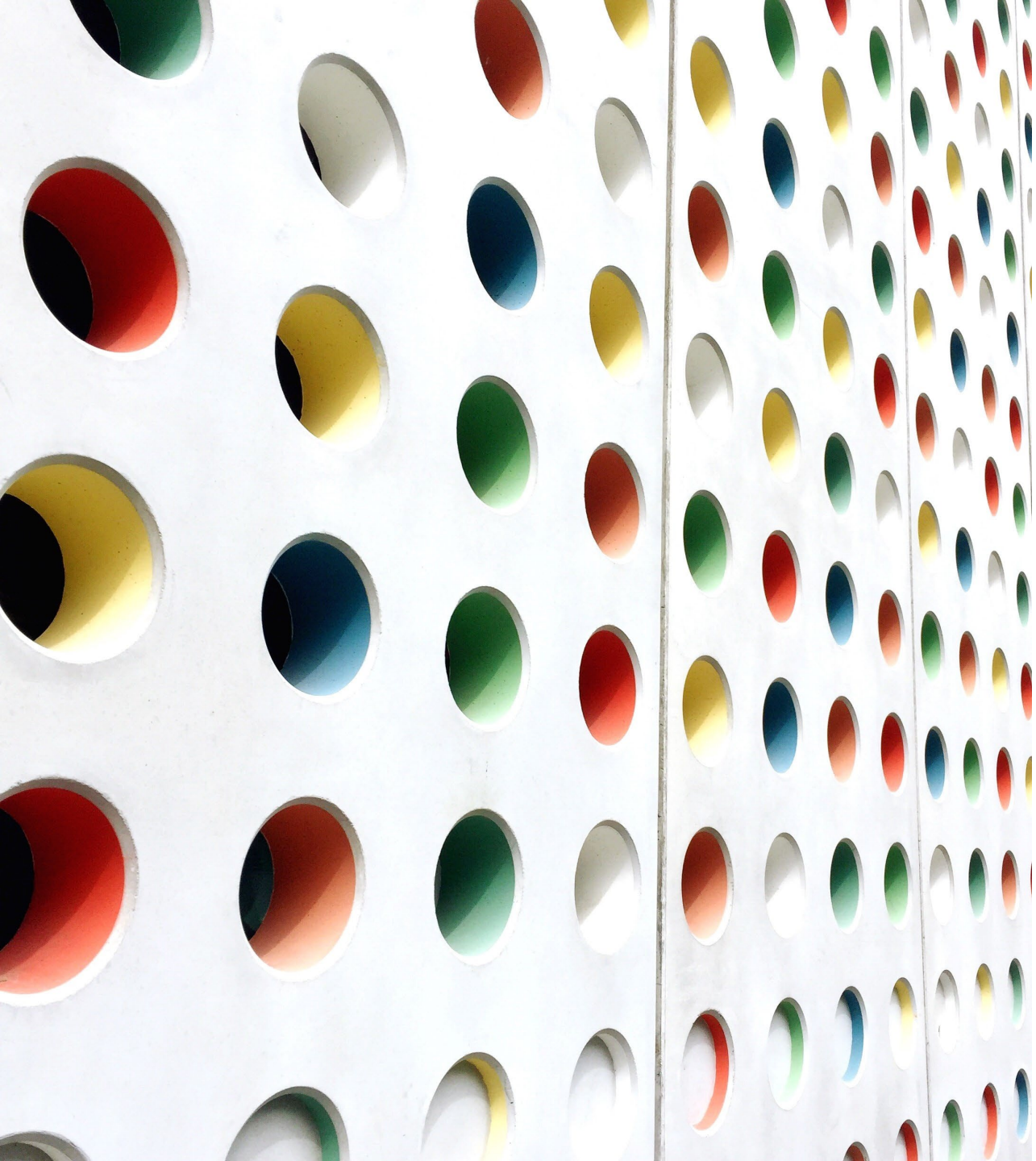


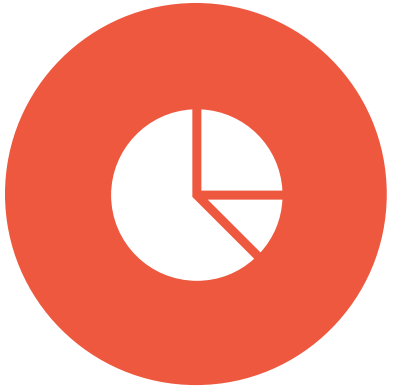
minGPT -vs- Encoder/Decoder

HOW IT WORKS

A VISUAL WALKTHROUGH

DAVE BROWN
SIGNALPOP LLC
WWW.SIGNALPOP.COM

Encoder/Decoder Model Use Cases



LANGUAGE TRANSLATION

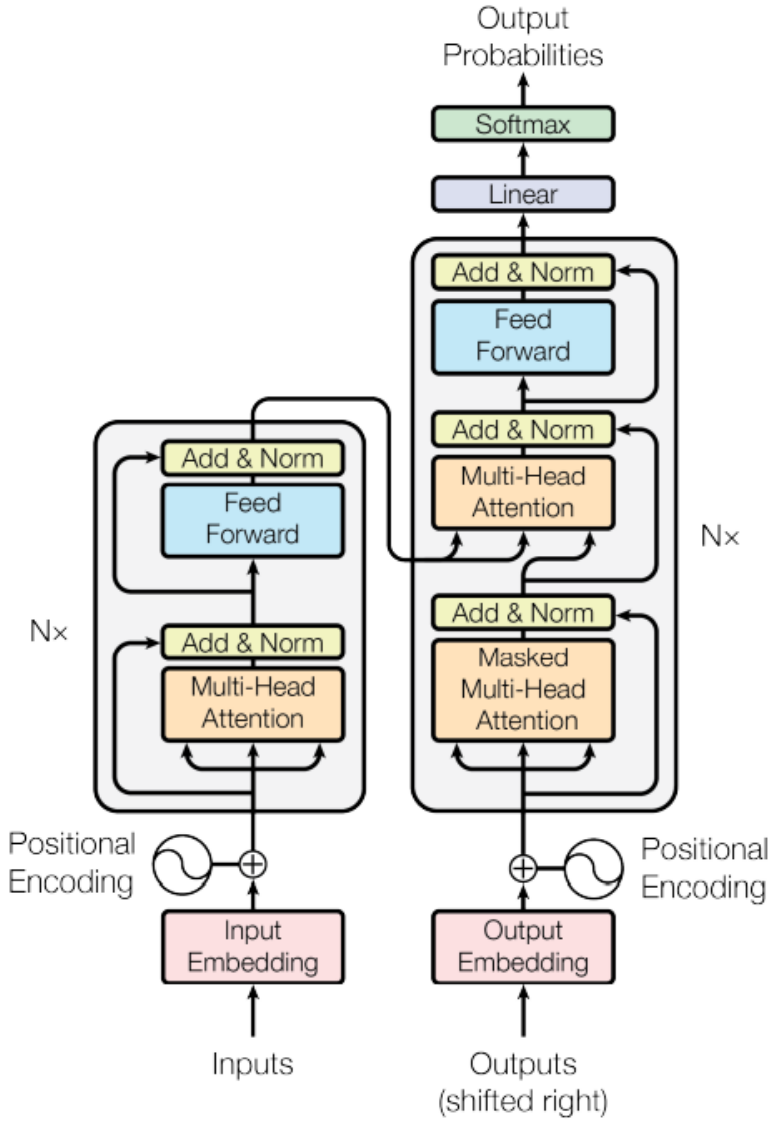
NATURAL LANGUAGE PROCESSING

TIME SERIES

Attention is ALL you need!

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

"Attention is all you need", Vaswani et al. arXiv, 2017
https://arxiv.org/abs/1706.03762



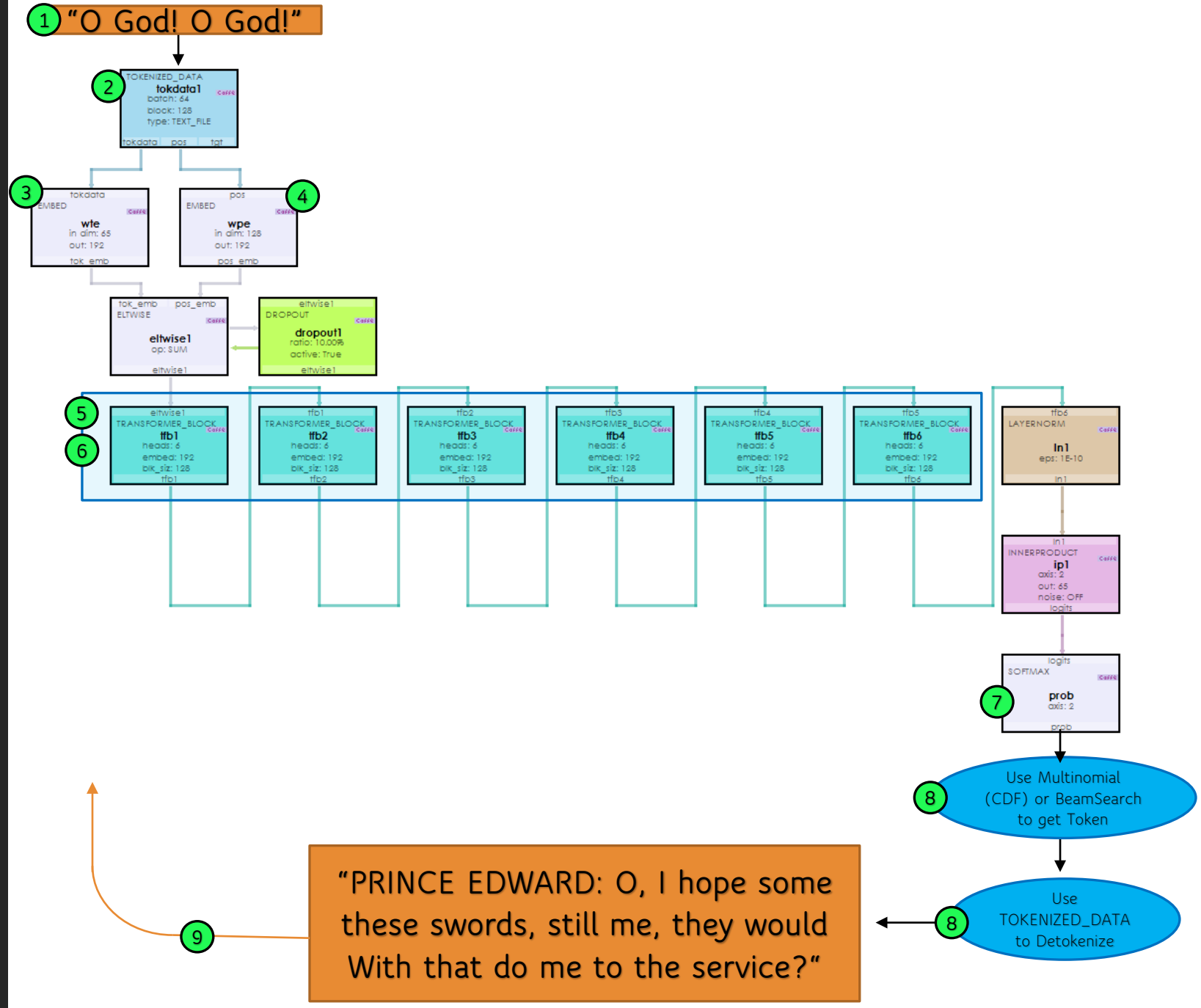
From "Attention Is All You Need", Fig1

minGPT in action

1. Input sentence
2. Tokenize it (char, word, etc.)
3. Create token embedding and
4. Position encoding (e.g., sin(w))
5. Encoder creates encoding
6. Run self attention on encoding
7. Softmax creates probabilities
8. Detokenize
9. Produce predicted next char/word.

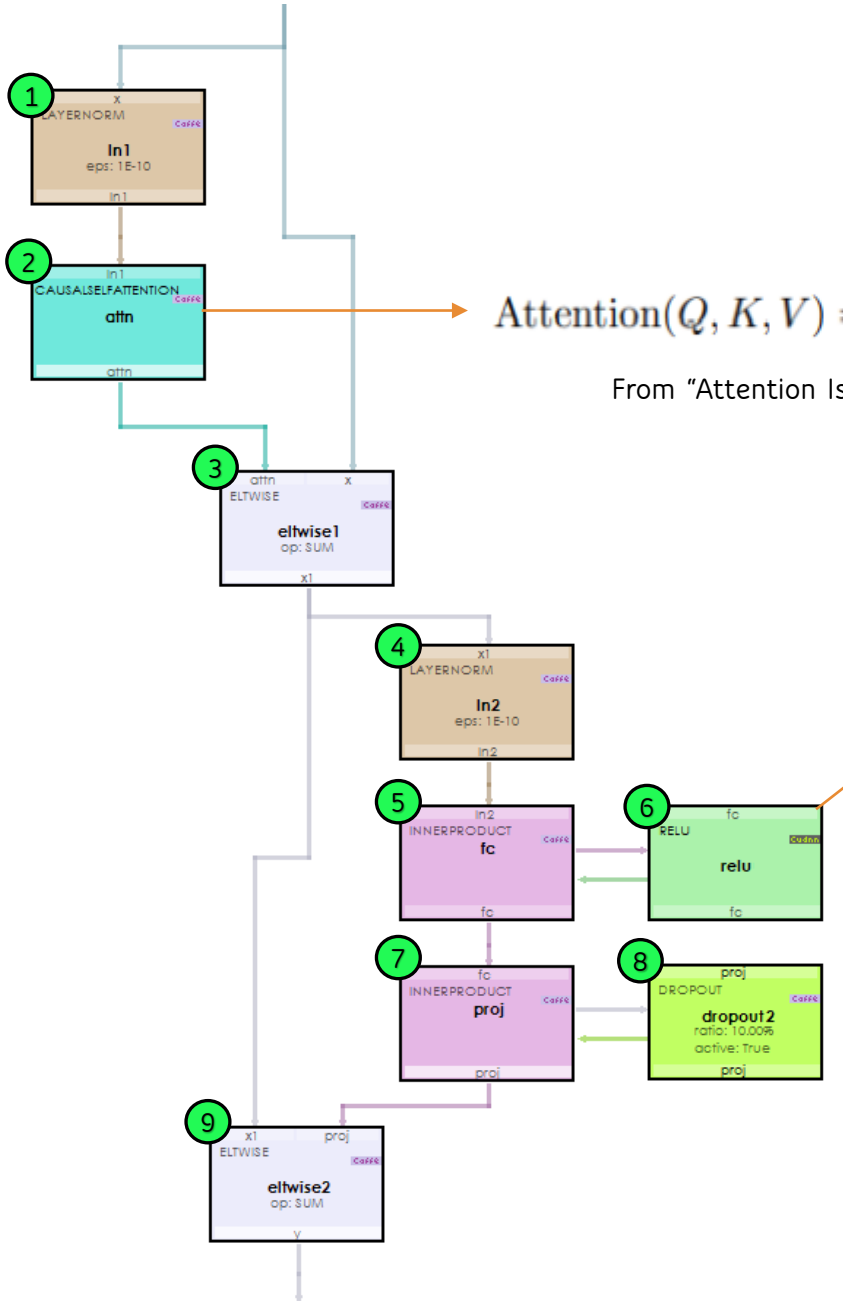
GPT-mini: 6 layers, 6 heads, 192 embedding size

GPT2-Large: 36 layers, 20 heads, 1280 embedding size = 774M parameters



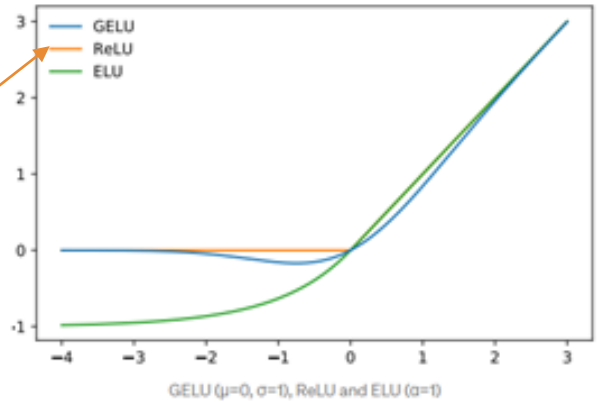
Transformer Block Internals

1. Layer Normalization 1
2. Causal Self Attention
3. Add X to Attn
4. Layer Normalization 2
5. Fc Inner Product
6. Activation (RELU, GELU, etc.)
7. Projection Inner Product
8. Dropout
9. Add proj to X1 (from #3 above)



$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

From "Attention Is All You Need"

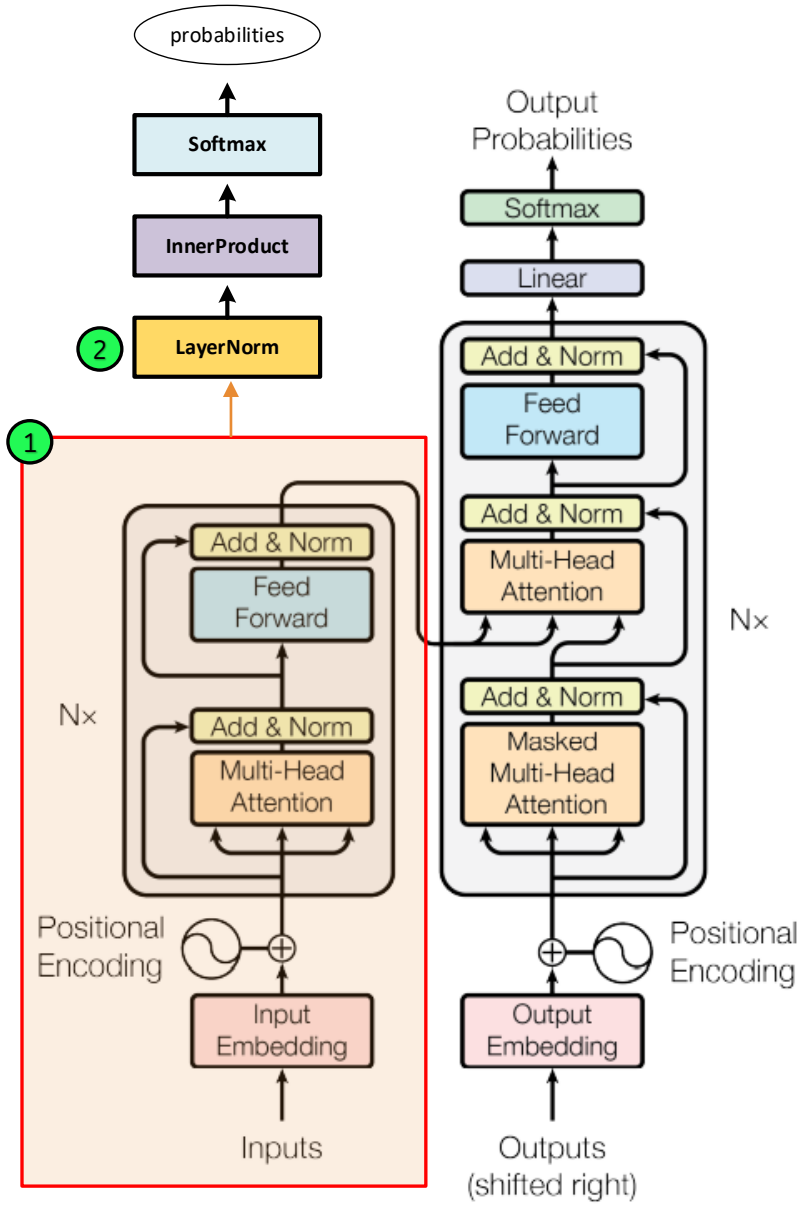


From "GELU activation"

minGPT vs full encode/decode model

1. minGPT only implements the 'encoder' side of the full model.
2. Just add LayerNorm + InnerProduct + Softmax to get output probabilities

How do we convert GPT to an Encoder/Decoder model?



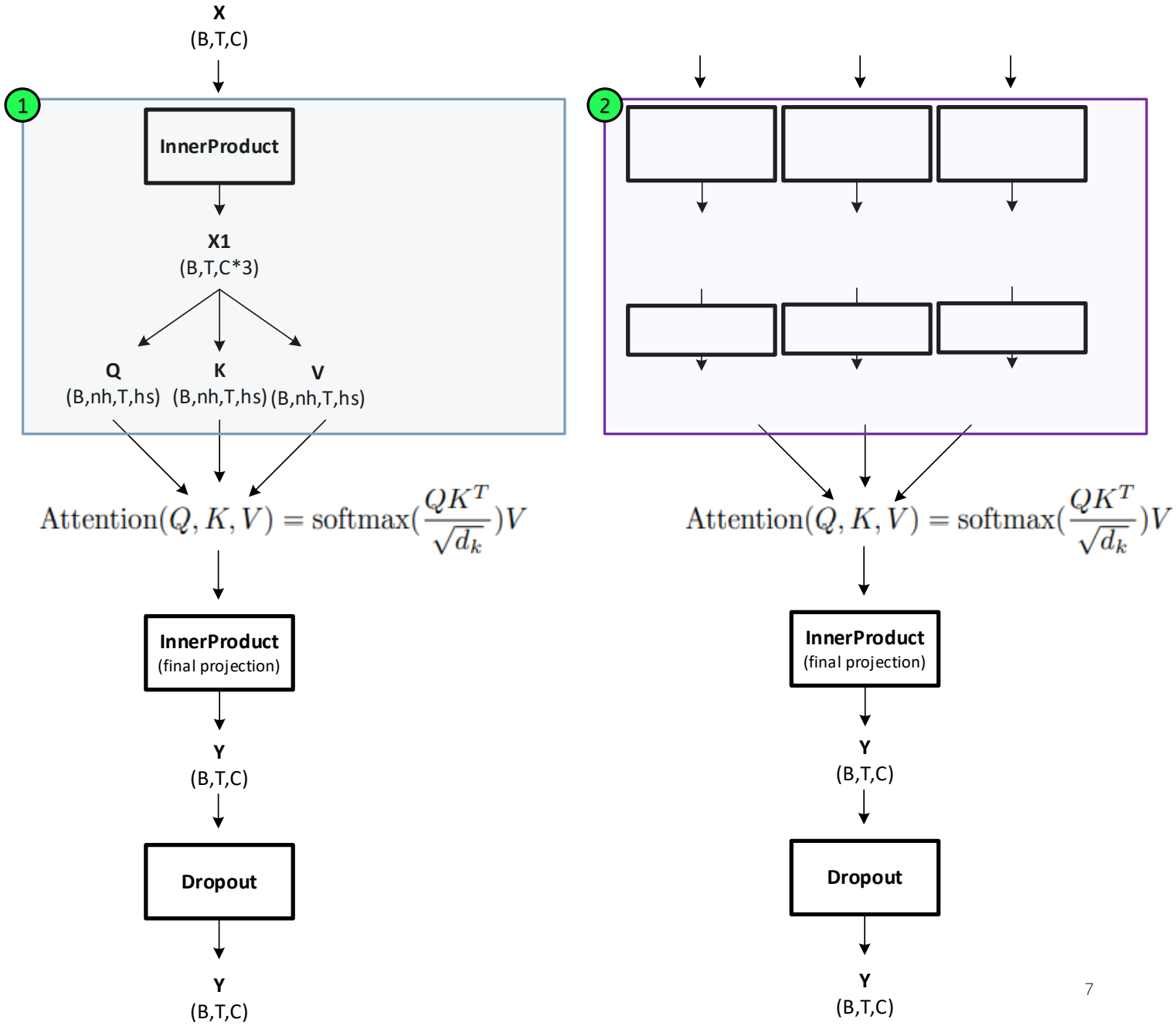
From "Attention Is All You Need", Fig1

First, convert …

CausalSelfAttention to MultiHeadedAttention

1. CausalSelfAttention takes one input 'X' and converts it to Q, K, V using a single InnerProduct

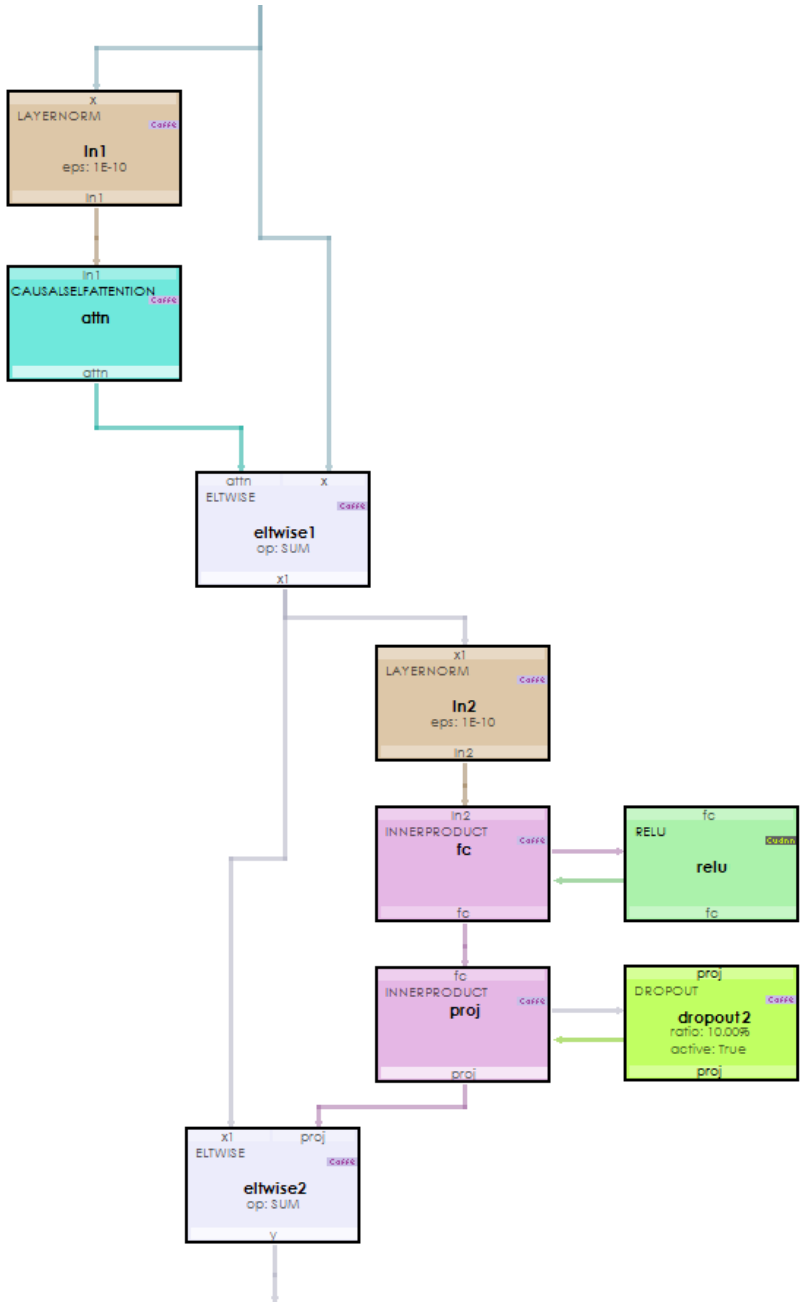
2. MultiHeadedAttention takes 3 inputs that are converted to Q, K, V using three InnerProducts.



Decoder Transformer Block Internals

Starting with Encoder Transformer Block (Transformer Block in GPT)…

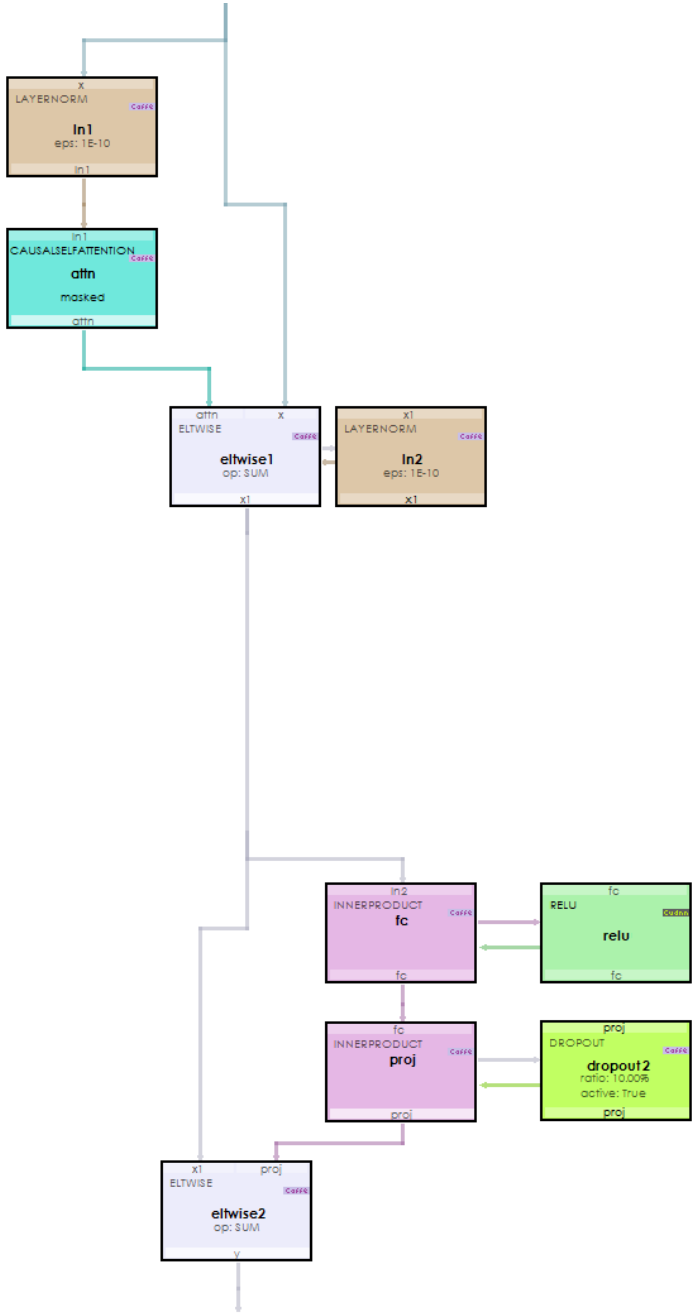
Encoder Transformer Block



Decoder Transformer Block Internals

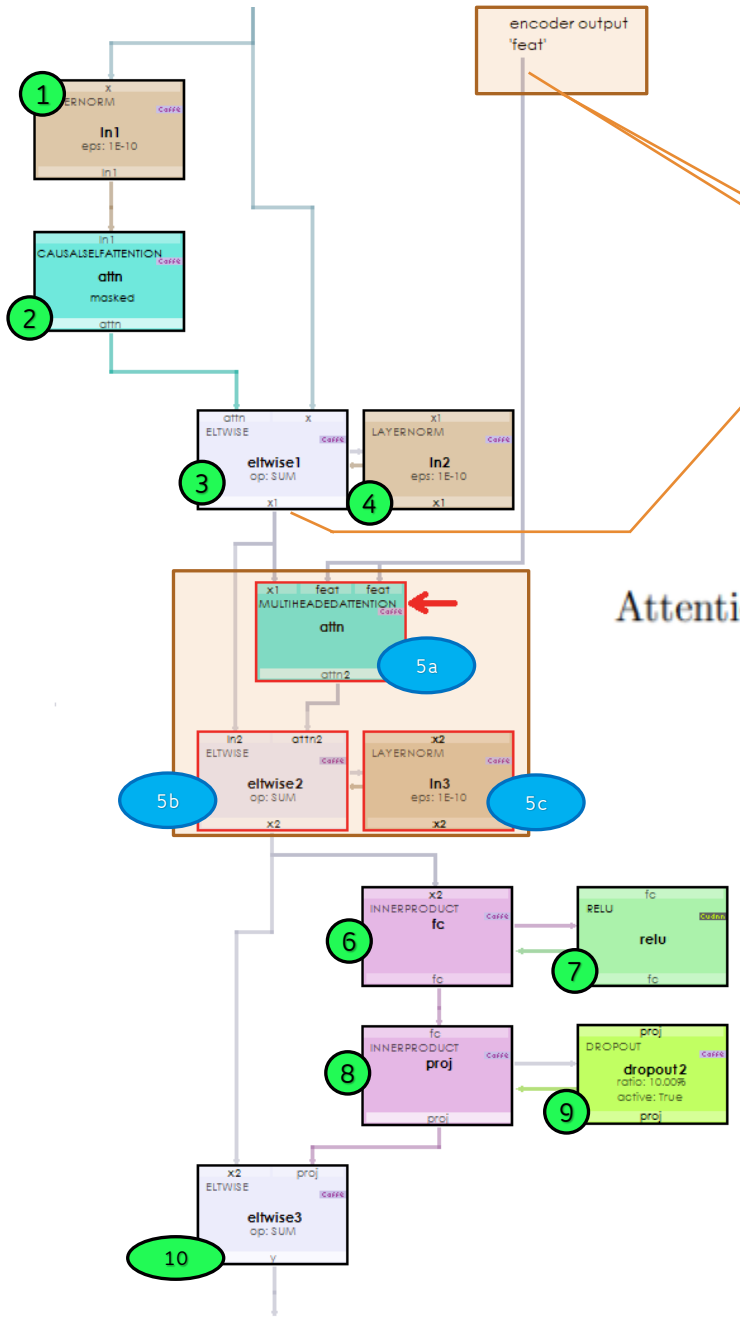
Make room for new layers used by Decoder Transformer Block…

Encoder Transformer Block



Decoder Transformer Block Internals

1. Layer Normalization 1
2. Causal Self Attention
3. Add X to Attn
4. Layer Normalization 2
5. 5a. Multi-Head Attention
6. 5b. Add X1 to Attn2
7. 5c. Layer Normalization 3
8. Fc Inner Product
9. Activation (RELU, GELU, etc.)
10. Projection Inner Product
11. Dropout
12. Add proj to X2 (from #5c above)



Learning to match input and target probability distributions occurs here.

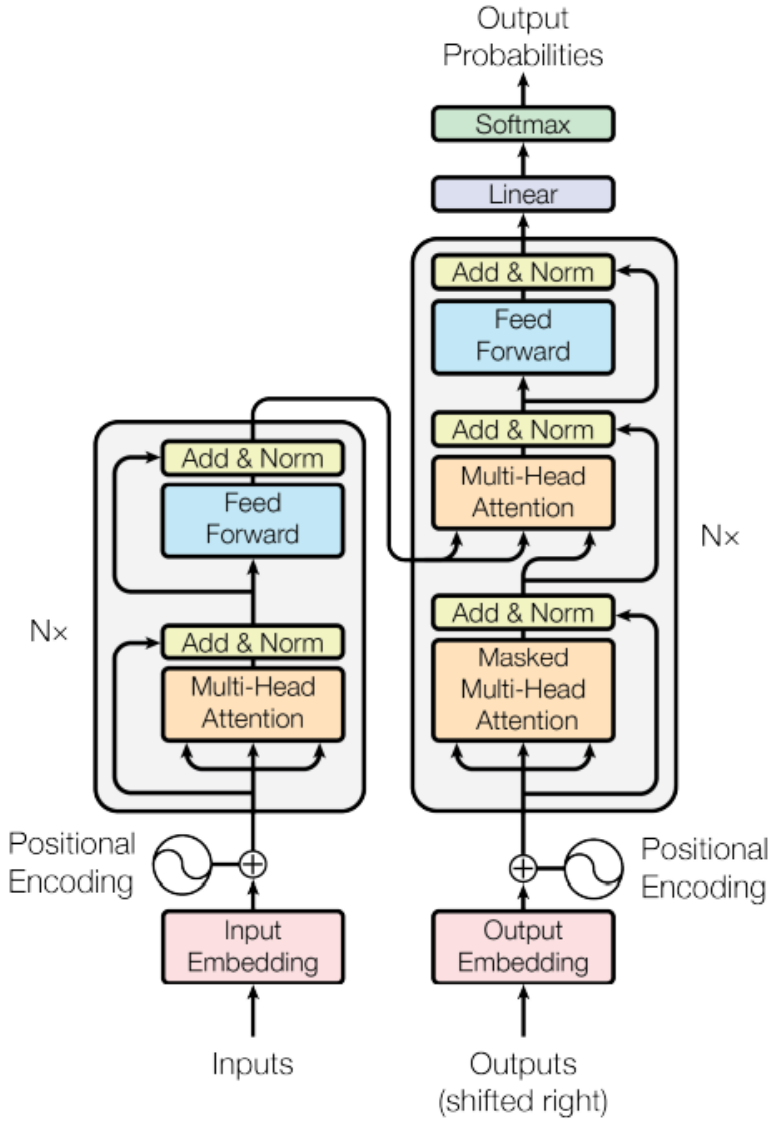
The same encoder output is sent to each decoder transformer block layer.

Q = x1
K = feat
V = feat

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

minGPT vs full encode/decode model

Full model uses both encode (left) and decode (right).



From "Attention Is All You Need", Fig1

Full encode/decode model

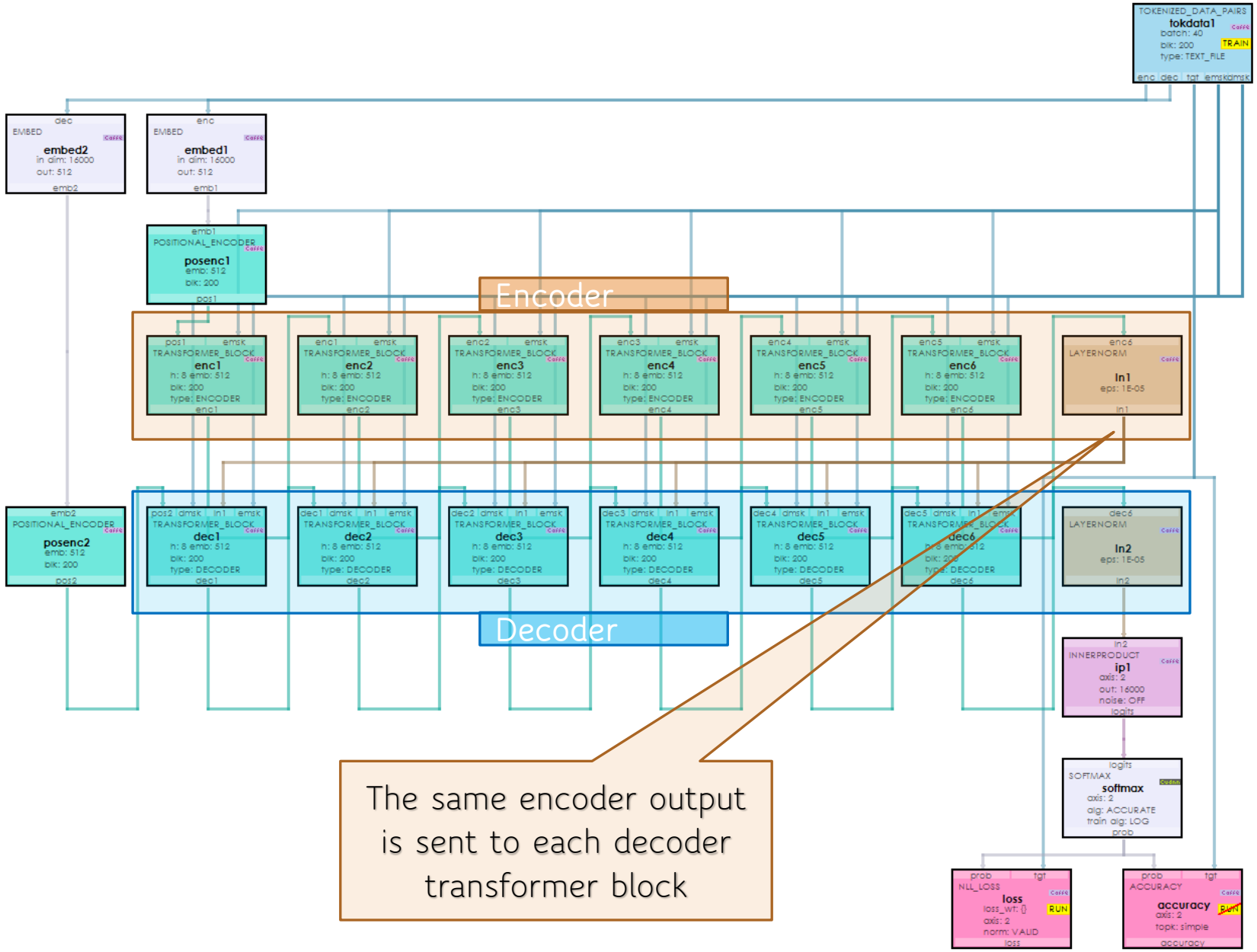
Full encoder/decoder model based off original minGPT model.

Note:

Target values are offset by one and may use a separate positional embedding.

Also Note:

Decoder TRANSFORMER_BLOCKS have ‘decode’ enabled which uses both the **CausalSelfAttention** and **MultiHeadAttention** layers internally.



Full encode/decode model

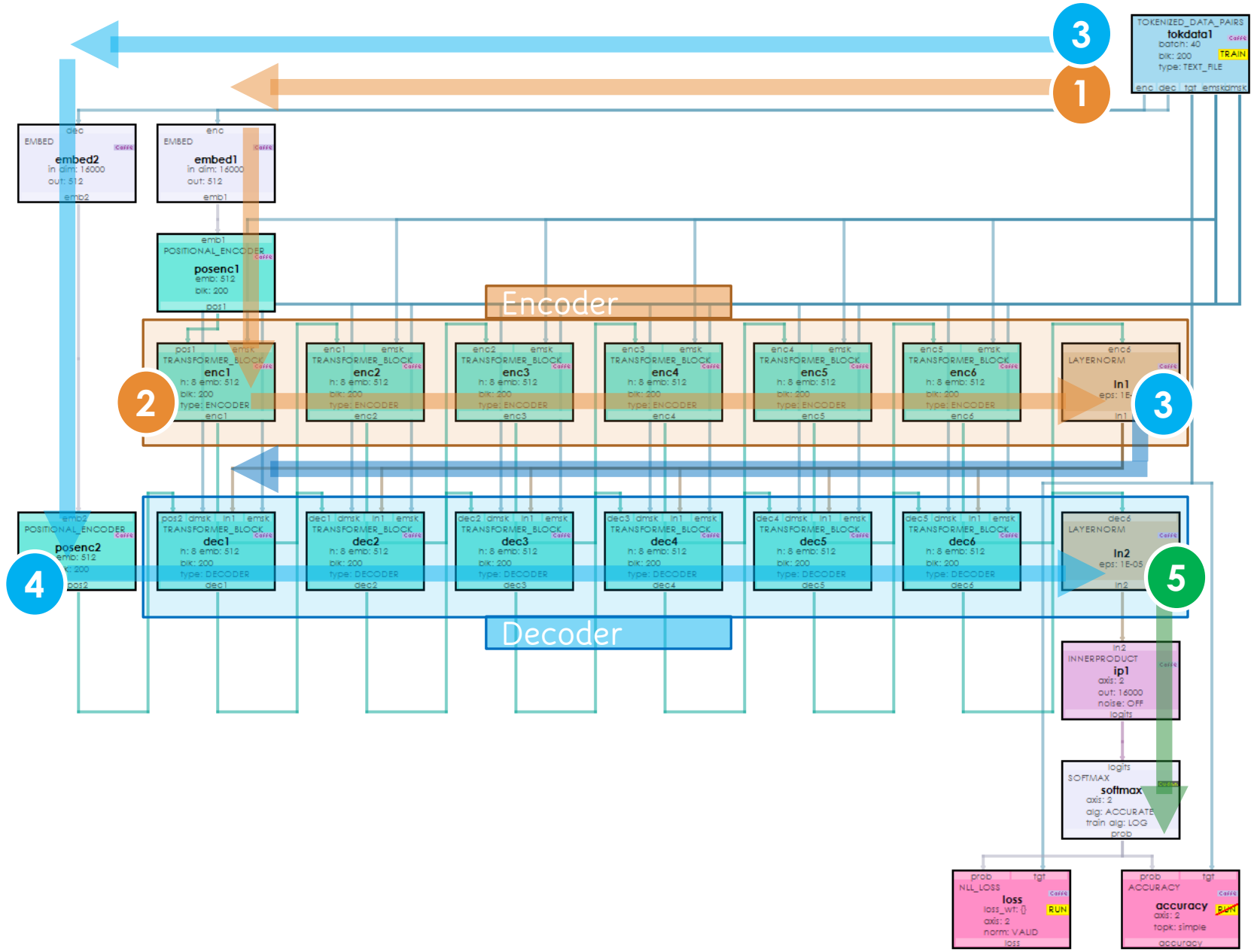
Full encoder/decoder model based off original minGPT model.

Note:

Target values are offset by one and may use a separate positional embedding.

Also Note:

Decoder TRANSFORMER_BLOCKS have ‘decode’ enabled which uses both the **CausalSelfAttention** and **MultiHeadAttention** layers internally.

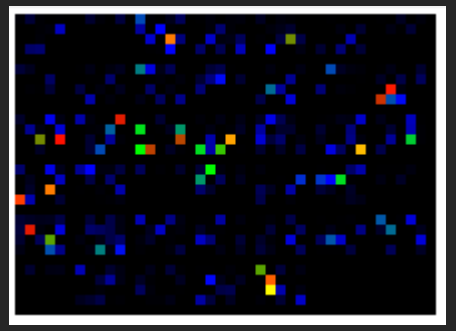


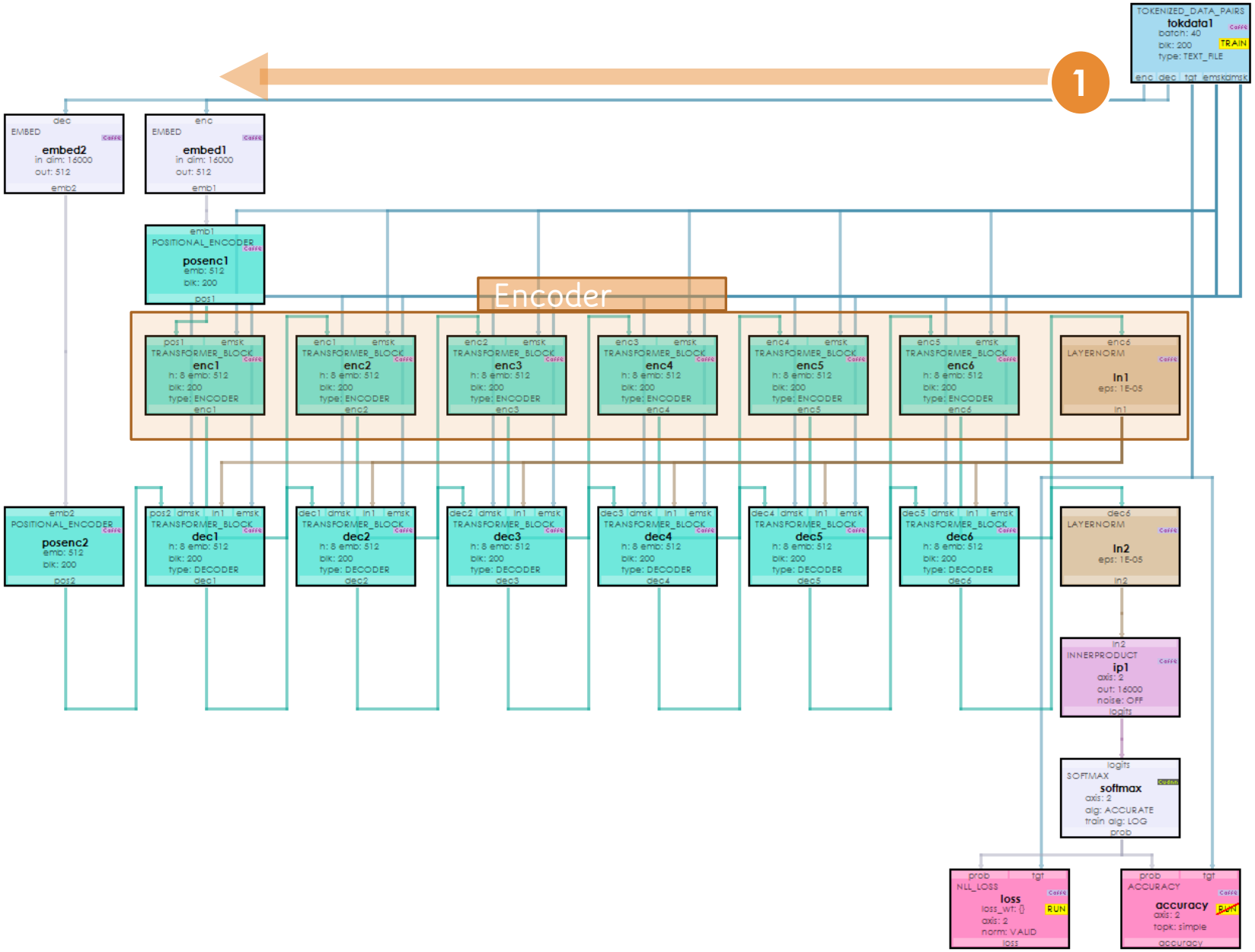
1 encoder in tokens

Encoder Input Tokens

Vocabulary Size = 16,000

"When does the European Union session start?"

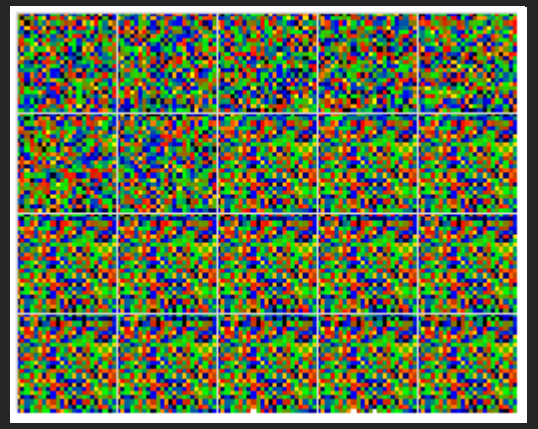


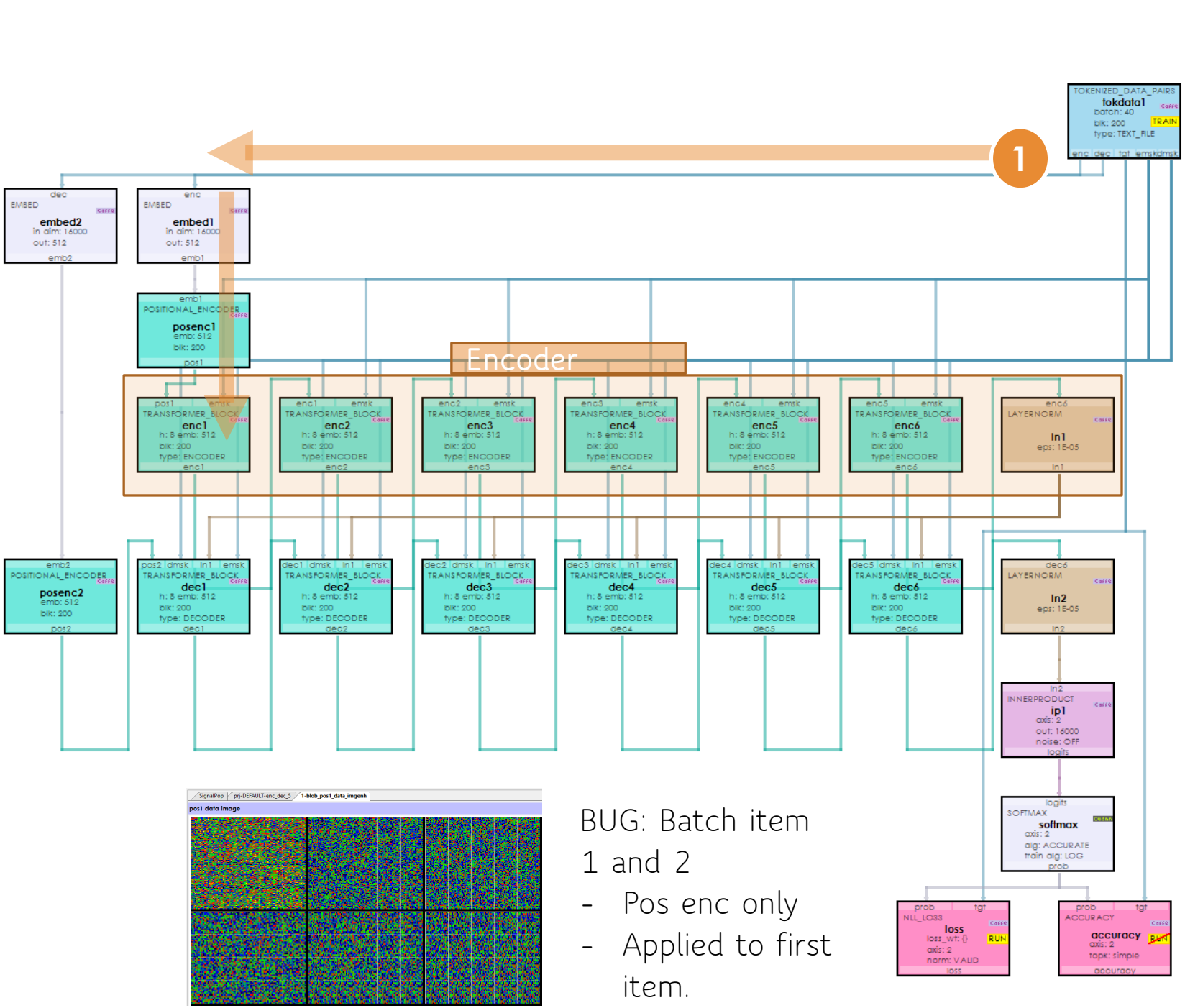


1 encoder in embedding

Embedding to 512 emb size

+ Position Embedding





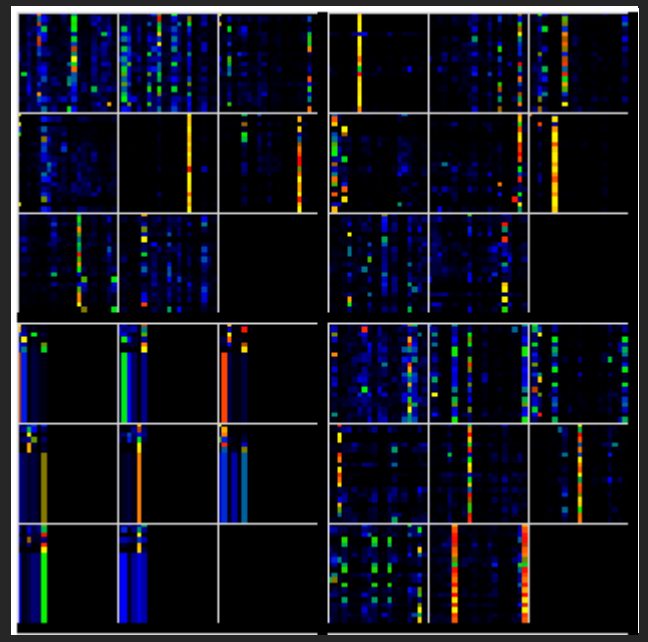
BUG: Batch item 1 and 2

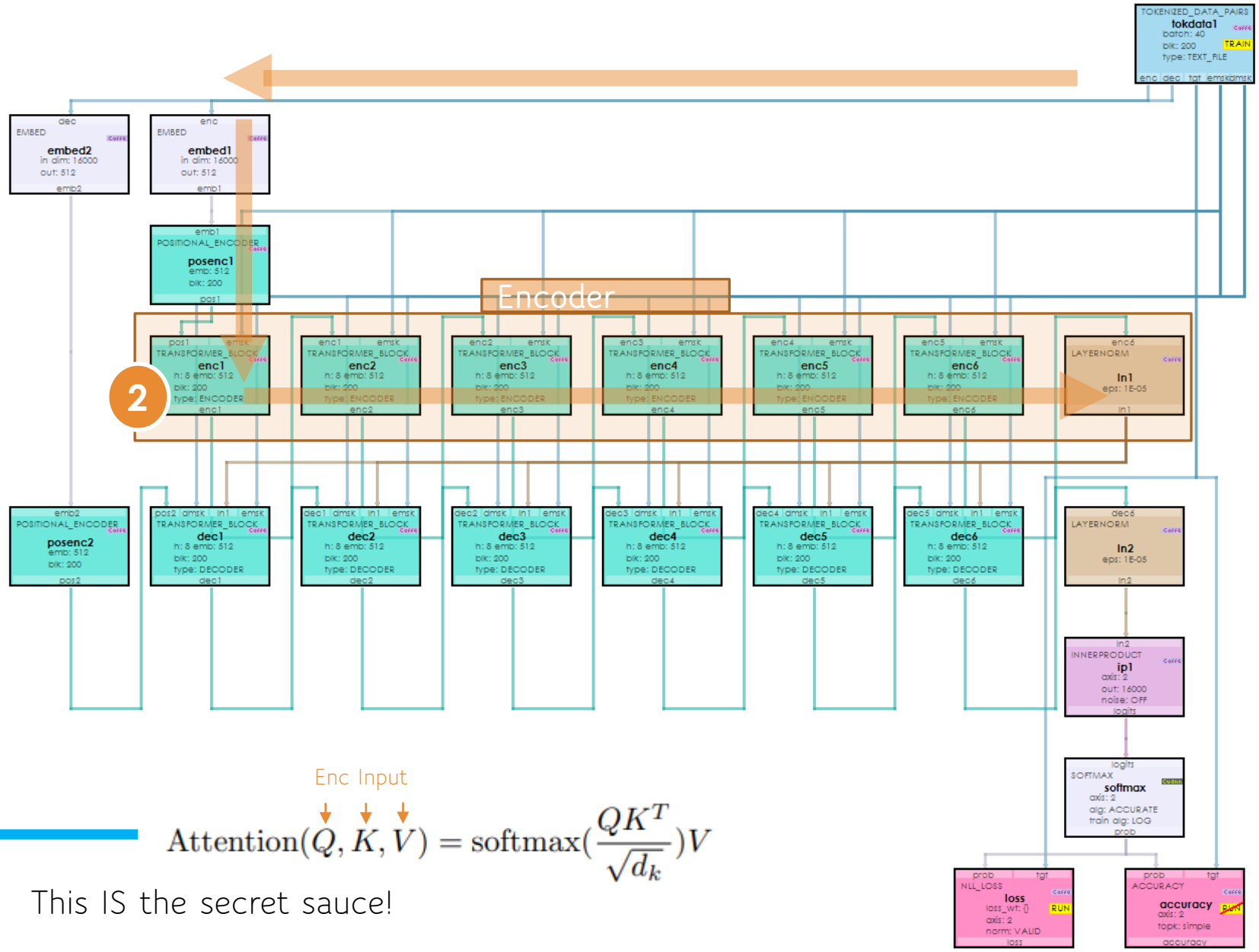
- Pos enc only
- Applied to first item.

2 encoder transformer

Encoder Transformer

Mask,, Softmax (self attention)…





Enc Input

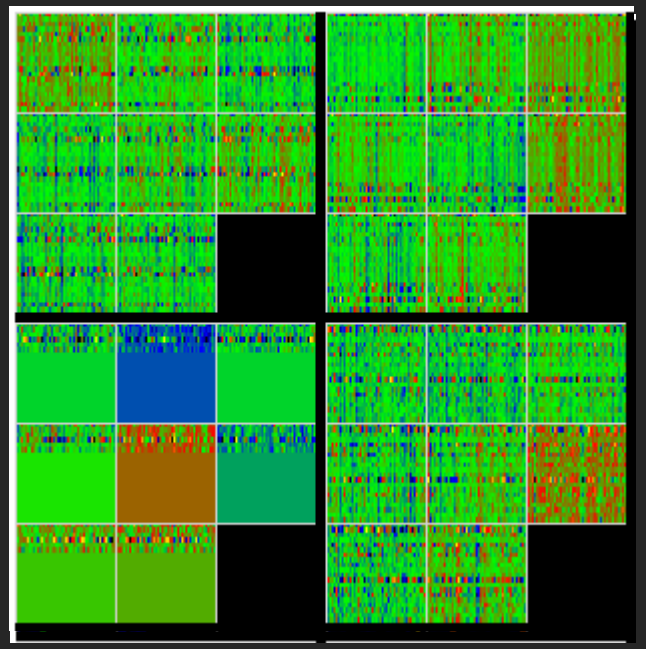
$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

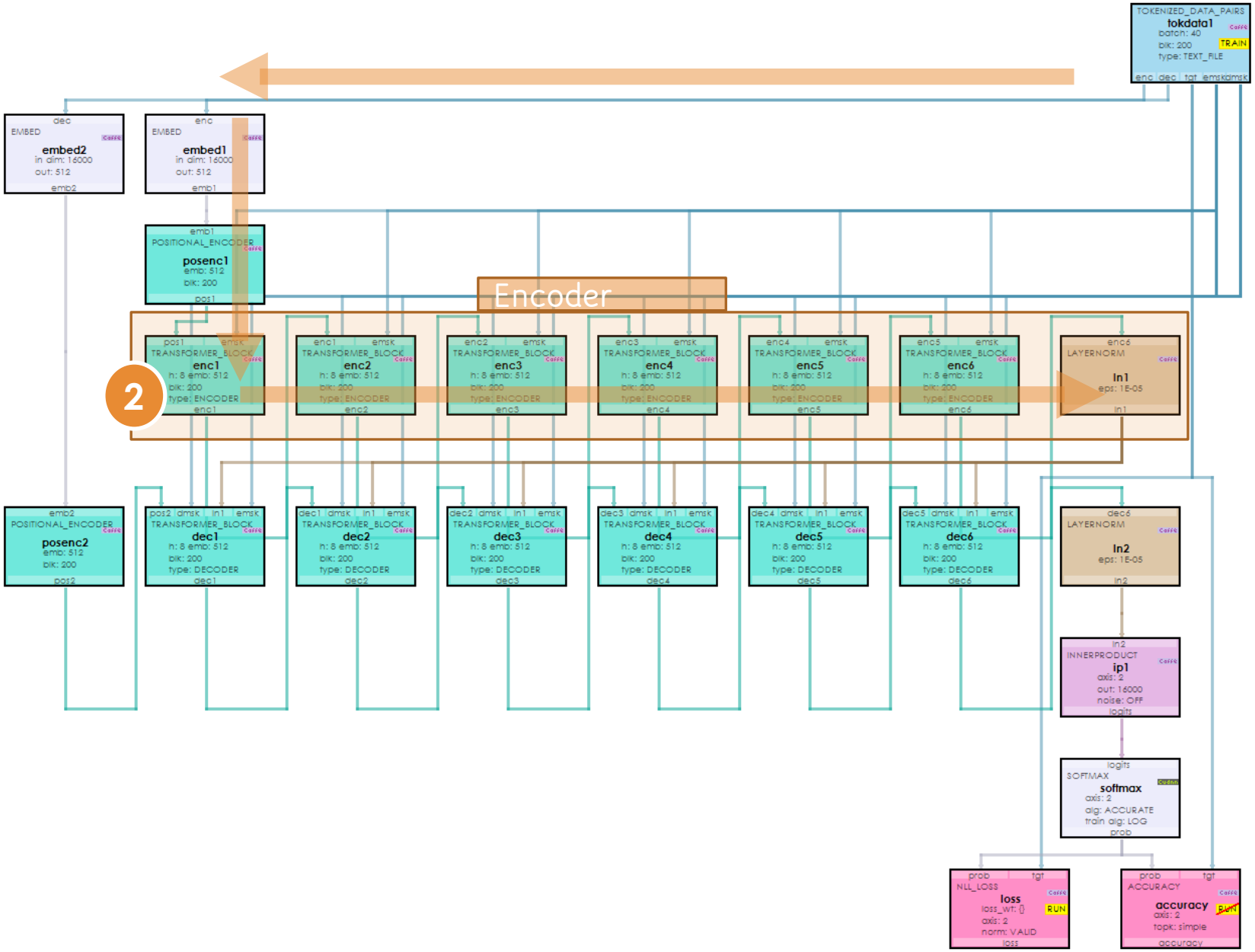
This IS the secret sauce!

2 encoder transformer

Encoder Transformer

Mask, Softmax,. Projection…



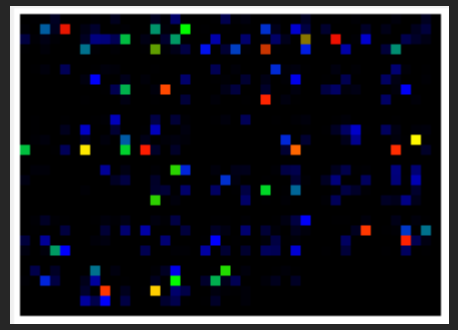


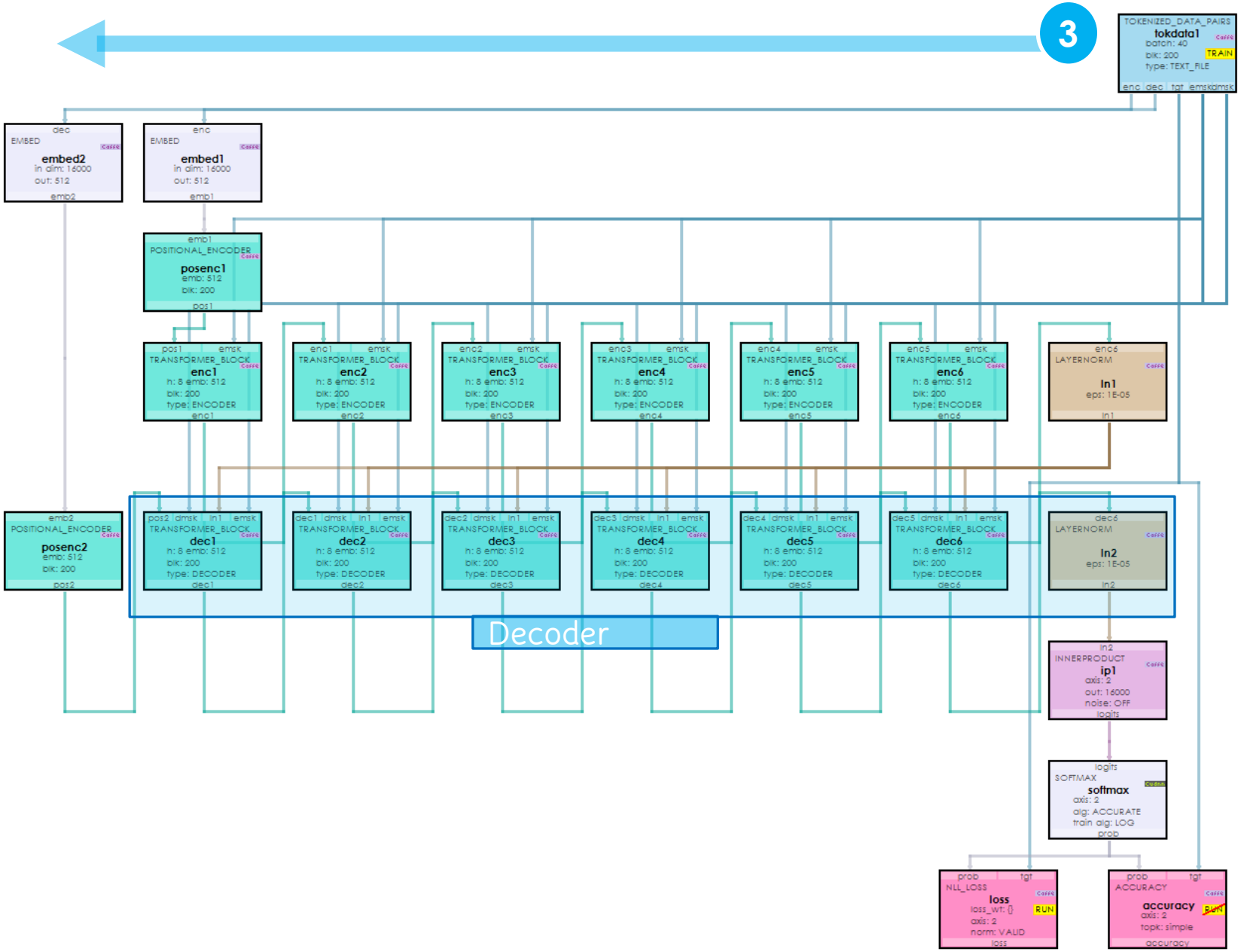
3 decoder in tokens

Decoder Input Tokens

Vocabulary Size = 16,000

Quand commence la session de l'Union européenne?

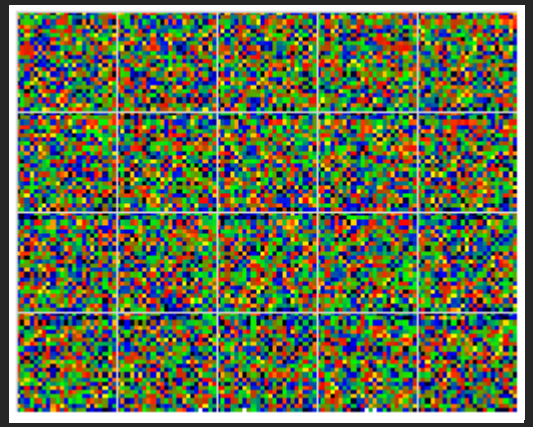


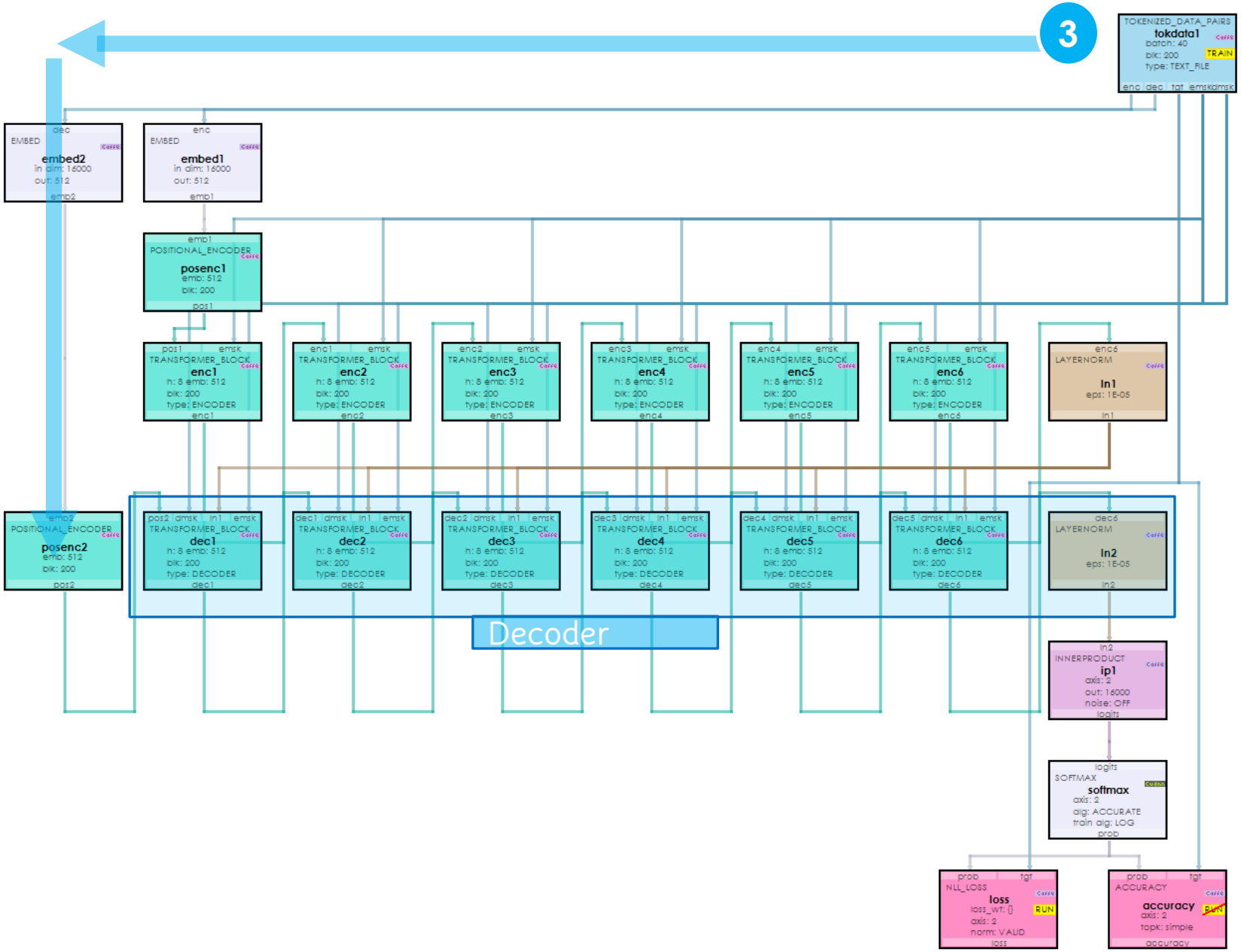


3 decoder in embedding

Embedding to 512 emb size

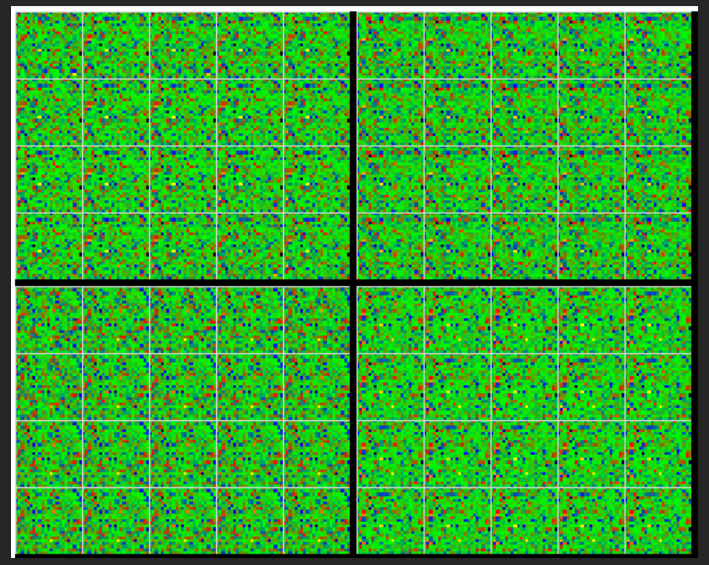
+ Position Embedding

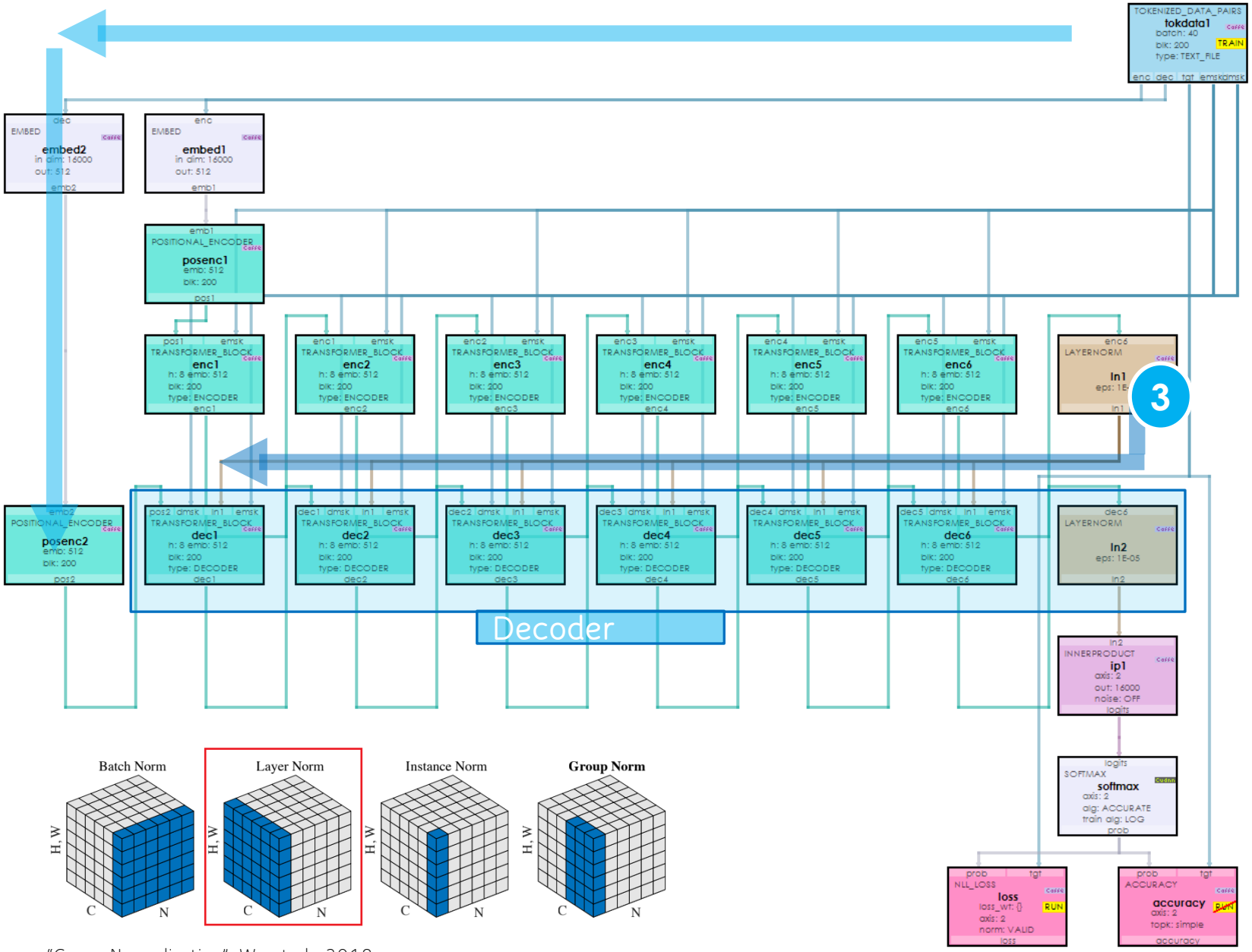




3 decoder transformer

Last encoder output (layer normalized)…



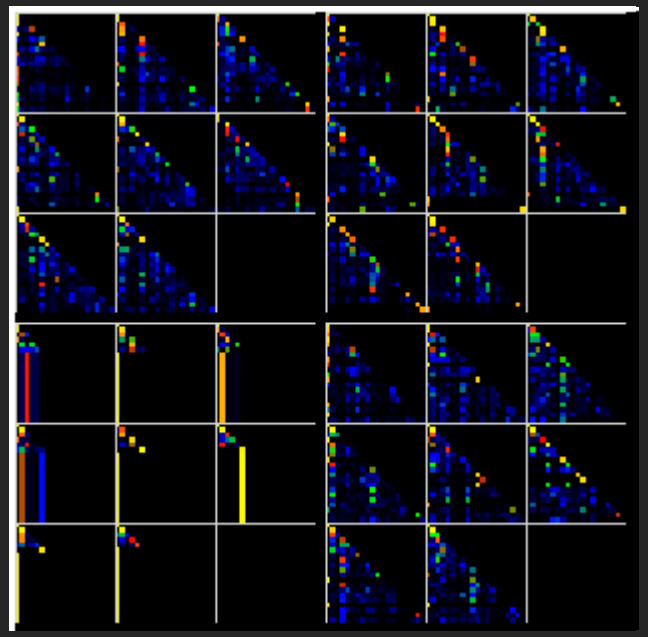


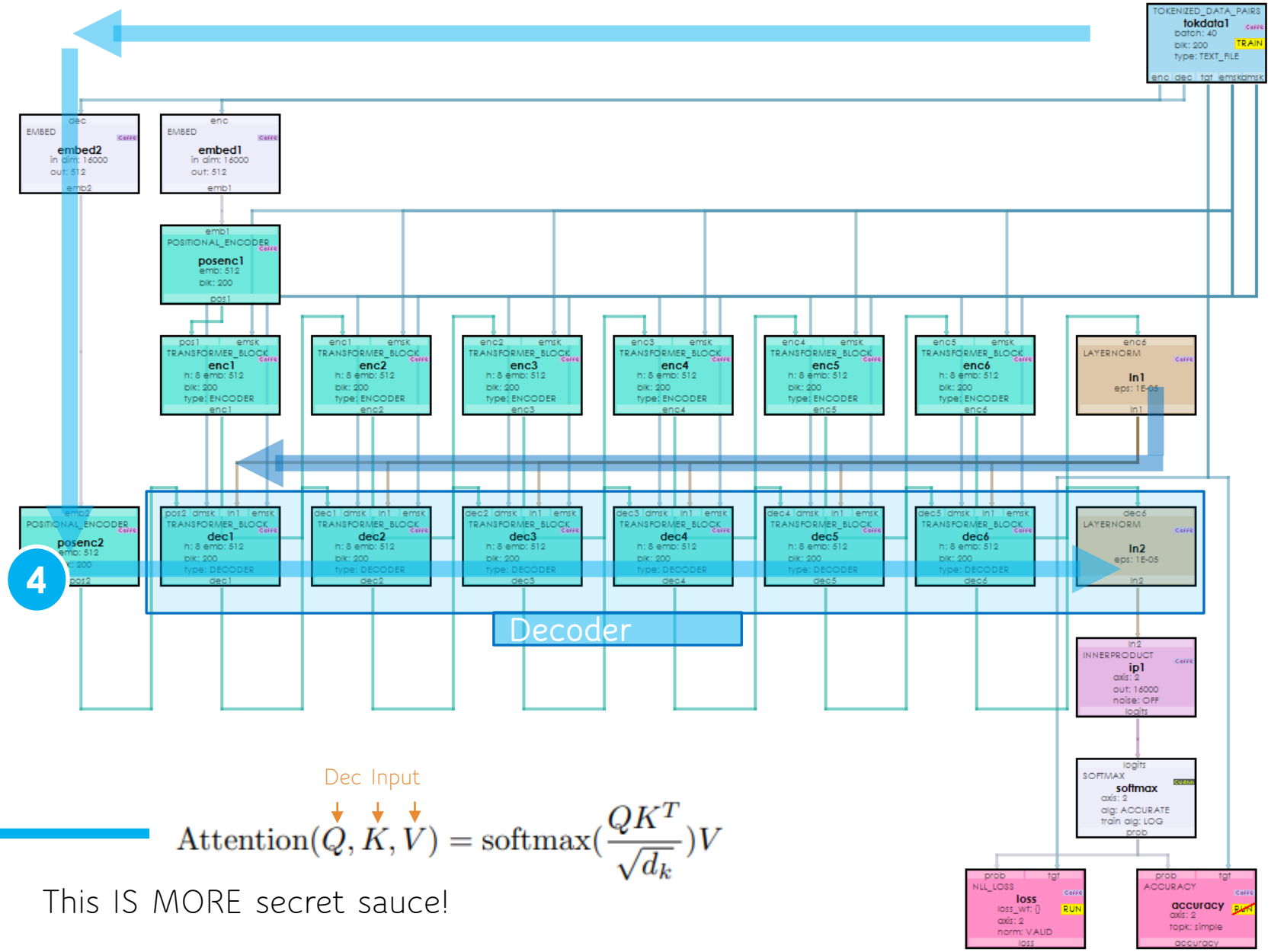
"Group Normalization", Wu et al., 2018

4 decoder transformer (multi-head 1)

Decoder Transformer

Mask, (Softmax(self attention)… right)…





Dec Input

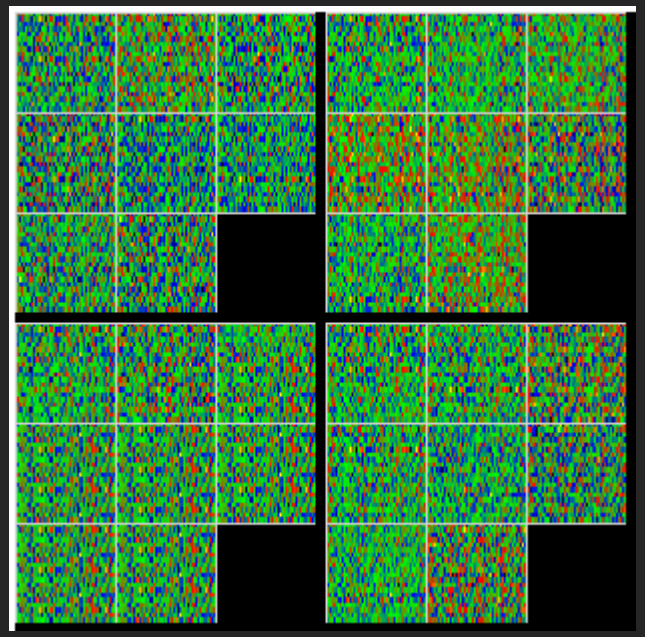
$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

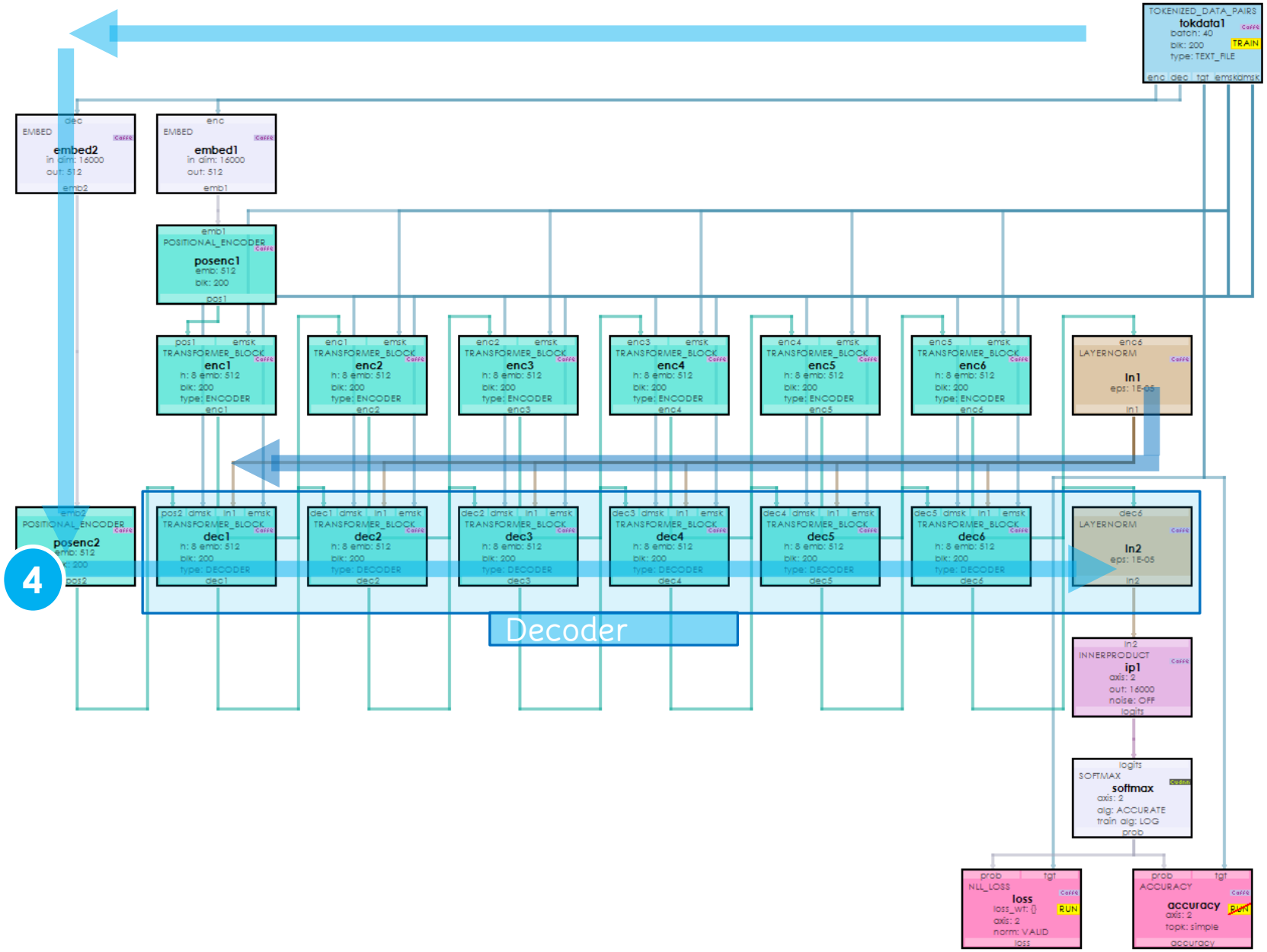
This IS MORE secret sauce!

4 decoder transformer (multi-head 1)

Decoder Transformer

Mask, Softmax( ) projection…

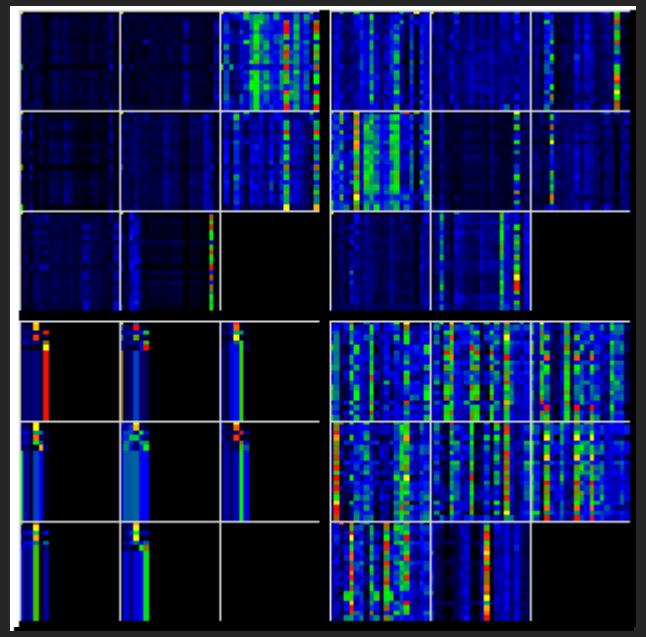


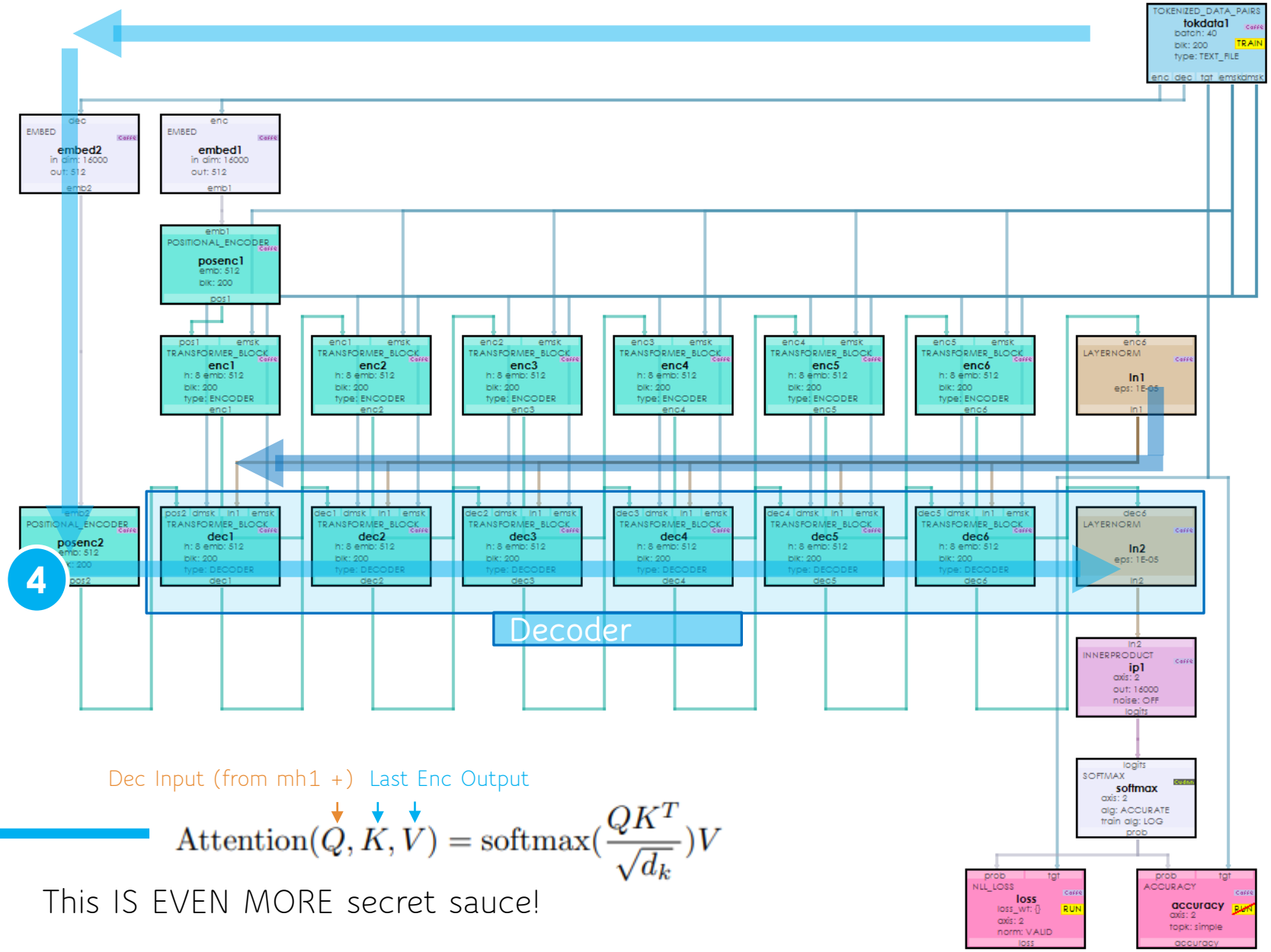


4 decoder transformer (multi-head 2)

Decoder Transformer

Mask„ Softmax (attention)…





Dec Input (from mh1 +) Last Enc Output

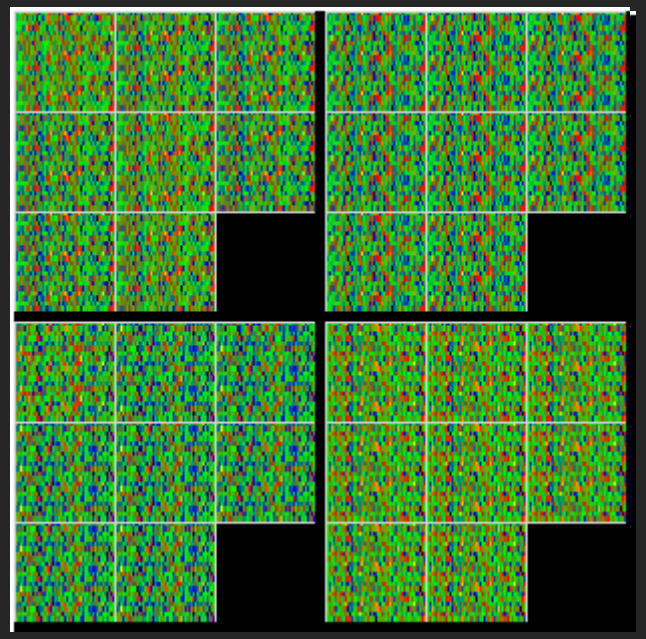
$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

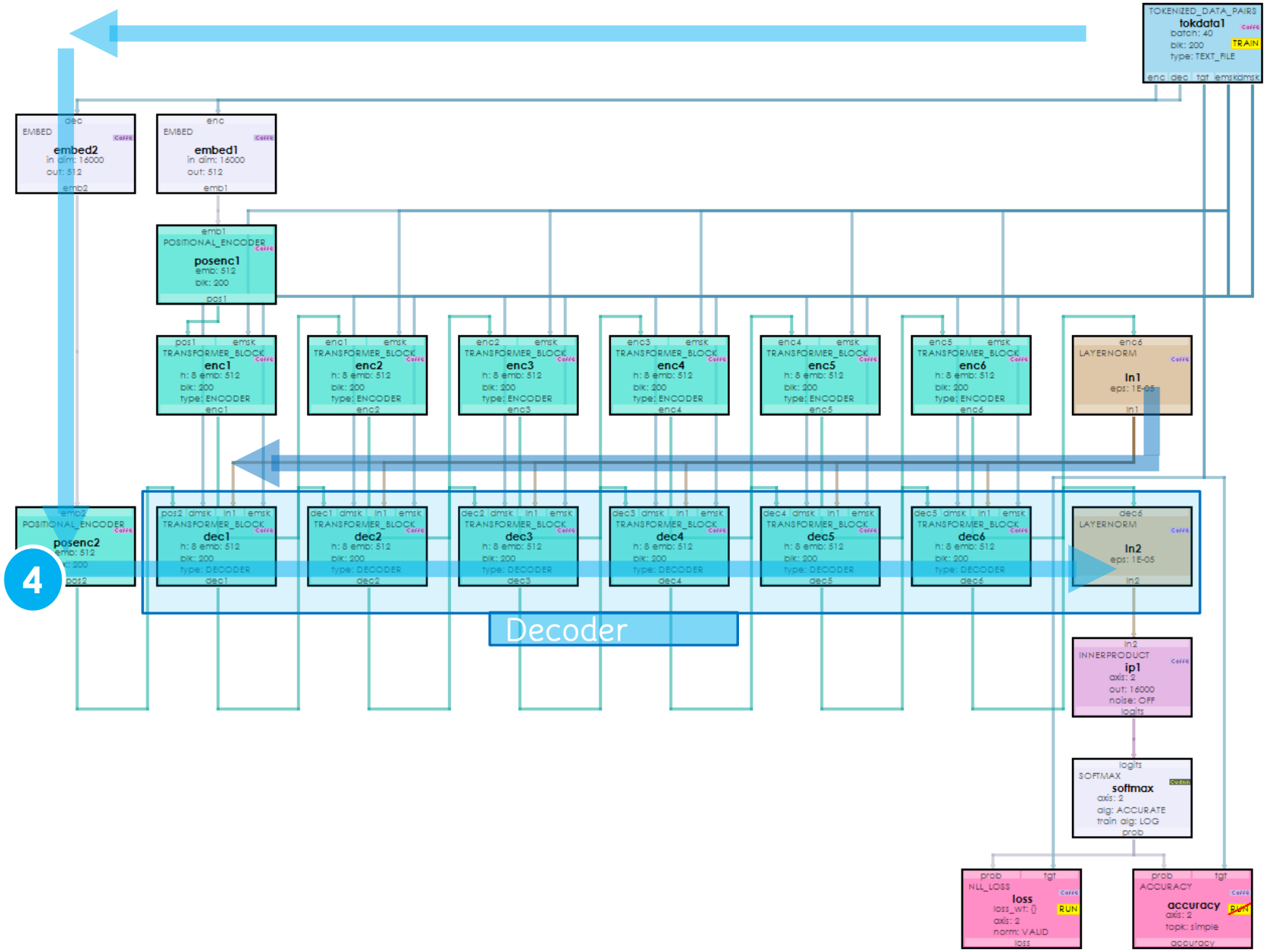
This IS EVEN MORE secret sauce!

4 decoder transformer (multi-head 2)

Decoder Transformer

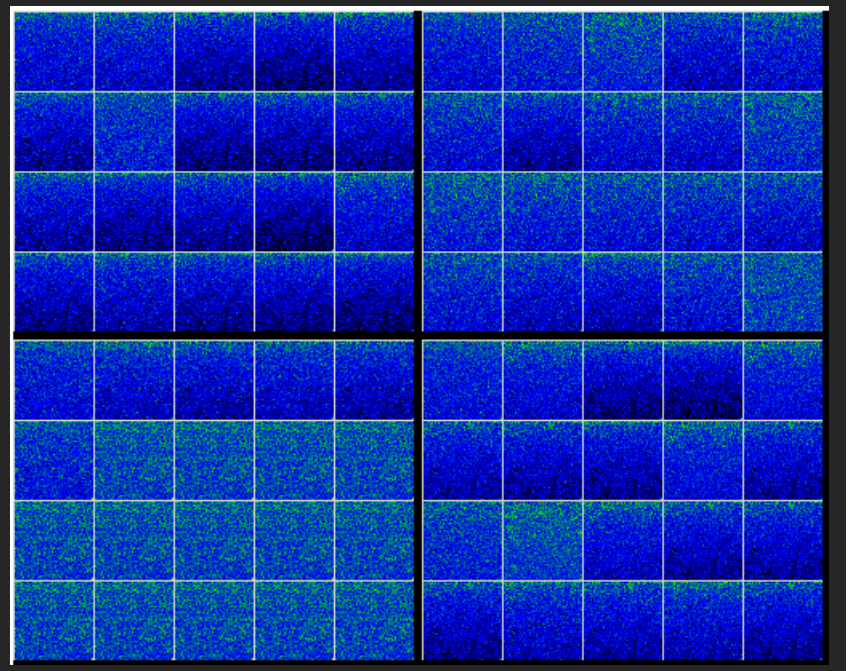
Mask, Softmax (attention),.. projection…

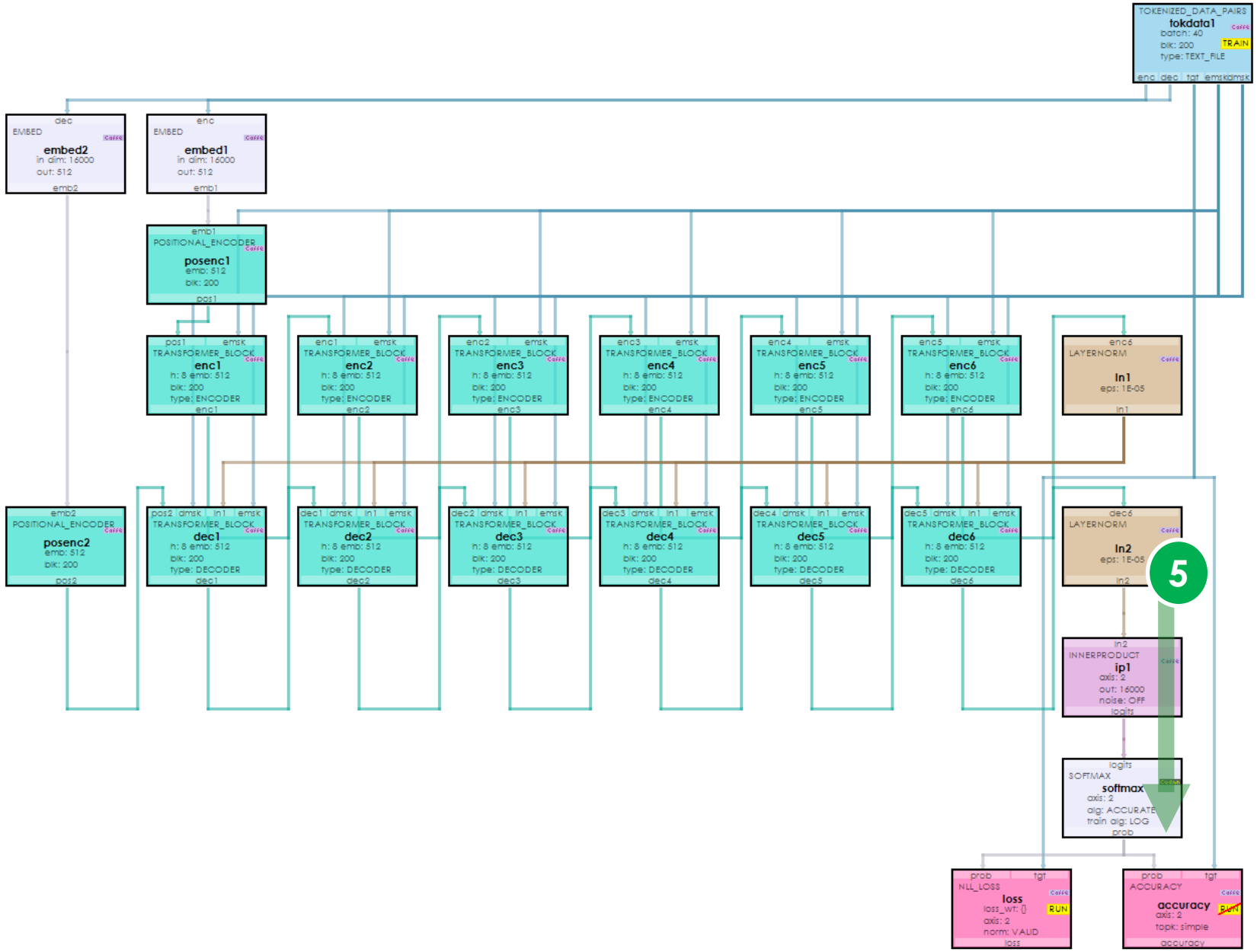




5
Logits

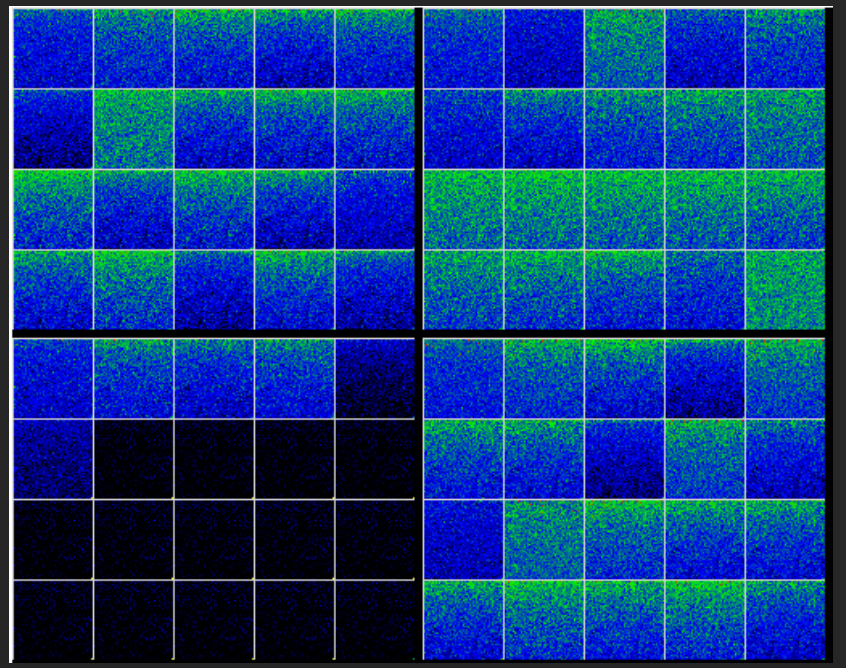
Final logits…

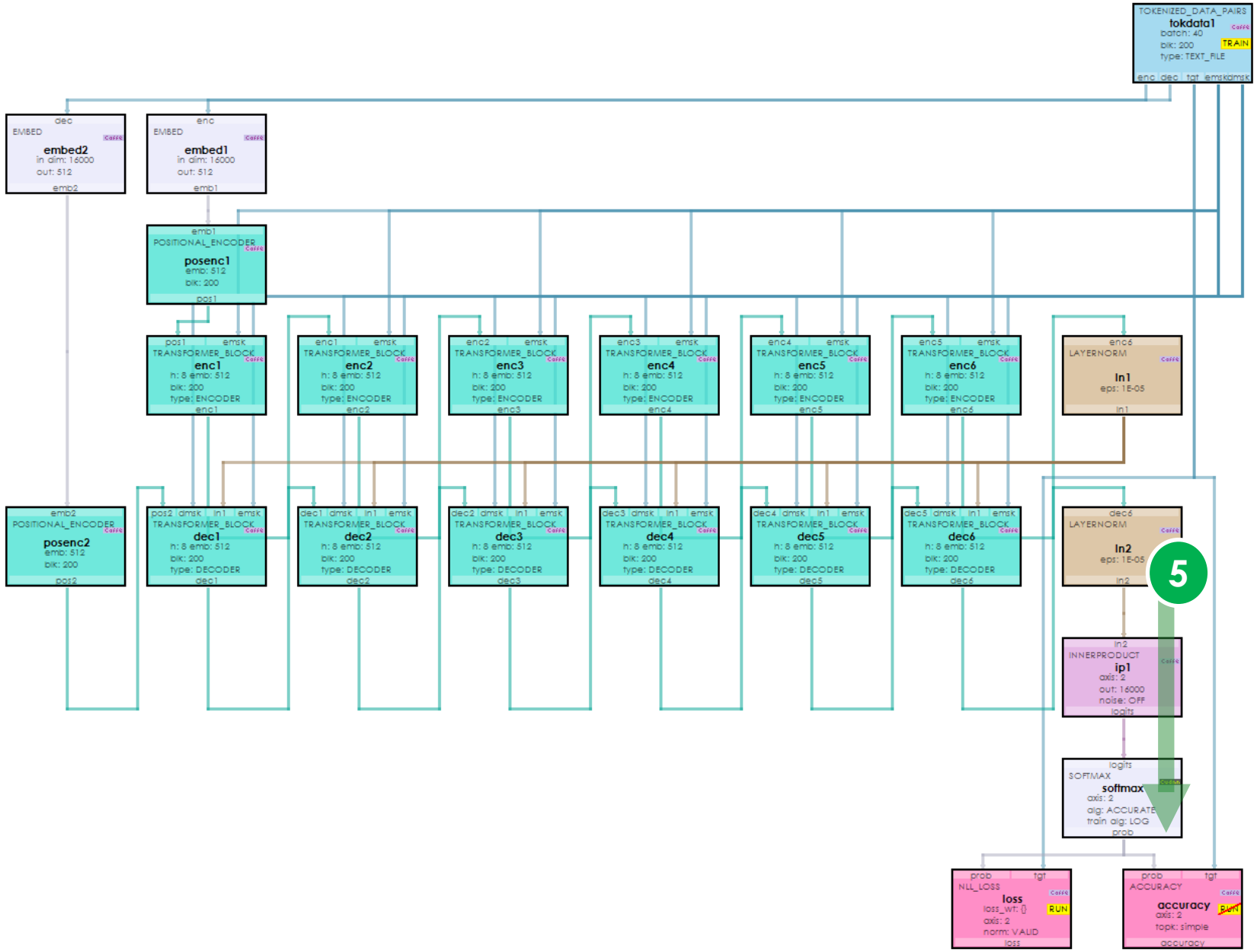




5 Probabilities

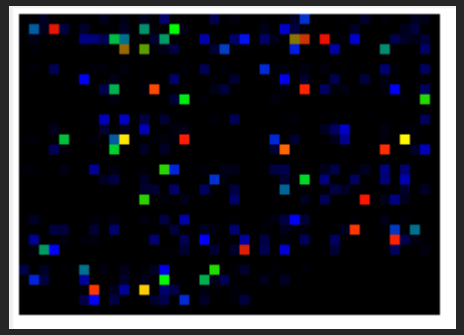
Softmax logits to get probabilities…



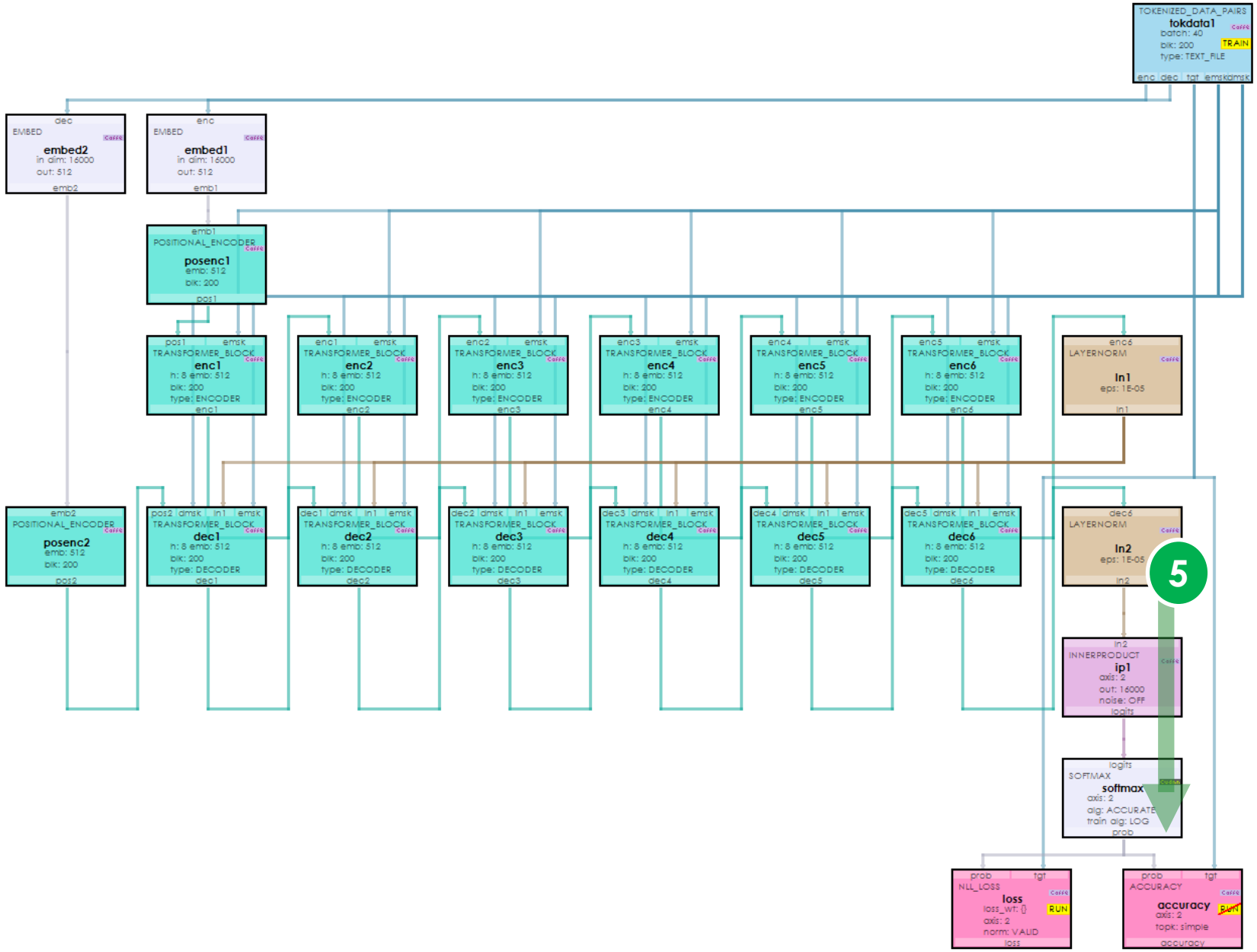


5
Targets

Decoder output (target tokens)



Quand la session de l ' Union europ?enne commence - t - elle ?



Verify Results

Extra '?' characters found in results, believed to be caused by our custom sentence-piece tokenizer and lack of double byte character handling for words like 'européenne'.

Google Output:
"Quand commence la session de l'Union européenne ?"

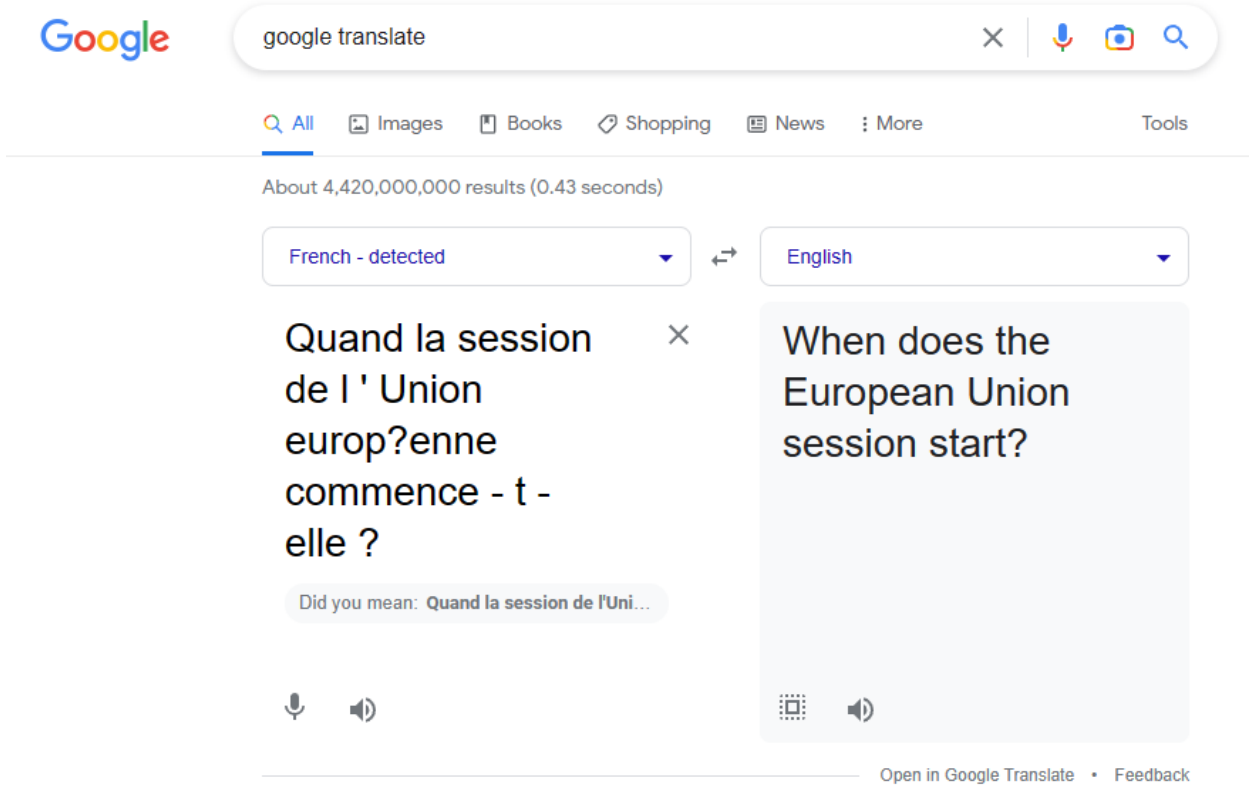
Our Input

"When does the European Union session start?"

Our Output

"Quand la session de l ' Union europ?enne commence - t - elle ?"

Google Translate Verification



Thank You!

References

Attention Is All You Need; Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin, arXiv 2017 https://arxiv.org/abs/1706.03762

Transformer's Encoder-Decoder: Let's Understand The Model Architecture; KiKaBeN, 2021; https://kikaben.com/transformers-encoder-decoder/

The Illustrated Transformer; Jay Alammar, 2020, Jay Alammar Blog, https://jalammar.github.io/illustrated-transformer/

Transformer-based Encoder/Decoder Models; Patrick von Platen, Hugging Face, 2020; https://huggingface.co/blog/encoder-decoder

GELU activation; Shaurya Goel; Medium, 2019; https://medium.com/@shauryagoel/gelu-gaussian-error-linear-unit-4ec59fb2e47c

Group Normalization; Yuxin Wu and Kaiming He; arXiv, 2018; https://arxiv.org/abs/1803.08494