Transformer

Understanding Attention

Daniel Klauser

June, 2024

HSLU Hochschule Luzern

Basics

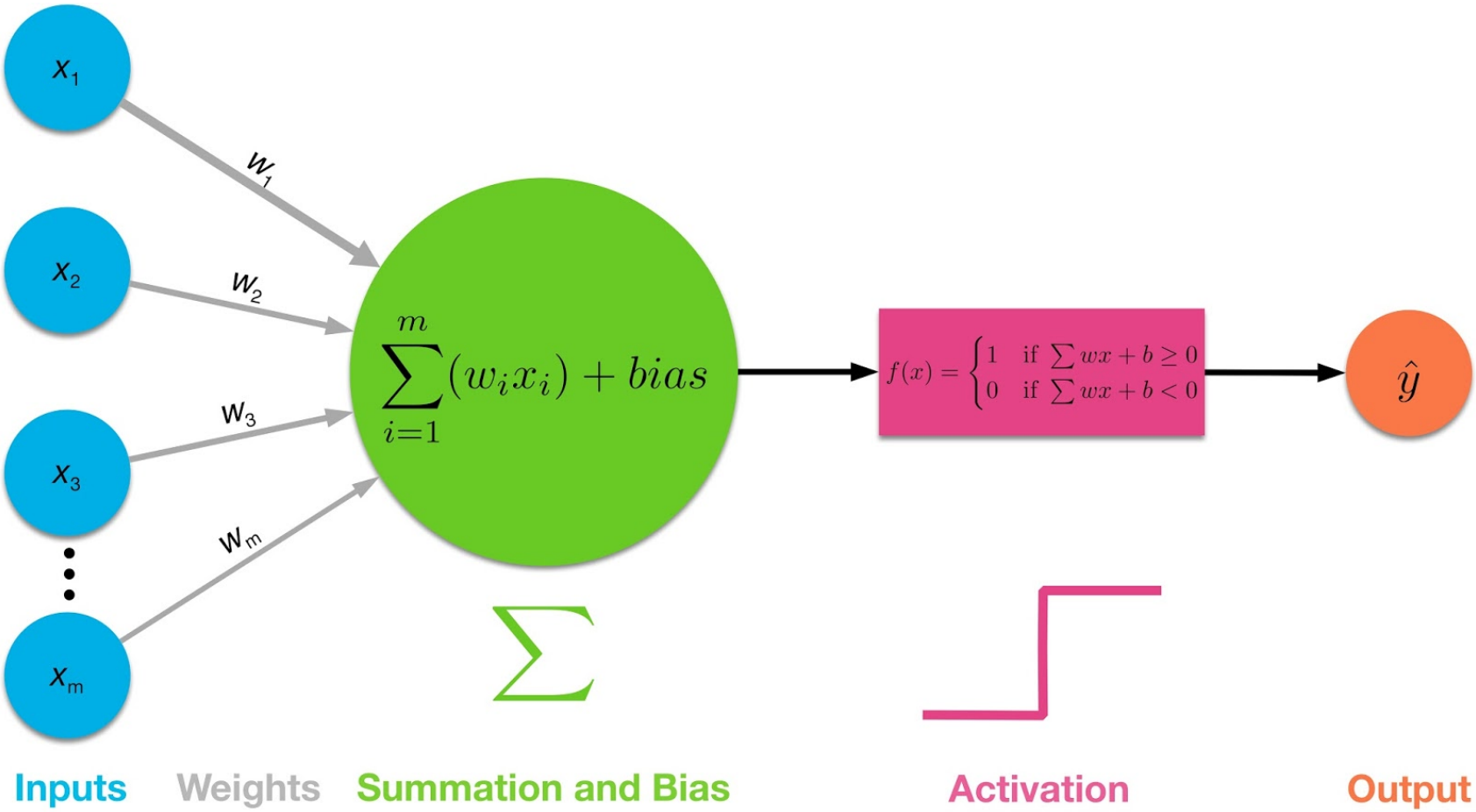


图 1: Basics from feature input to activated output.

What is trainable?

Transformer emerge from sequence task

1950: ANN

1980: CNN & RNN

1997: LSTM

2014: GRU

2017: Transformer

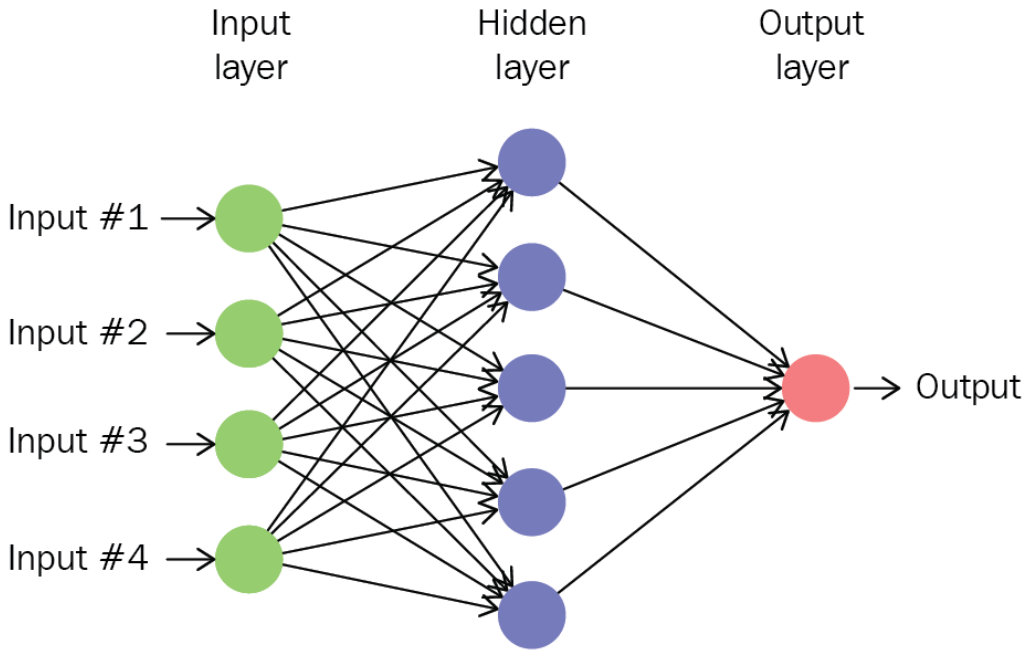


图 2: A small fully connected network with one hidden layer.

Mention universal approximation theorem

Recurrent Neuronal Network

1950: ANN

1980: CNN & RNN

1997: LSTM

2014: GRU

2017: Transformer

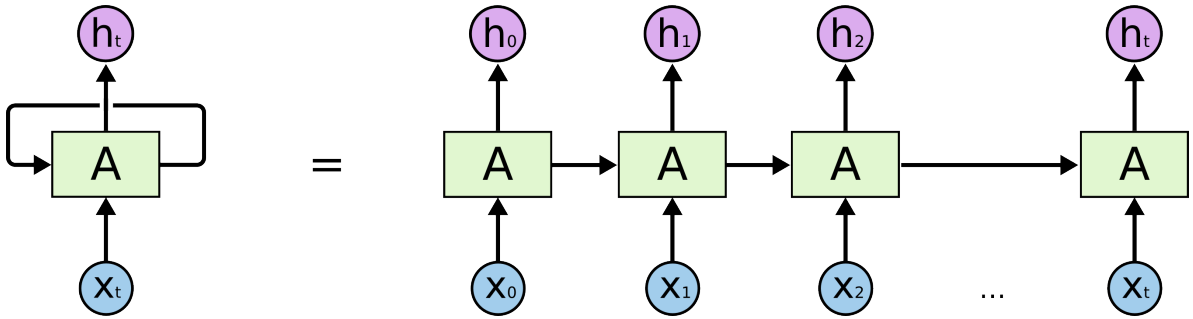


图 3: An unrolled RNN.

What is the inductive bias?

Language Translation Task

1950: ANN

1980: CNN & RNN

1997: LSTM

2014: GRU

2017: Transformer

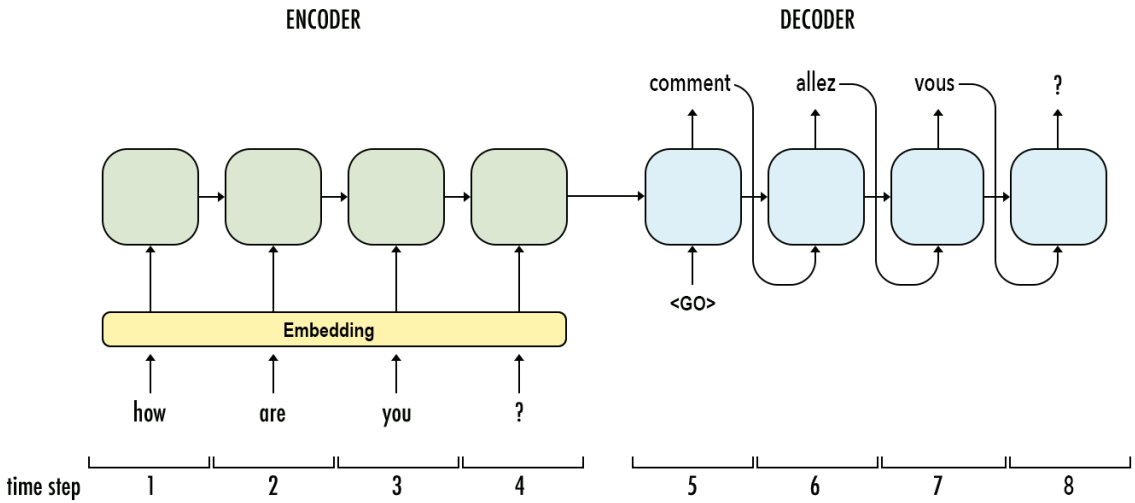


图 4: Encoder-Decoder with RNN's.

Bottleneck on long sentences.

Long Short Term Memory

1950: ANN

1980: CNN & RNN

1997: LSTM

2014: GRU

2017: Transformer

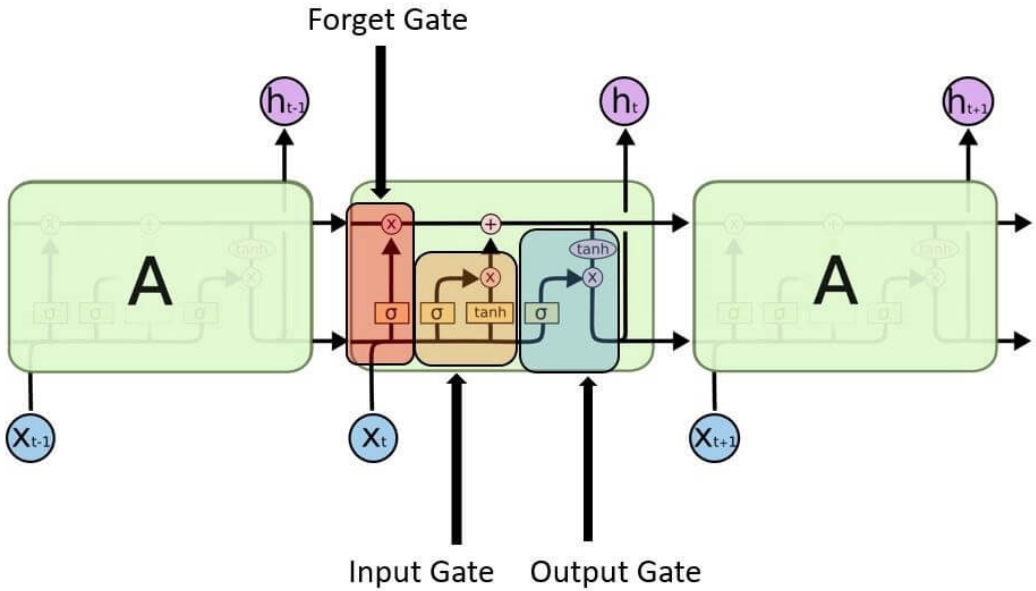


图 5: An unrolled LSTM block.

Less gradient vanishing and parallelism but more parameters and therefore storing capability.

Gated Recurrent Units

1950: ANN

1980: CNN & RNN

1997: LSTM

2014: GRU

2017: Transformer

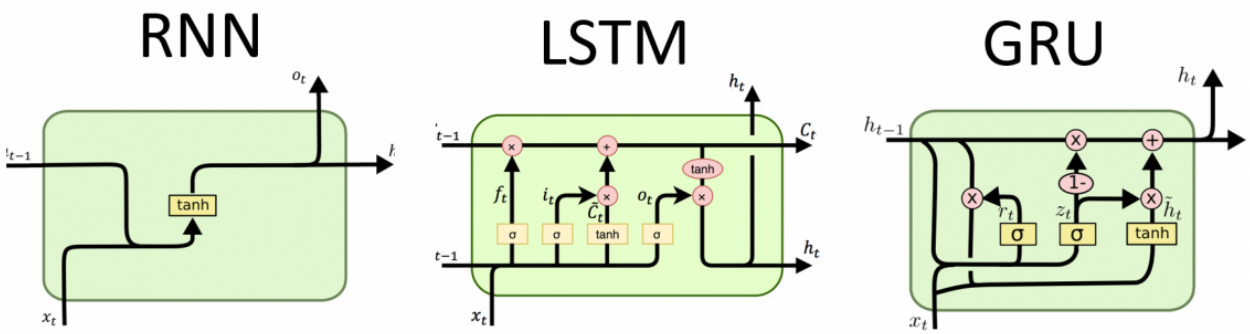


图 6: All popular building blocks.

Less Parameters, same performance as LSTM.

What else was there? Bidirectional LSTM/RNN

1950: ANN

1980: CNN & RNN

1997: LSTM

2014: GRU

2017: Transformer

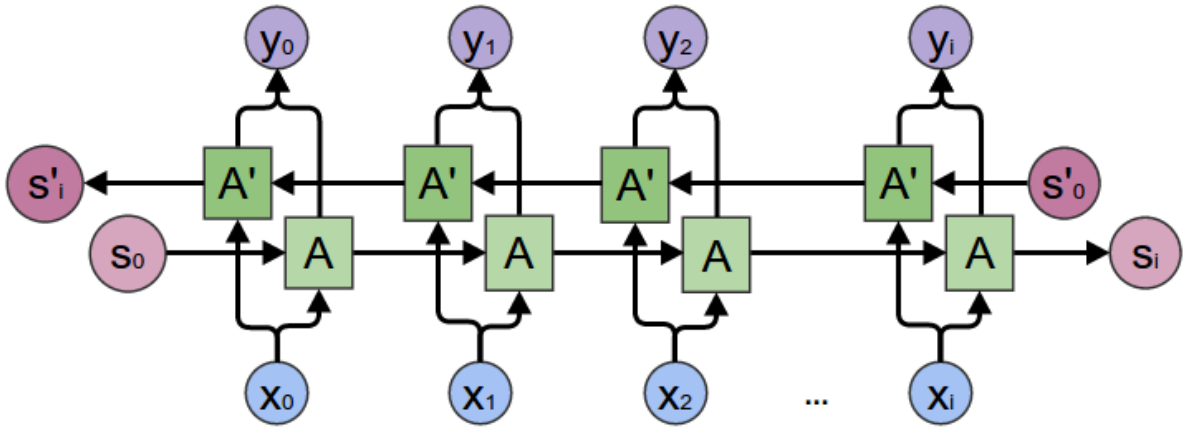


图 7: Bidirectional architecture can consists of RNN/LSTM/GRU.

To know if something is important, we need to know the context.

What else was there? Origin of attention

1950: ANN

1980: CNN & RNN

1997: LSTM

2014: GRU

2017: Transformer

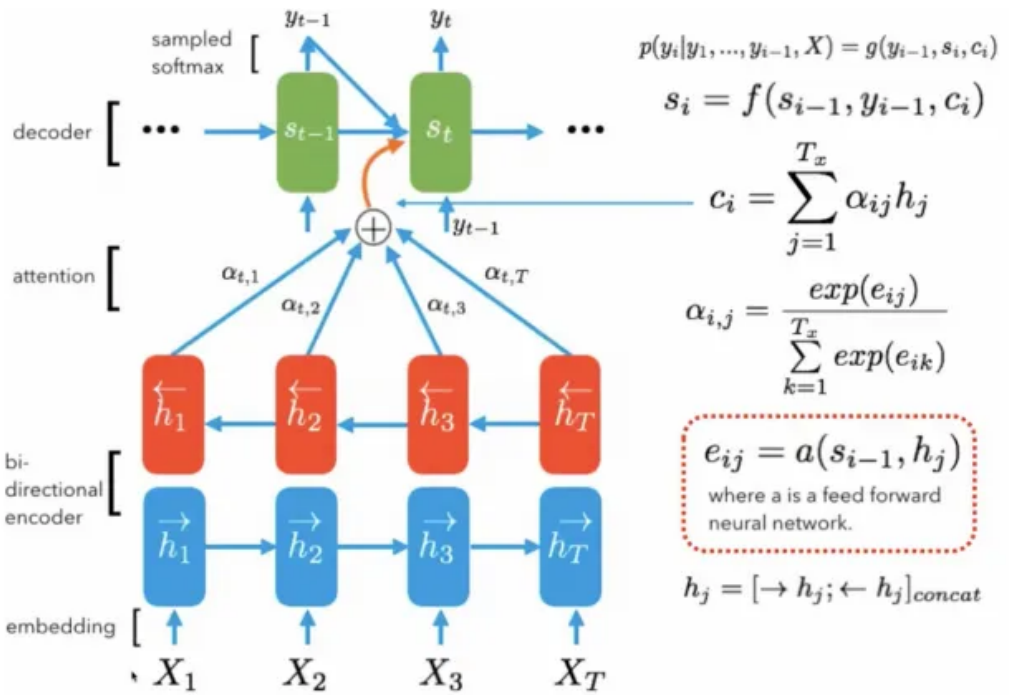


图 8: Origin of attention, from [1].

Explain the softmax.

The Transformer

1950: ANN

1980: CNN & RNN

1997: LSTM

2014: GRU

2017: Transformer

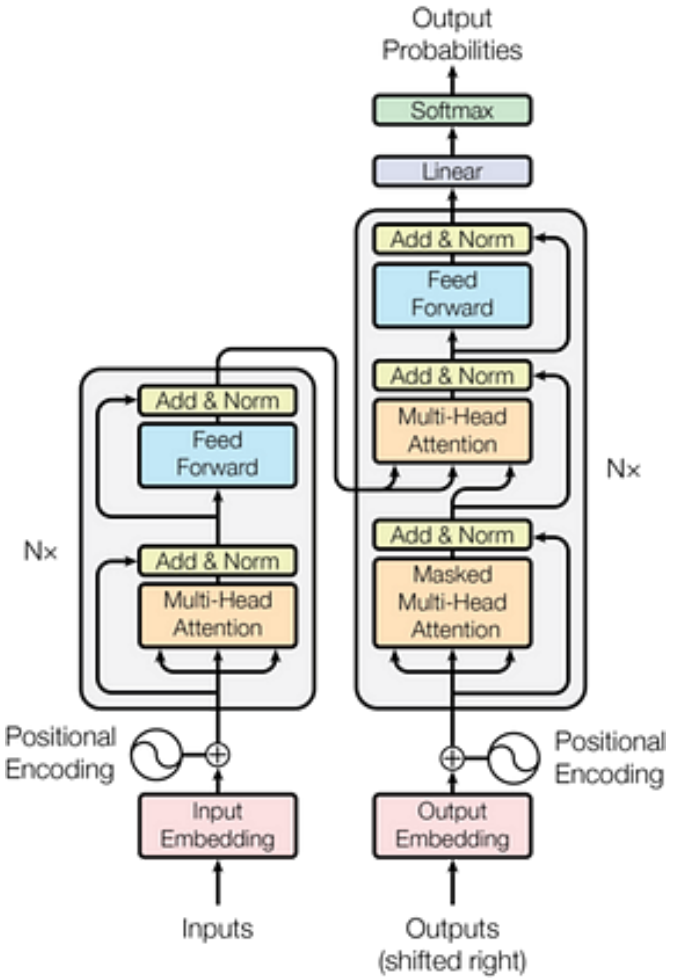


图 9: The transformer.

Building Blocks

- Embeddings
- Self Attention
- Multi-Head Self Attention
- Layer Normalization
- Positional Encoding
- Activation Function
- Explainability

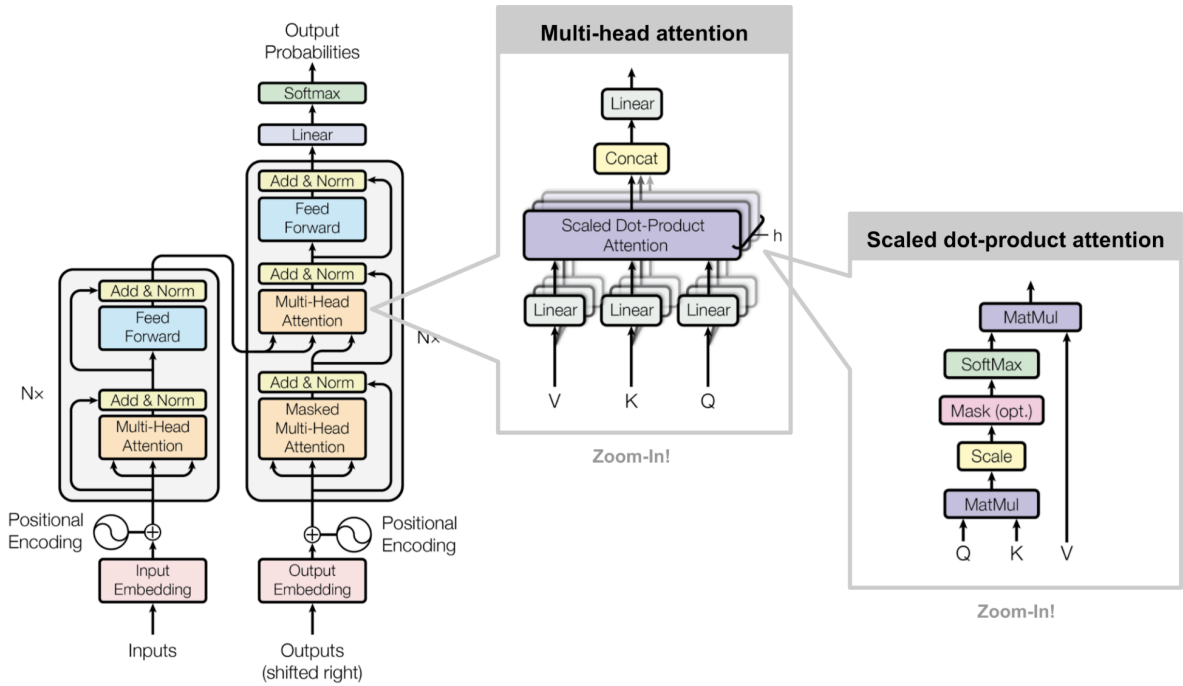


图 10: A transformer, with focus on attention.

Embeddings

- Embeddings
- Self Attention
- Multi-Head Self Attention
- Layer Normalization
- Positional Encoding
- Activation Function
- Explainability

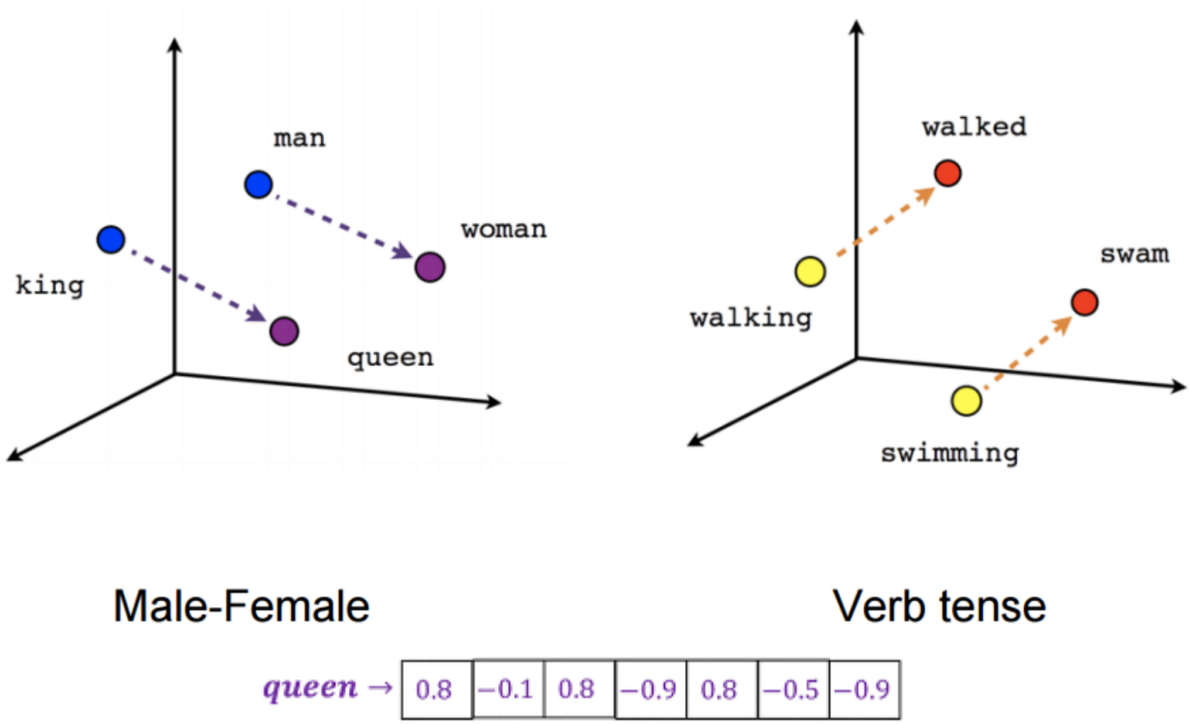


图 11: The embedding feature space.

King –Man + Woman = Queen.

Word Pulling

- Embeddings
- Self Attention
- Multi-Head Self Attention
- Layer Normalization
- Positional Encoding
- Activation Function
- Explainability

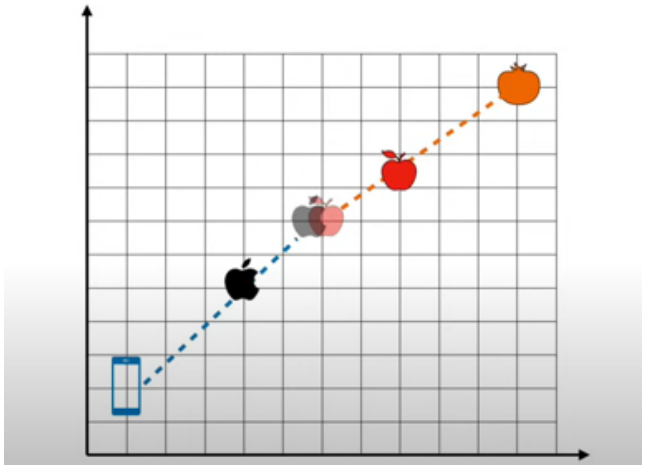


图 12: Through attention, the meaning of the word is pulled in the "right" direction of its context.

Keep the example in mind.

Concept of similarity in a vector space

- Embeddings
- Self Attention
- Multi-Head Self Attention
- Layer Normalization
- Positional Encoding
- Activation Function
- Explainability

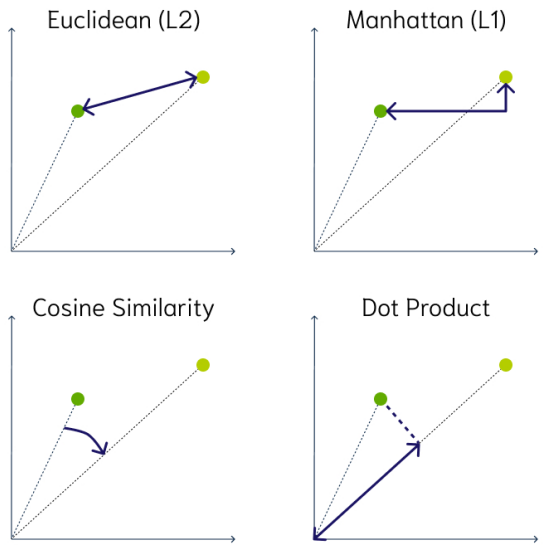


图 13: Similarity metrics

- *in high dimensional spaces, one prefer manhatten over euclidean.*
- *if you normalize your data, cosine similarity and dot product are indistinguishable. But the dot product is cheaper to calculate.*

Key, Query & Value

- Embeddings
- Self Attention
- Multi-Head Self Attention
- Layer Normalization
- Positional Encoding
- Activation Function
- Explainability

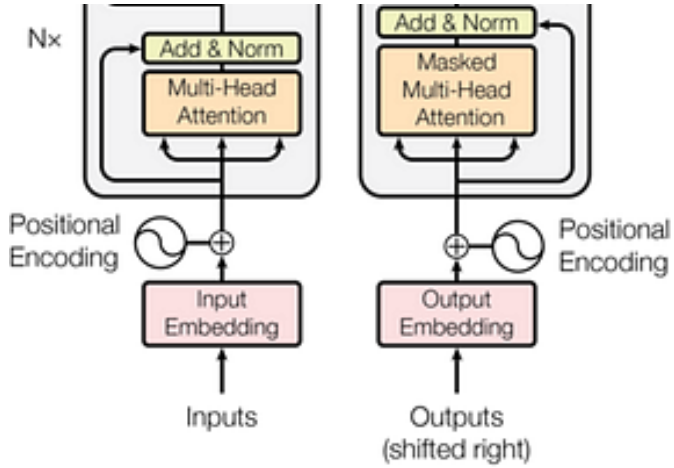


图 14: Pay attention to the three inputs of the multi-head attention block.

Key, Query & Value

- Embeddings
- Self Attention
- Multi-Head Self Attention
- Layer Normalization
- Positional Encoding
- Activation Function
- Explainability

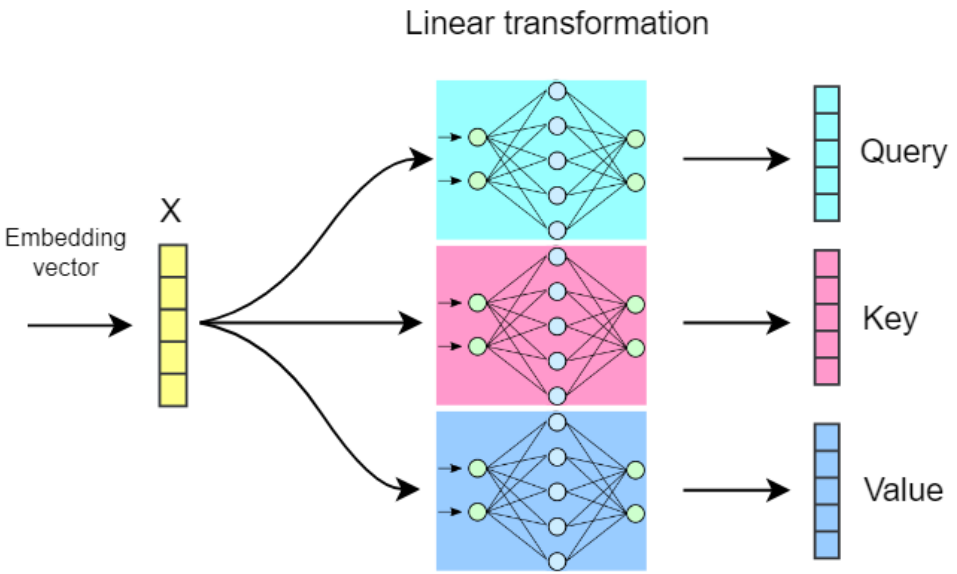


图 15: Showing the linear projection/transformation of the input vector to Q,K,V.

The dimension are often the same.

Key, Query & Value

- Embeddings
- Self Attention
- Multi-Head Self Attention
- Layer Normalization
- Positional Encoding
- Activation Function
- Explainability

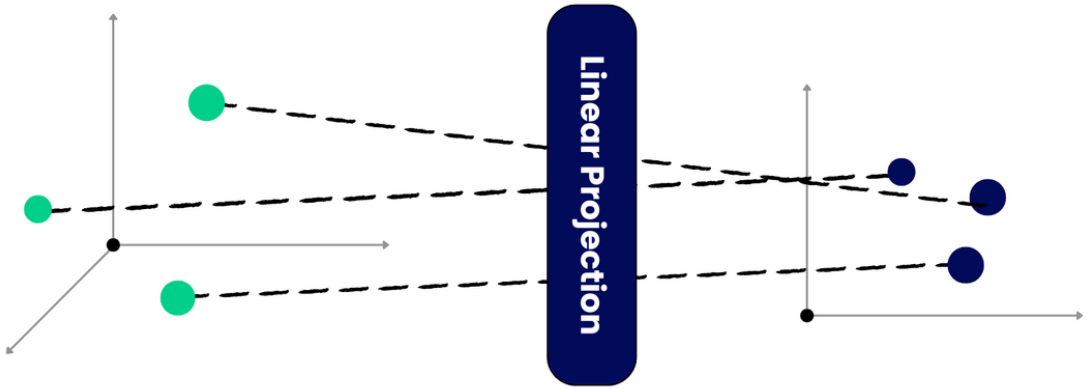


图 16: Showing a linear projection from high into lower dimensional space.

Self Attention Operation

- Embeddings
- Self Attention
- Multi-Head Self Attention
- Layer Normalization
- Positional Encoding
- Activation Function
- Explainability

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \quad (1)$$

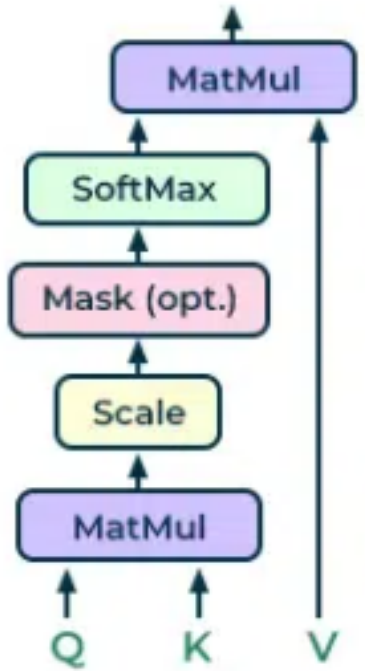


图 17: Self attention workflow.

Query Example

- Embeddings
- Self Attention
- Multi-Head Self Attention
- Layer Normalization
- Positional Encoding
- Activation Function
- Explainability

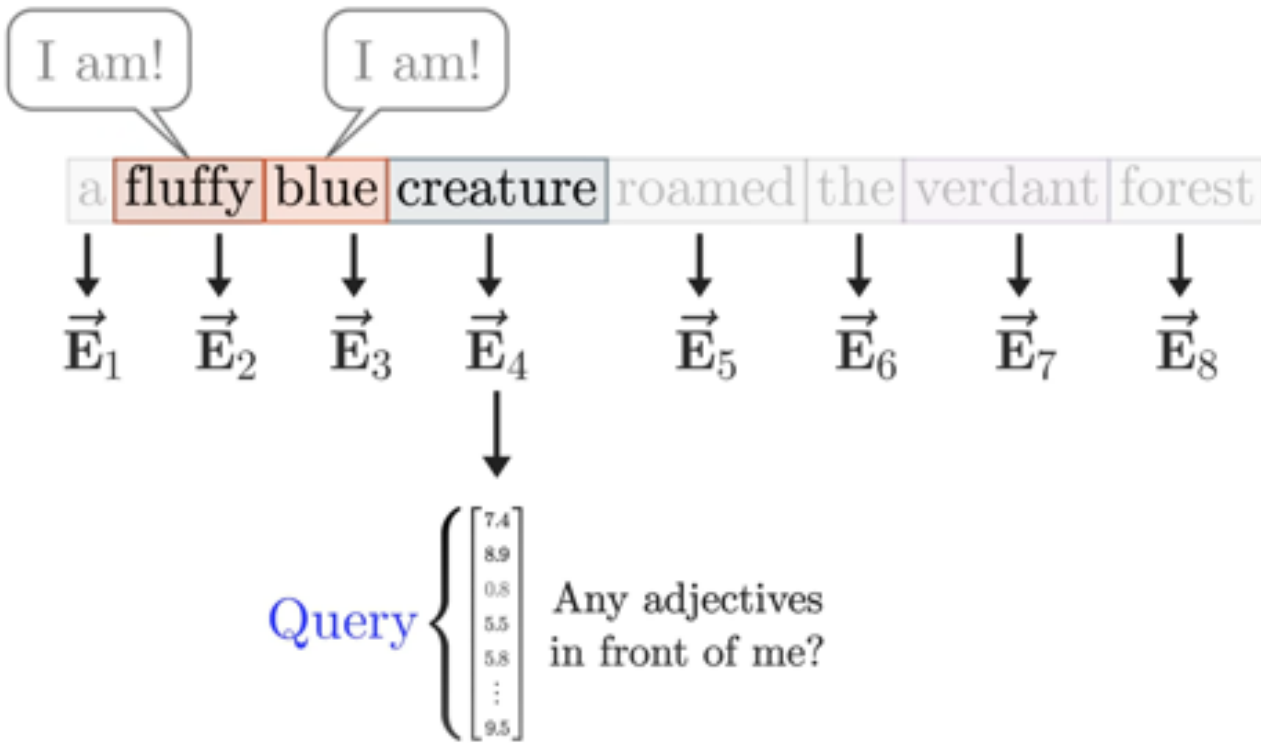


图 18: Text example, from [2]

The dot product of QE4xKE3.T would be high.

Key & Query Example (ViT)

- Embeddings
- Self Attention
- Multi-Head Self Attention
- Layer Normalization
- Positional Encoding
- Activation Function
- Explainability

Query : What is on the image (positive are white)

Key : looks like an airplane

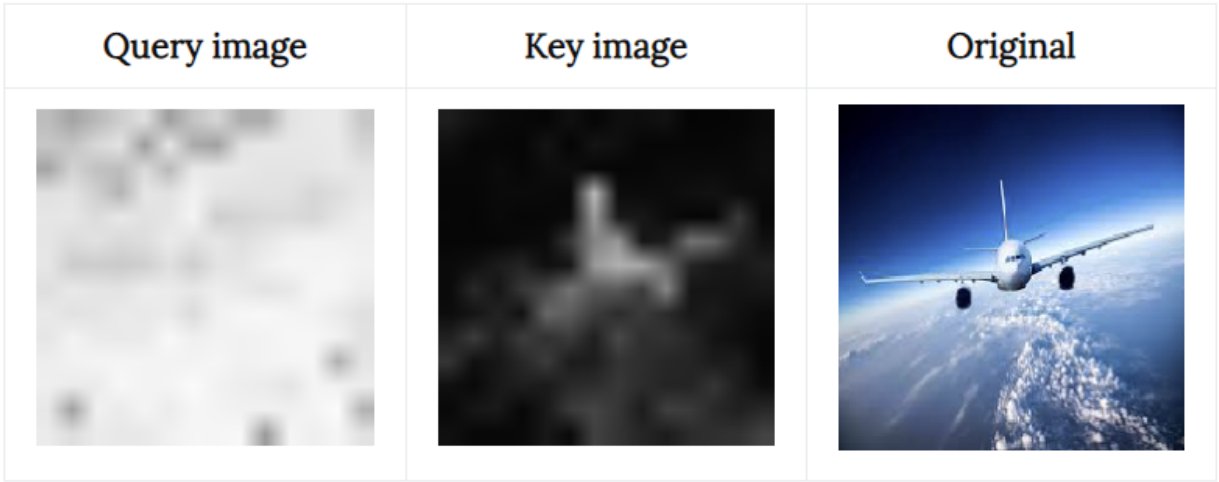


图 19: Image example, from [3]

Self Attention Operation Meaning

- Embeddings
- Self Attention
- Multi-Head Self Attention
- Layer Normalization
- Positional Encoding
- Encoder & Decoder
- Explainability

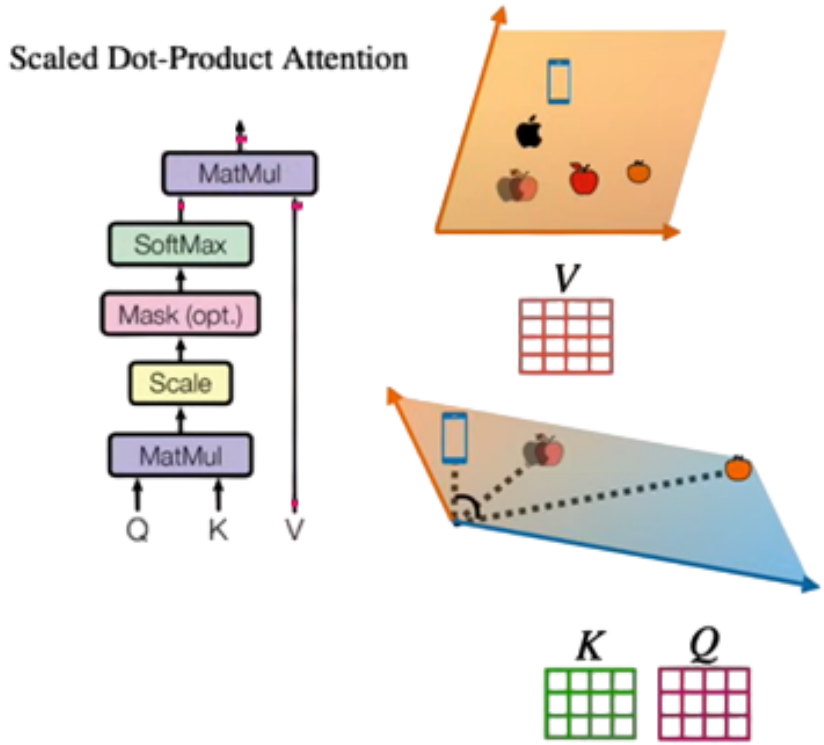


图 20: Applied word pulling, from [4]

The result of QxK.T gives us the scaled direction to move V.

Multi Head Self Attention

- Embeddings
- Self Attention
- Multi-Head Self Attention
- Layer Normalization
- Positional Encoding
- Activation Function
- Explainability

$$MultiHead(Q, K, V) = Concat(head_1, ..., head_h)W^O \quad (2)$$

$$head_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \quad (3)$$

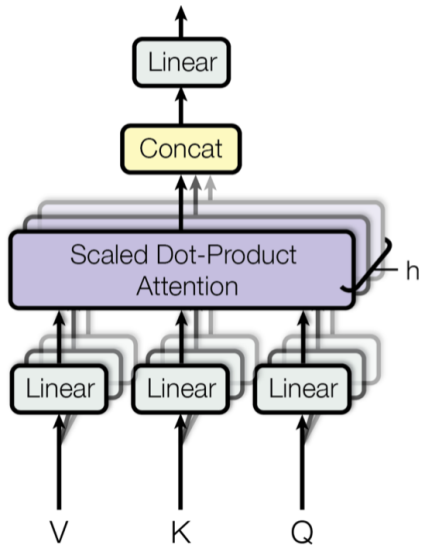
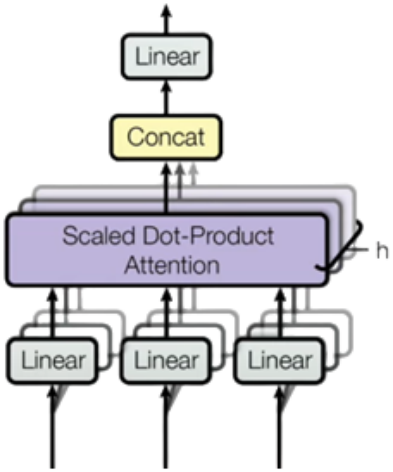


图 21: Remember, convolution operation also dont have just one filter

Mention parallelism.

Multi Head Self Attention Combined

- Embeddings
- Self Attention
- Multi-Head Self Attention
- Layer Normalization
- Positional Encoding
- Activation Function
- Explainability



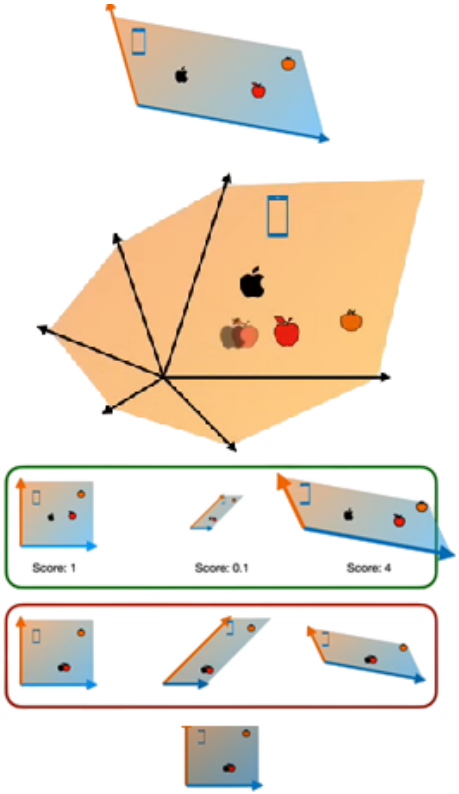
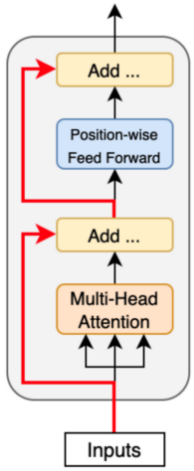


图 22: Pay attention to the concatentation of the different transformation, followed by its projection in the lower feature space, from [4]

mention the weighting from the linear layer.

Residual Connection

- Embeddings
- Self Attention
- Multi-Head Self Attention
- Layer Normalization
- Positional Encoding
- Activation Function
- Explainability



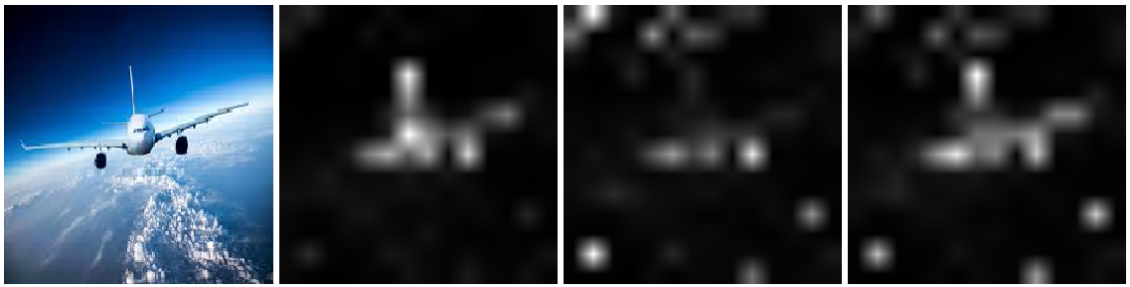
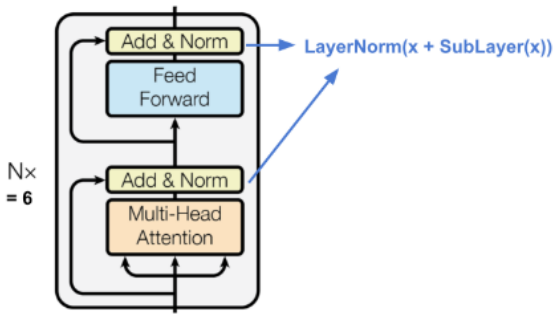


图 23: What would happen without residual connections?

Layer Norm

- Embeddings
- Self Attention
- Multi-Head Self Attention
- Layer Normalization
- Positional Encoding
- Activation Function
- Explainability



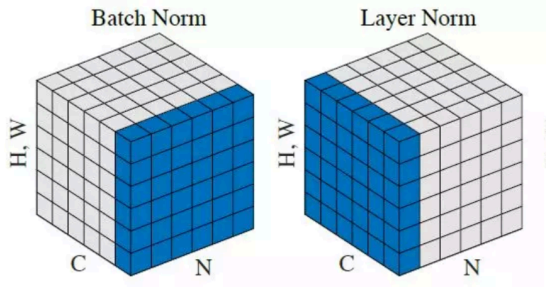


图 24: Transformer use layer norm, CNN's up to now, batch norm. Quote below from [5]

"With all the modifications in network architecture and training techniques, here we revisit the impact of using LN in place of BN. We observe that our ConvNet model does not have any difficulties training with LN; in fact, the performance is slightly better.."

Positional Encoding

- Embeddings
- Self Attention
- Multi-Head Self Attention
- Layer Normalization
- Positional Encoding
- Encoder & Decoder
- Explainability

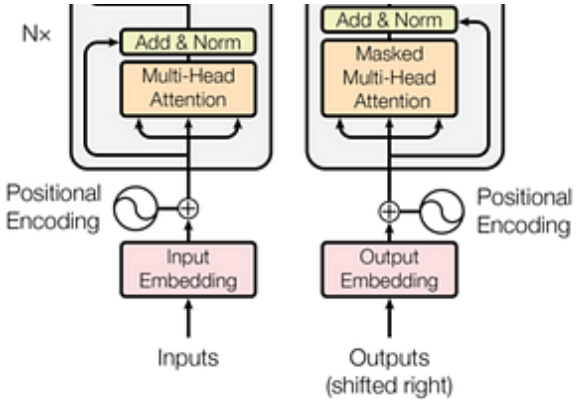


图 25: Pay attention how the position encoding is beeing included. How else would the transformer know position?

Positional Encoding

- Embeddings
- Self Attention
- Multi-Head Self Attention
- Layer Normalization
- Positional Encoding
- Encoder & Decoder
- Explainability

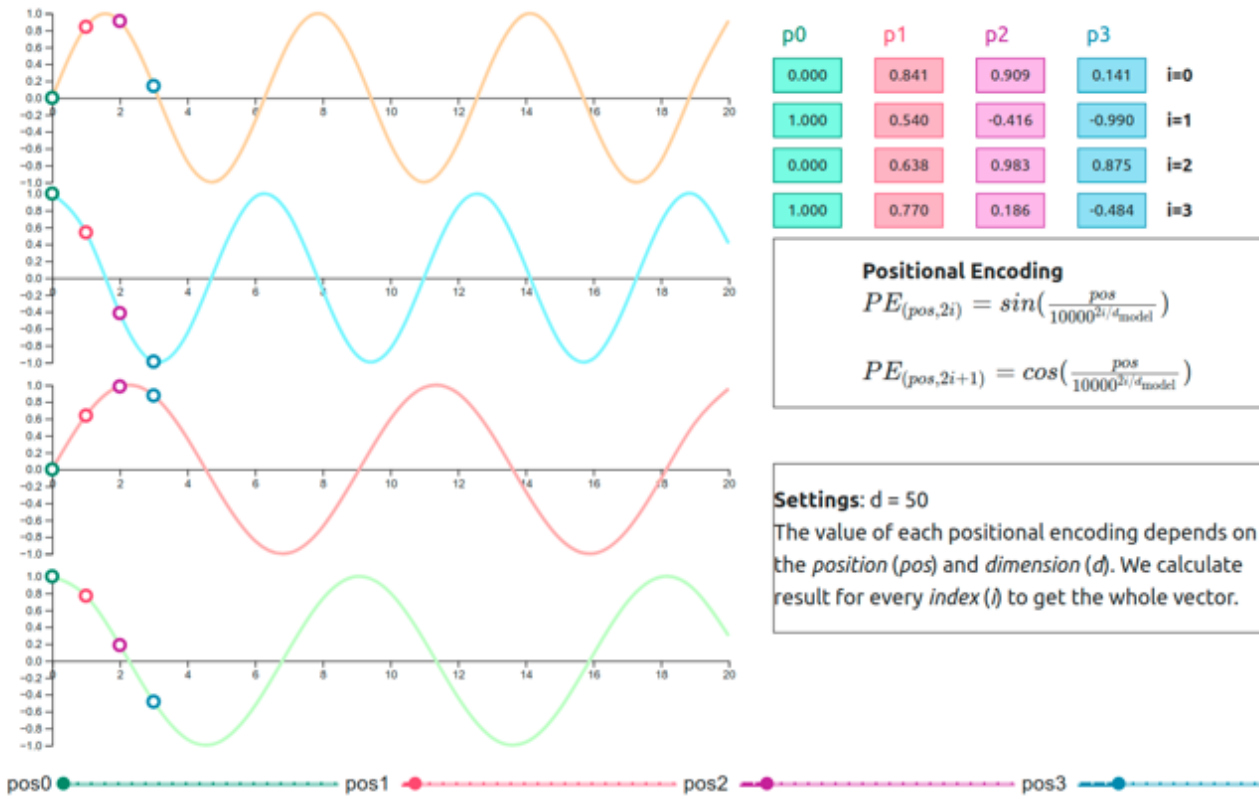


图 26: Original position encoding for GPT's.

Positional Encoding

- Embeddings
- Self Attention
- Multi-Head Self Attention
- Layer Normalization
- Positional Encoding
- Encoder & Decoder
- Explainability

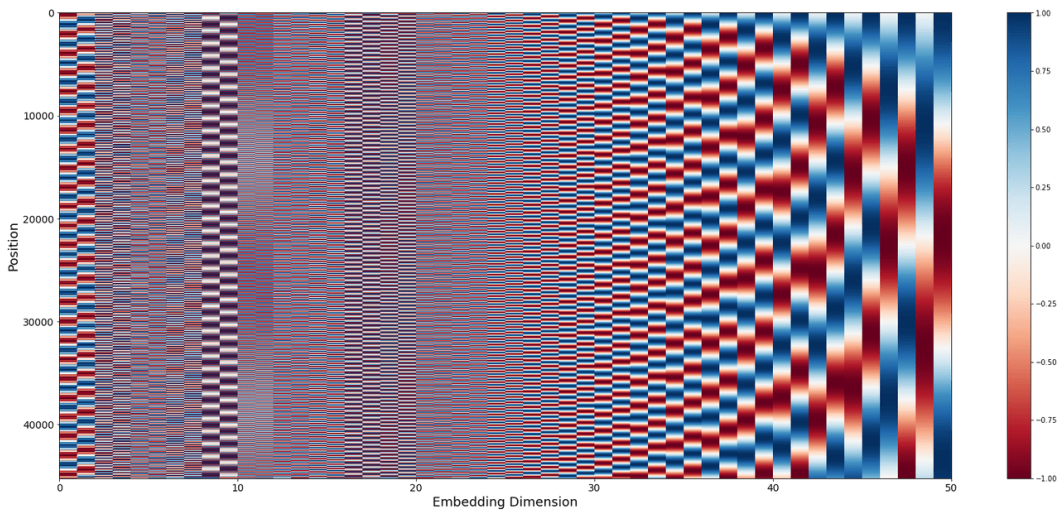


图 27: Visualized position encoding for GPT's.

Positional Encoding Vision Transformer (ViT)

- Embeddings
- Self Attention
- Multi-Head Self Attention
- Layer Normalization
- Positional Encoding
- Activation Function
- Explainability

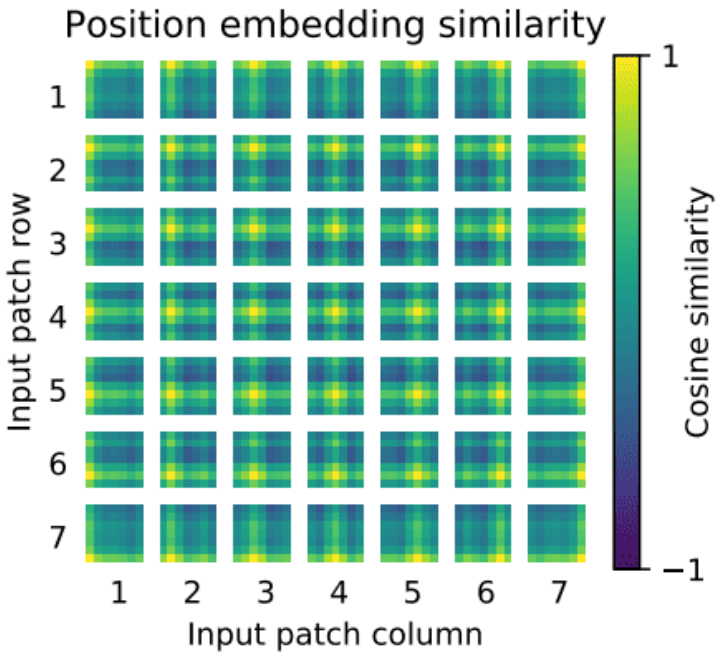


图 28: Shows the similarity of position encoding patchwise, from [6]. Not the same principle as with sin & cos pos encoding!

Activation Function

- Embeddings
- Self Attention
- Multi-Head Self Attention
- Layer Normalization
- Positional Encoding
- Activation Function
- Explainability

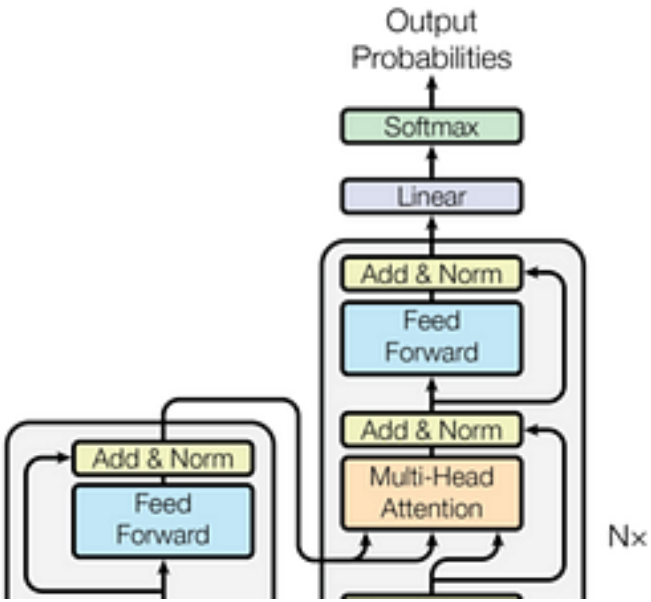


图 29: The activation is used in the feed forward part, between two fully connected layers.

Activation Function

- Embeddings
- Self Attention
- Multi-Head Self Attention
- Layer Normalization
- Positional Encoding
- Activation Function
- Explainability

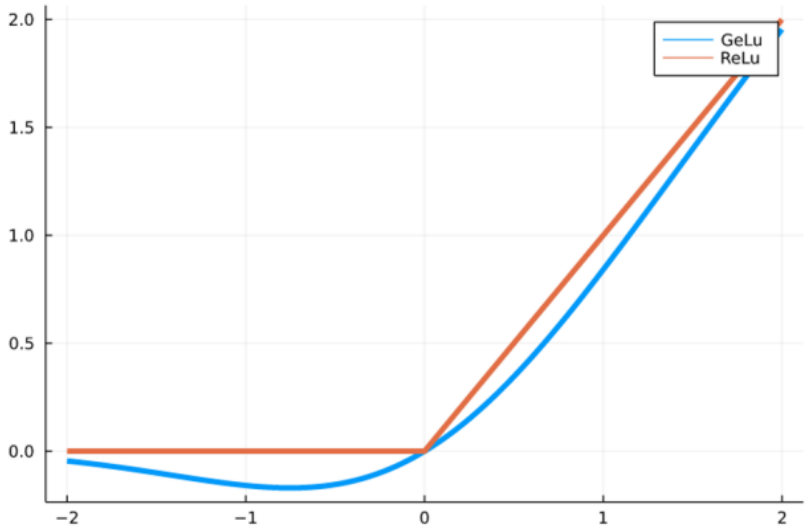


图 30: Comparison of the ReLu and GeLu activation functions. ReLu is simpler to compute, but GeLu avoids overfitting, as it is able to work better with negative-valued neurons.

Attention Matrix (ViT)

- Embeddings
- Self Attention
- Multi-Head Self Attention
- Layer Normalization
- Positional Encoding
- Activation Function
- Explainability

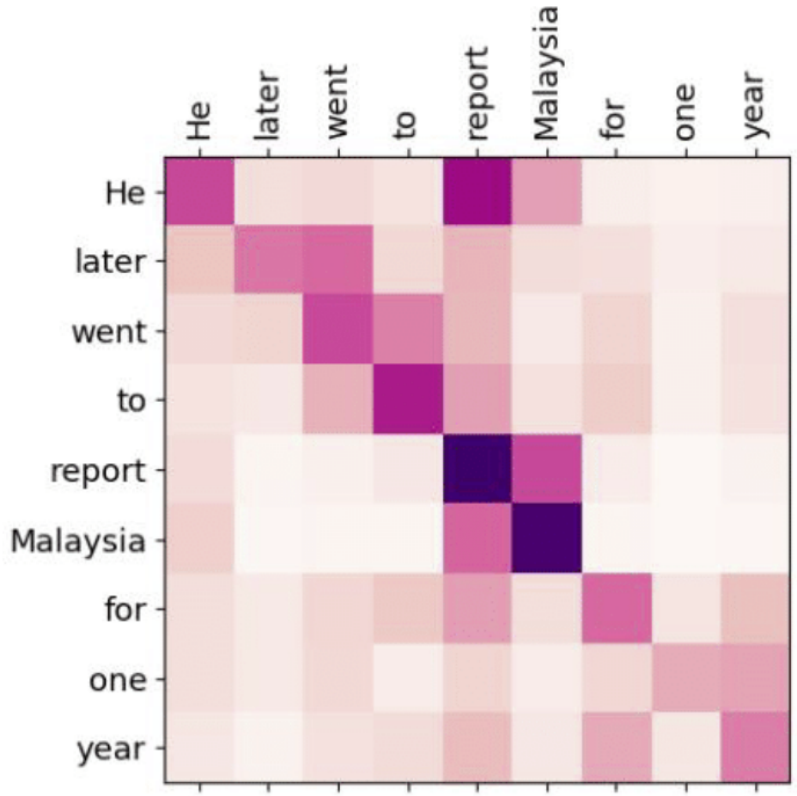


图 31: Attentionmatrix for one encoder in NLP.

Attention Rollout (ViT)

- Embeddings
- Self Attention
- Multi-Head Self Attention
- Layer Normalization
- Positional Encoding
- Activation Function
- Explainability

$$AttentionMatrix_L = (A_L + I) \quad (4)$$

$$AttentionRollout_L = (AttentionMatrix_L) * AttentionRollout_{L-1} \quad (5)$$

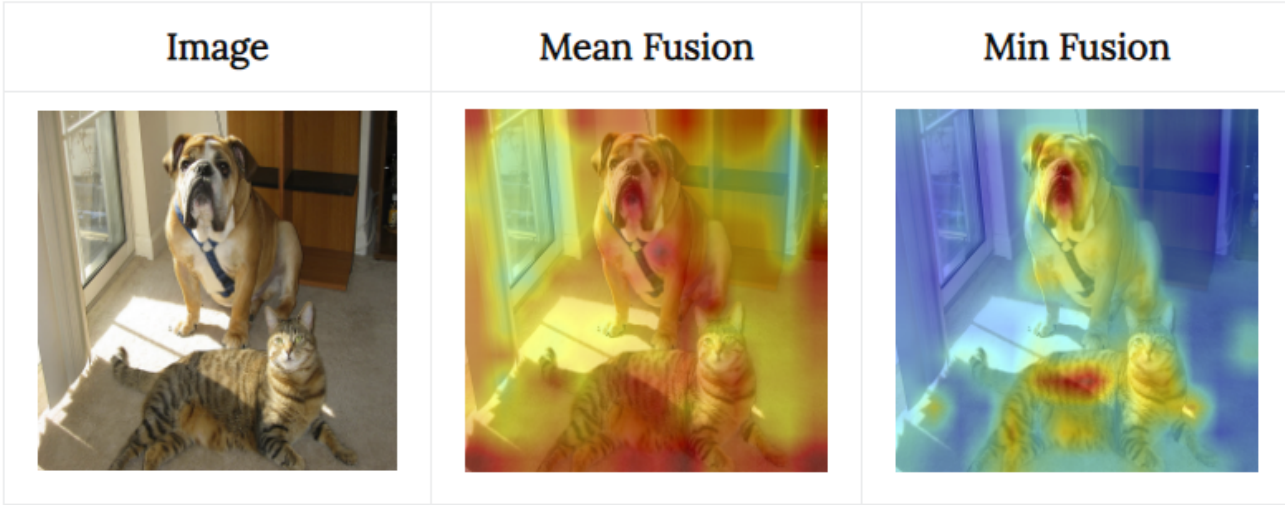


图 32: Different schemas to visualize attention rollout, by mean or min values of the heads.

AttentionMatrix = AlignmentMatrix -> after softmax, before dot product with value

Gradient Attention Rollout (ViT)

- Embeddings
- Self Attention
- Multi-Head Self Attention
- Layer Normalization
- Positional Encoding
- Activation Function
- Explainability

$$GradientAttentionRollout = A_{ij} * grad_{ij} \quad (6)$$

图 33: Gradient per Class visualized.

Not implemented yet.

Transformer Variations

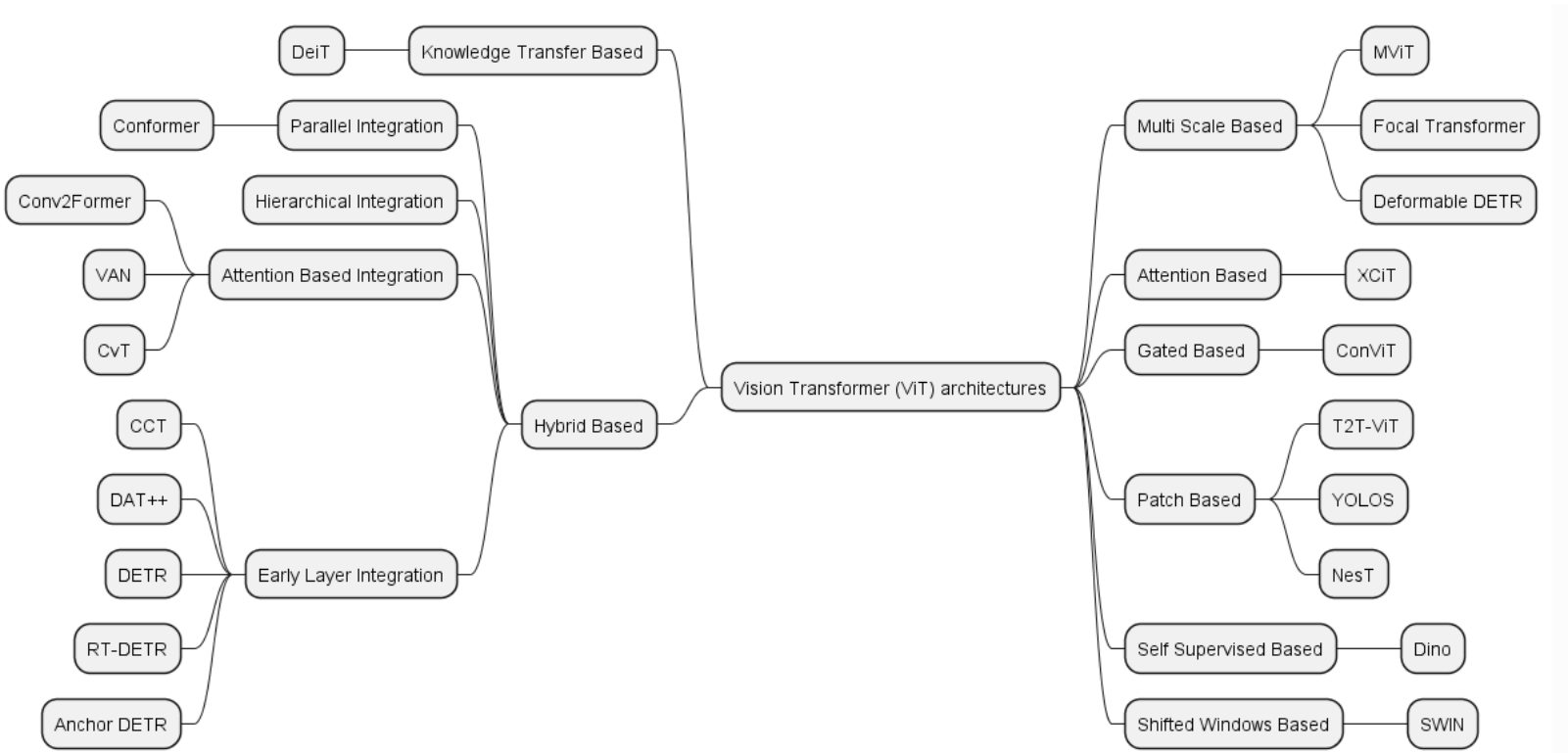


图 34: The transformer architecture is not final, many weak points.

Mention patching & trainings time & data

Attention Variants

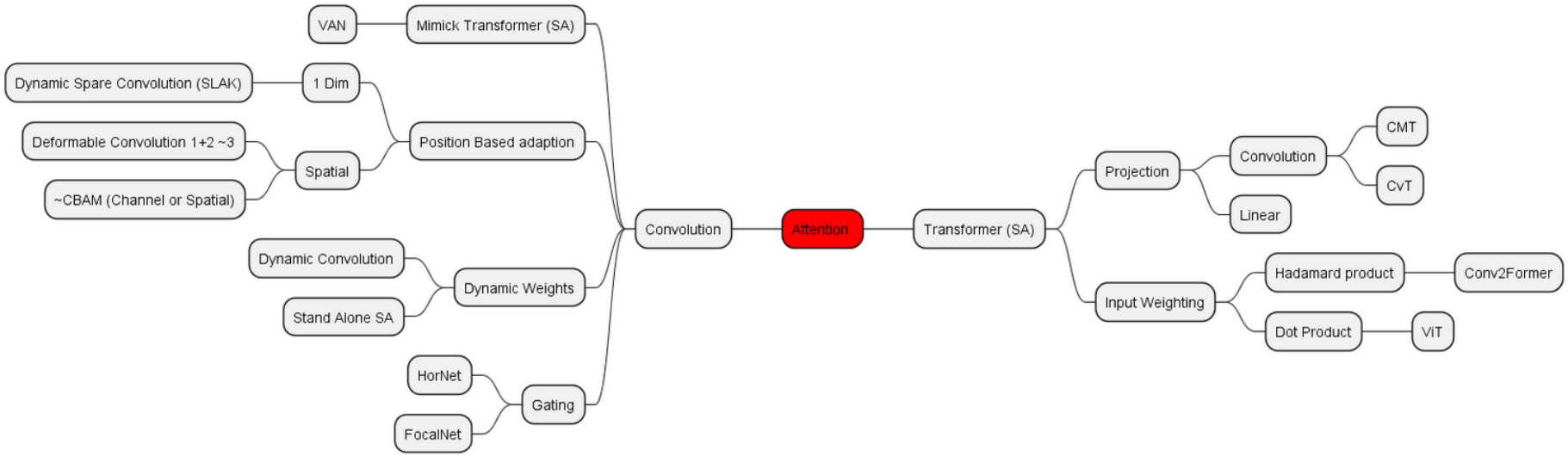


图 35: There are multiple forms of attention, also in convolution operations.

convolution operation are masters of pattern recognition but shape is often preferred.

Vision Transformer (ViT)

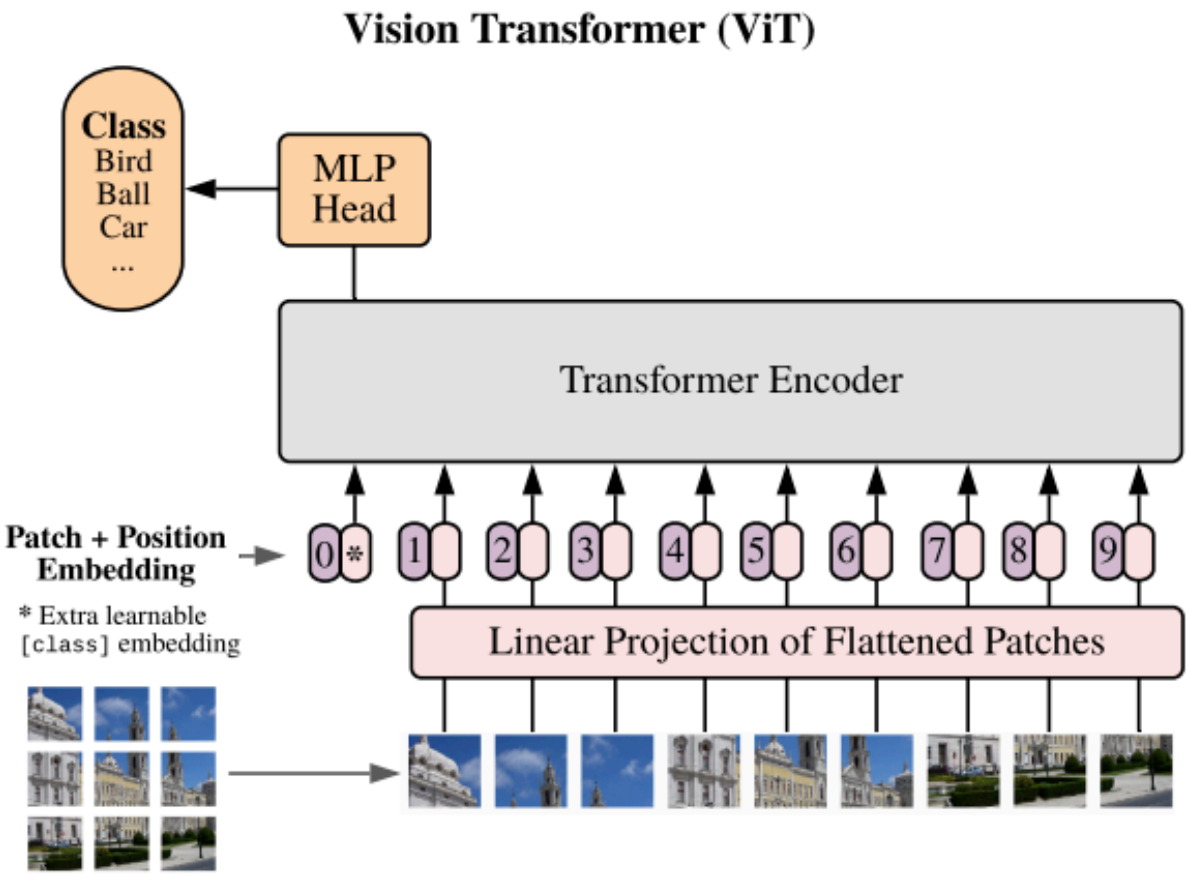


图 36: ViT Workflow, from [6].

ViT Image Patching

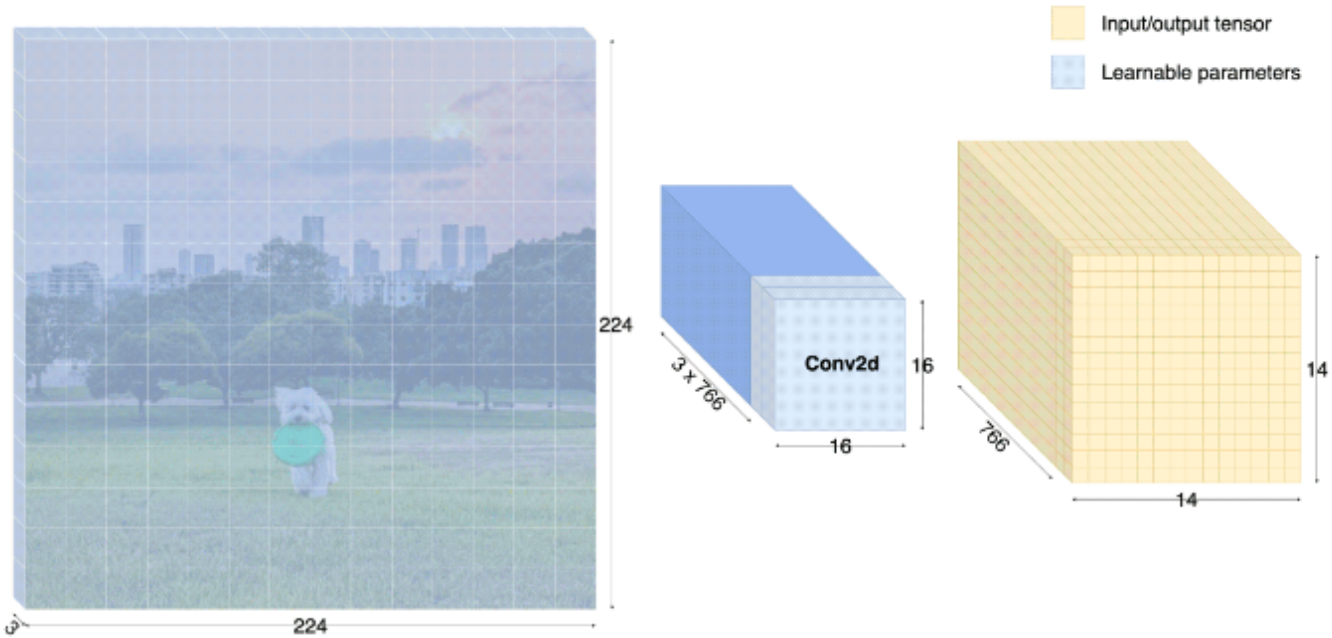


图 37: Instead of fully connected projection, we use a convolution to "patchify" the image.

ViT Class Token

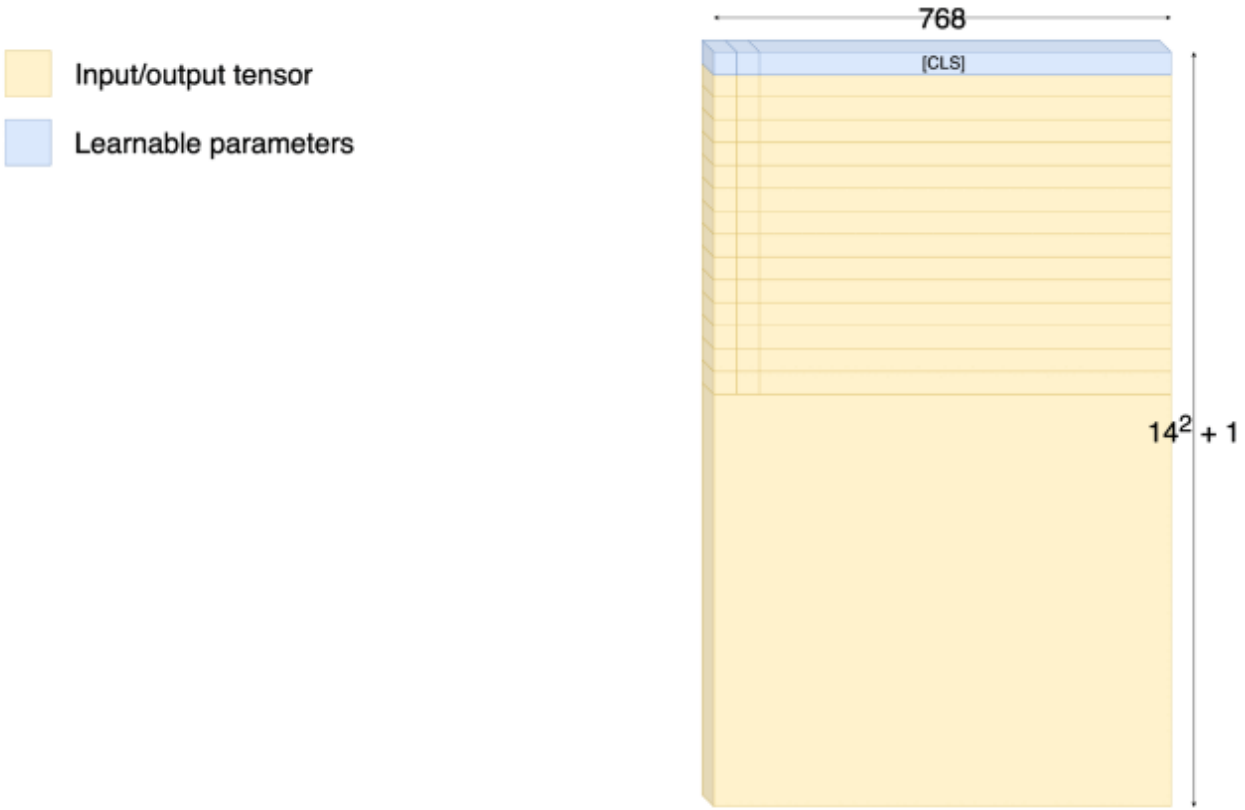


图 38: The class token is just another learnable token appended on top of the input.

ViT Position Encoding

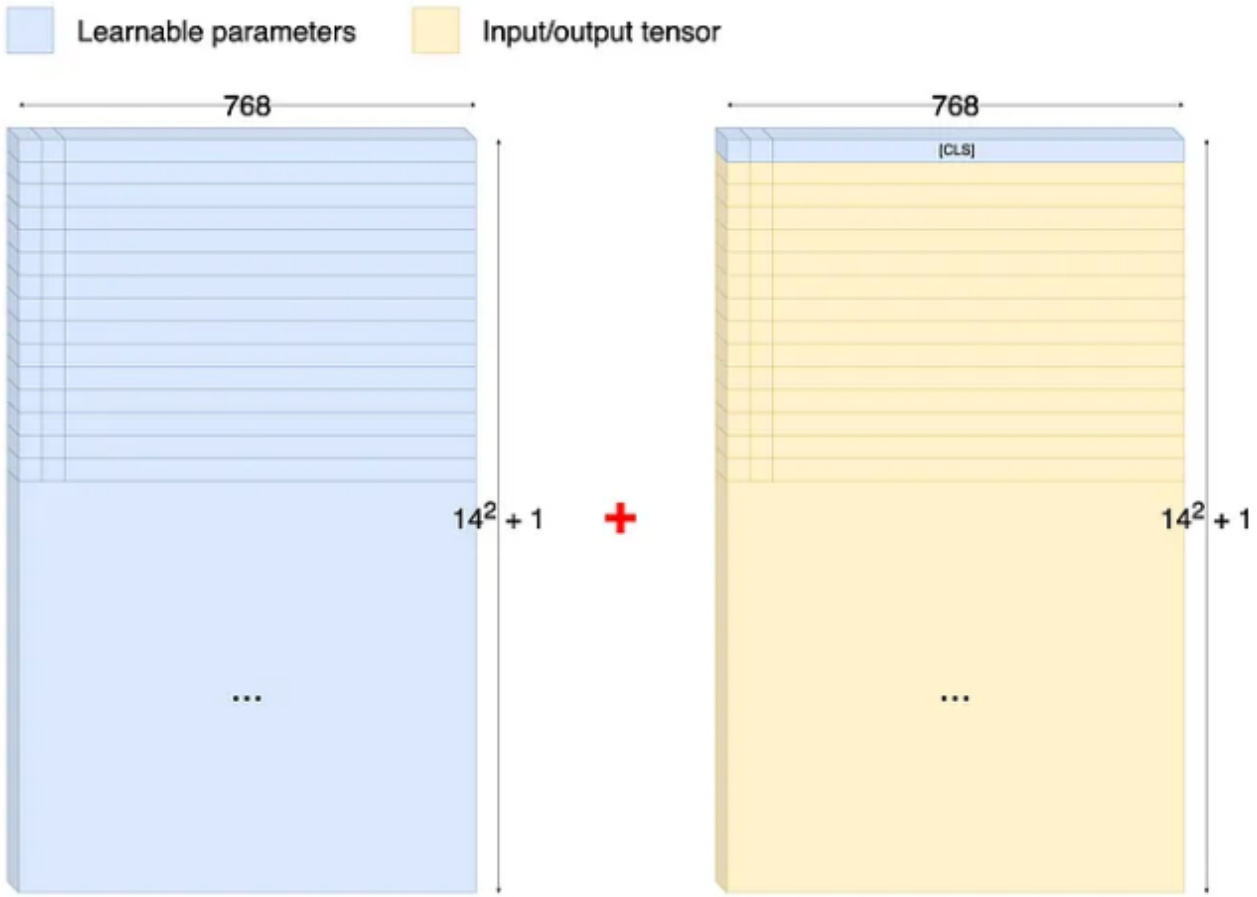


图 39: Position encoding, from [7].

References I

[1] D. Bahdanau, K. Cho, and Y. Bengio, *Neural machine translation by jointly learning to align and translate*, 2016. arXiv: 1409.0473 [cs.CL]. [Online]. Available: https://arxiv.org/abs/1409.0473.

[2] 3Blue1Brown, *Aber was ist ein gpt? visuelle einführung in transformers*, Apr. 2024. [Online]. Available: https://www.youtube.com/watch?v=wjZofJX0v4M.

[3] J. Gil, *Exploring explainability for vision transformers*, Dec. 2020. [Online]. Available: https://jacobgil.github.io/deeplearning/vision-transformer-explainability.

[4] Serrano.Academy, *Serrano.academy*, [Online]. Available: https://www.youtube.com/@SerranoAcademy.

References II

[5] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, *A convnet for the 2020s*, 2022. arXiv: 2201.03545 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2201.03545.

[6] A. Dosovitskiy, L. Beyer, A. Kolesnikov, *et al.*, *An image is worth 16x16 words: Transformers for image recognition at scale*, 2021. arXiv: 2010.11929 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2010.11929.

[7] K. Yurkova, "Vision transformer (vit) under the magnifying glass, part 1," *Medium*, Feb. 2023. [Online]. Available: https://yurkovak.medium.com/vision-transformer-vit-under-the-magnifying-glass-part-1-70be8d6661a7.

References III

[8] G. Bertasius, H. Wang, and L. Torresani, *Is space-time attention all you need for video understanding?* Jul. 2021.