# Part III

# TRANSFORMER

# TABLE OF CONTENTS

## Origin and role

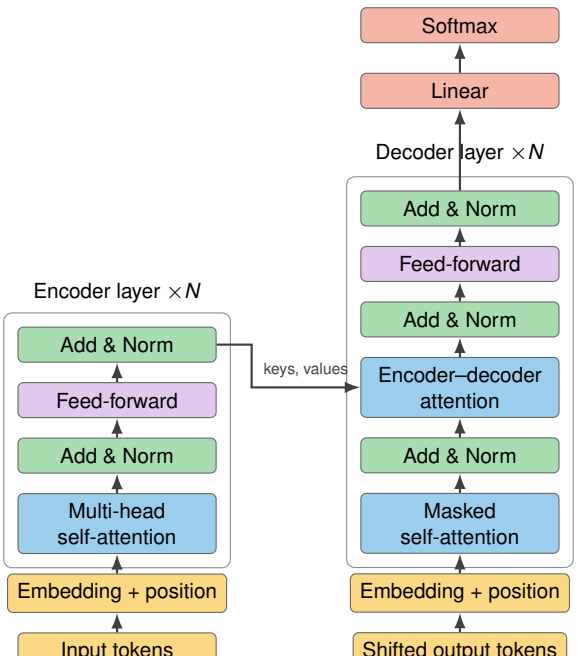
Transformers were introduced by Vaswani et al. (2017) to remove the sequential bottleneck of recurrent models (RNN, LSTM, GRU, etc). Their core operation, *self-attention*, compares all tokens in parallel and builds contextual representations by weighting token-token interactions.

- This makes them strong *sequence and set processors*: they can model long-range dependencies, condition on context, and scale efficiently with data and compute.
- With tokenized image patches, Diffusion Transformers (Peebles & Xie, 2023) show that replacing convolutional U-Nets by scalable attention blocks can substantially improve generative performance.

## Architecture

A Transformer stacks *attention*, *feed-forward layers*, *residual connections*, and *normalization*.

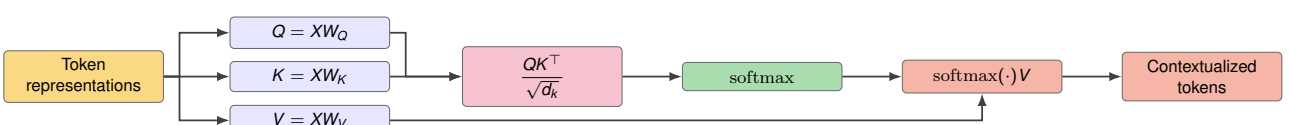
- *Attention* mixes information across tokens.
- *Feed-forward layers* transform each token independently.
- *Residuals and normalization* stabilize deep training.



## Self-attention mechanism

Each token produces a *query*, a *key*, and a *value*. Queries are matched with keys to form attention weights, which are then used to average the values:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right) V.$$



## Scaling

Transformers scale particularly well with data, parameters, and compute. Their performance improves predictably when trained with larger models and larger corpora.

- *Encoder-only* models, such as BERT, are mainly used for representation learning.
- *Decoder-only* models, such as GPT-style models, are trained autoregressively for generation.
- *Encoder–decoder* models, such as T5, are useful for sequence-to-sequence tasks.
- In practice, performance often comes from keeping attention *hardware-efficient* rather than replacing it completely.
  - *Parallel self-attention*: all token–token scores are computed by efficient matrix multiplications.
  - *FlashAttention*: keeps exact attention but reduces memory transfers on GPUs.
  - *GQA/MQA*: reduces key-value cache size and speeds up decoding.
  - *Sliding-window attention*: restricts attention locally for long-context efficiency.

# REFERENCES FOR THIS PART I



Peebles, W., & Xie, S. (2023).**Scalable diffusion models with transformers.** *International Conference on Computer Vision*.


Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017).**Attention is all you need.** *Advances in Neural Information Processing Systems*.