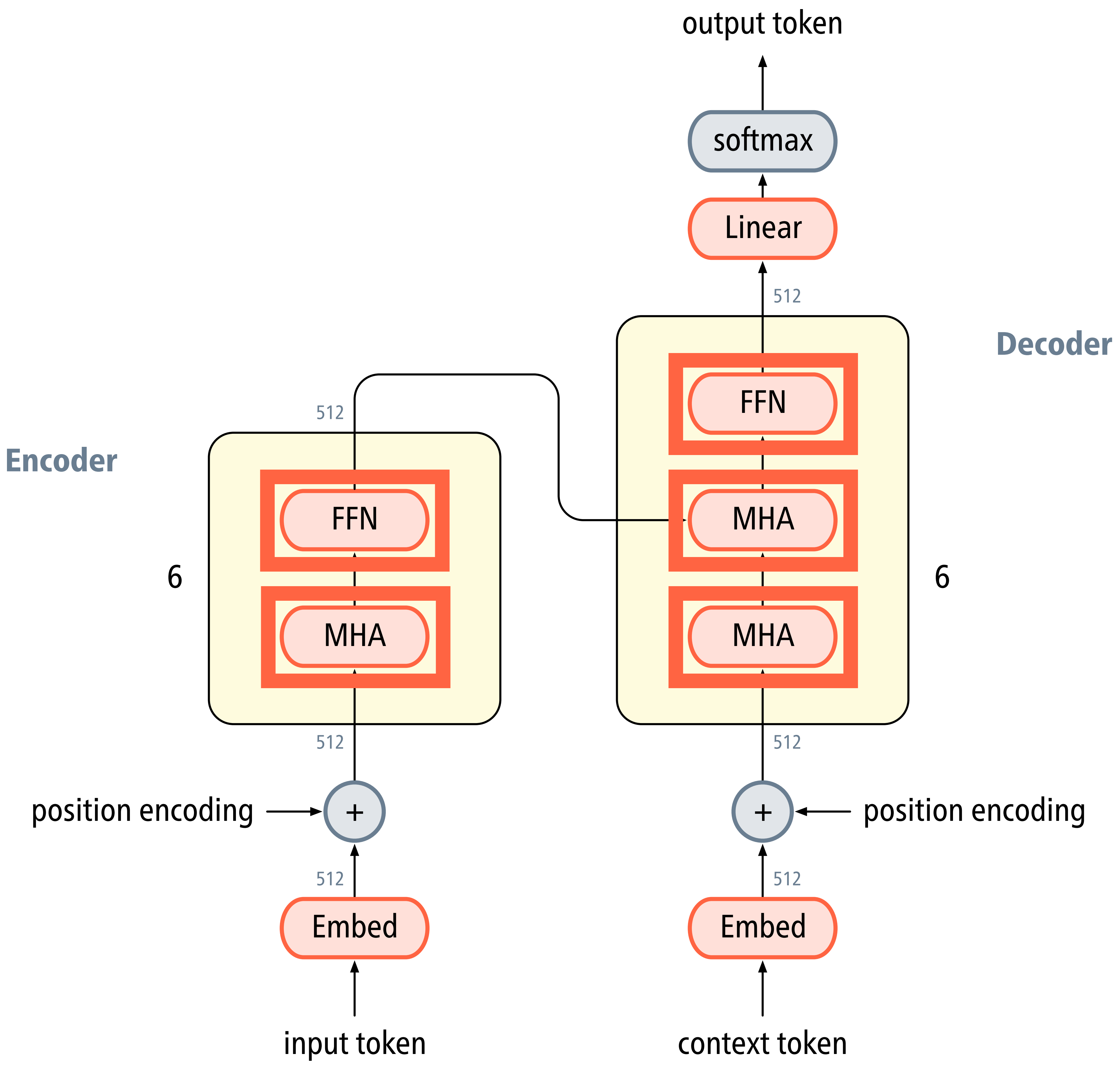



output token
softmax
Linear
512
Decoder
FFN
512
Encoder
FFN
MHA
6
MHA
MHA
6
512
512
position encoding
+
+
position encoding
512
512
Embed
Embed
input token
context token

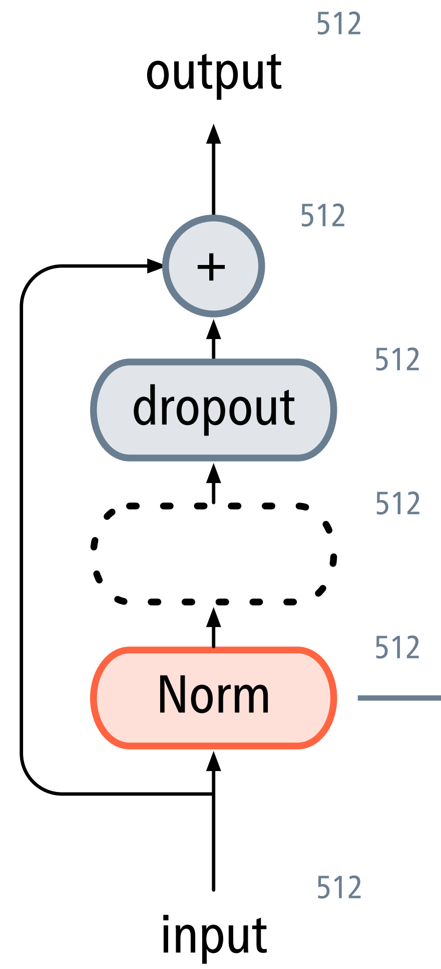
Position-wise feed-forward network



Parameters are shared across positions, but not across blocks.


output token
softmax
Linear
512
Decoder
FFN
MHA
MHA
6
512
Encoder
FFN
MHA
6
512
position encoding
+
512
Embed
input token
512
+
position encoding
512
Embed
context token

Normalise-and-add wrapper



residual connection

layer normalisation

$$\boldsymbol{y} = \frac{\overset{\substack{\text{gain parameter}\\ |}}{\boldsymbol{g}}}{\sigma + \varepsilon} \odot (\boldsymbol{x} - \mu) + \overset{\substack{\text{bias parameter}\\ |}}{\boldsymbol{b}} \qquad \mu = \frac{1}{|\boldsymbol{x}|} \sum_{i=1}^{|\boldsymbol{x}|} x_i \qquad \sigma = \sqrt{\frac{1}{|\boldsymbol{x}|} \sum_{i=1}^{|\boldsymbol{x}|} (x_i - \mu)^2}$$

Ba et al. (2016)

Further details

- Token representations are defined on word pieces computed using byte-pair encoding.
- Embeddings are augmented by position encodings.

  approximate encoding of absolute positions
- Training the model uses several tricks related to batching, masking, loss, and regularisation.

  for details and PyTorch code, see the ‘Annotated Transformer’

Translation performance

	BLEU	FLOPs
GNMT + RL (Wu et al., 2016)	39.92	$1.4 \cdot 10^{20}$
ConvS2S (Gehring et al., 2017)	40.46	$1.5 \cdot 10^{20}$
MoE (Shazeer et al., 2017)	40.56	$1.2 \cdot 10^{20}$
Transformer (big model)	41.80	$2.4 \cdot 10^{19}$

BLEU score and training cost (FLOPs) on the English-to-French newstest2014 test data | Vaswani et al. (2017)