Effective Transformers

Алина Августёнок, БПМИ211

Friendly Reminder

$\mathbf{X} \in \mathbb{R}^{L \times d}$ – input sequence

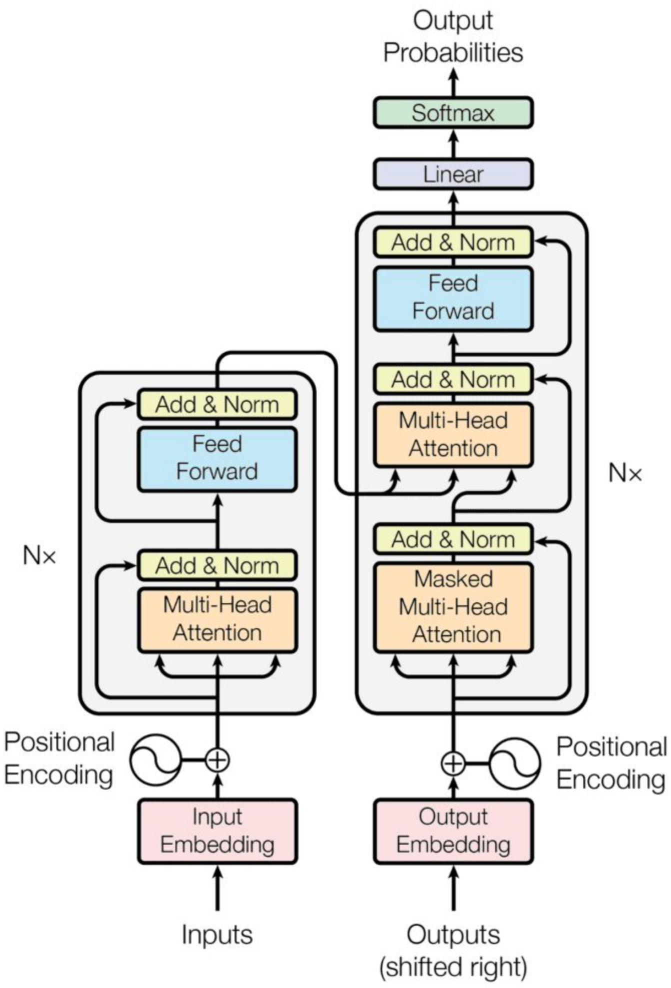
$\mathbf{Q} = \mathbf{X}\mathbf{W}^q \in \mathbb{R}^{L \times d_k}$ – query embedding inputs

$\mathbf{K} = \mathbf{X}\mathbf{W}^k \in \mathbb{R}^{L \times d_k}$ – key embedding inputs

$\mathbf{V} = \mathbf{X}\mathbf{W}^v \in \mathbb{R}^{L \times d_v}$ – value embedding inputs

Attention:

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



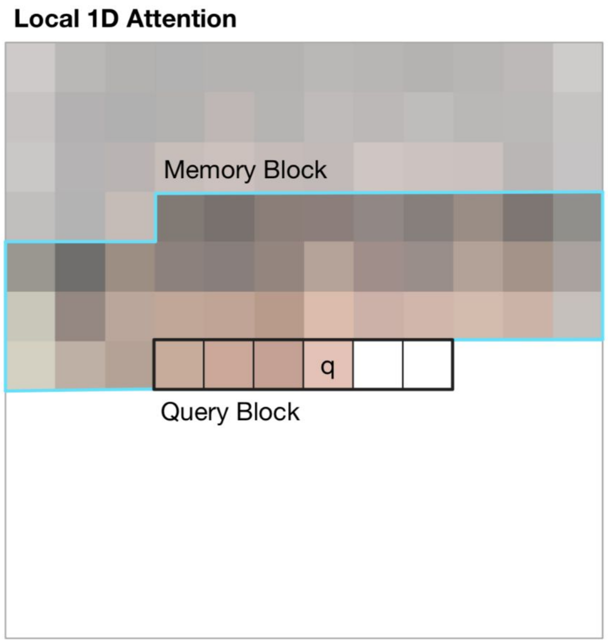
Efficient Attention

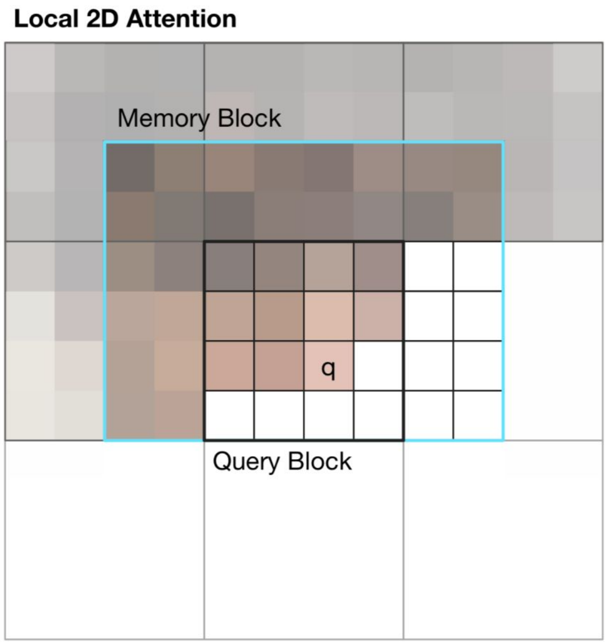
Problem: $\mathcal{O}(L^2 d)$ time and $\mathcal{O}(L^2)$ memory

Solutions:

- Sparse Attention
- Content-based Attention
- Low-Rank Attention

Sparse Attention: Fixed Local Context





q – current pixel,
M – memory block – other positions, used in computing q

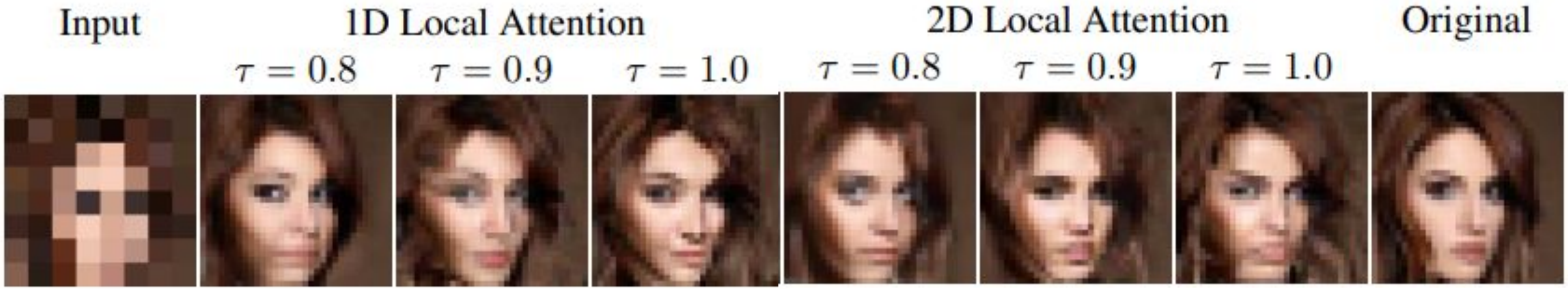
Fixed Local Context

image 8x8 ➔ image 32x32

dataset: CelebA

τ – temperature

Model Type	τ	%Fooled
ResNet	n/a	4.0
srez GAN	n/a	8.5
PixelRecursive	1.0	11.0
(Dahl et al., 2017)	0.9	10.4
	0.8	10.2
1D local	1.0	29.6 ± 4.0
Image Transformer	0.9	33.5 ± 3.5
	0.8	$\mathbf{35.94 \pm 3.0}$
2D local	1.0	30.64 ± 4
Image Transformer	0.9	34 ± 3.5
	0.8	$\mathbf{36.11 \pm 2.5}$



Content-based Attention

Problems:

- quadratic time and memory complexity in self-attention
- memory in N-layer model = N * memory in 1-layer model
- intermediate feed-forward layers can be large

Idea:

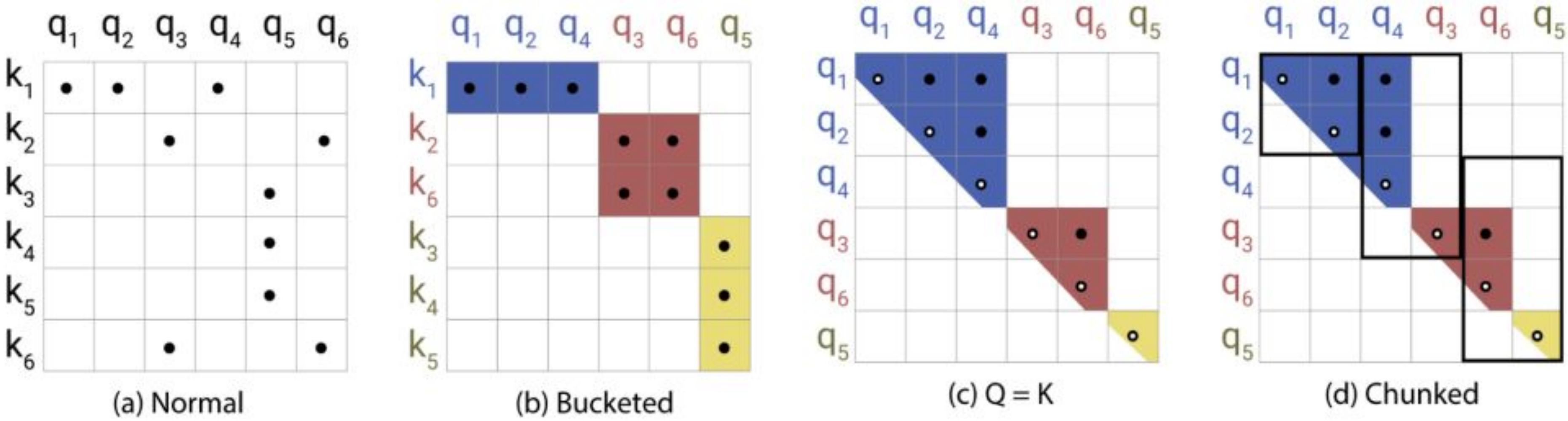
- dot-product attention ➔ locality-sensitive hashing attention
- residual blocks ➔ reversible residual layers

Locality-Sensitive Hashing Attention

$x \rightarrow h(x)$: $h(x) = \text{argmax}([xR; -xR])$,

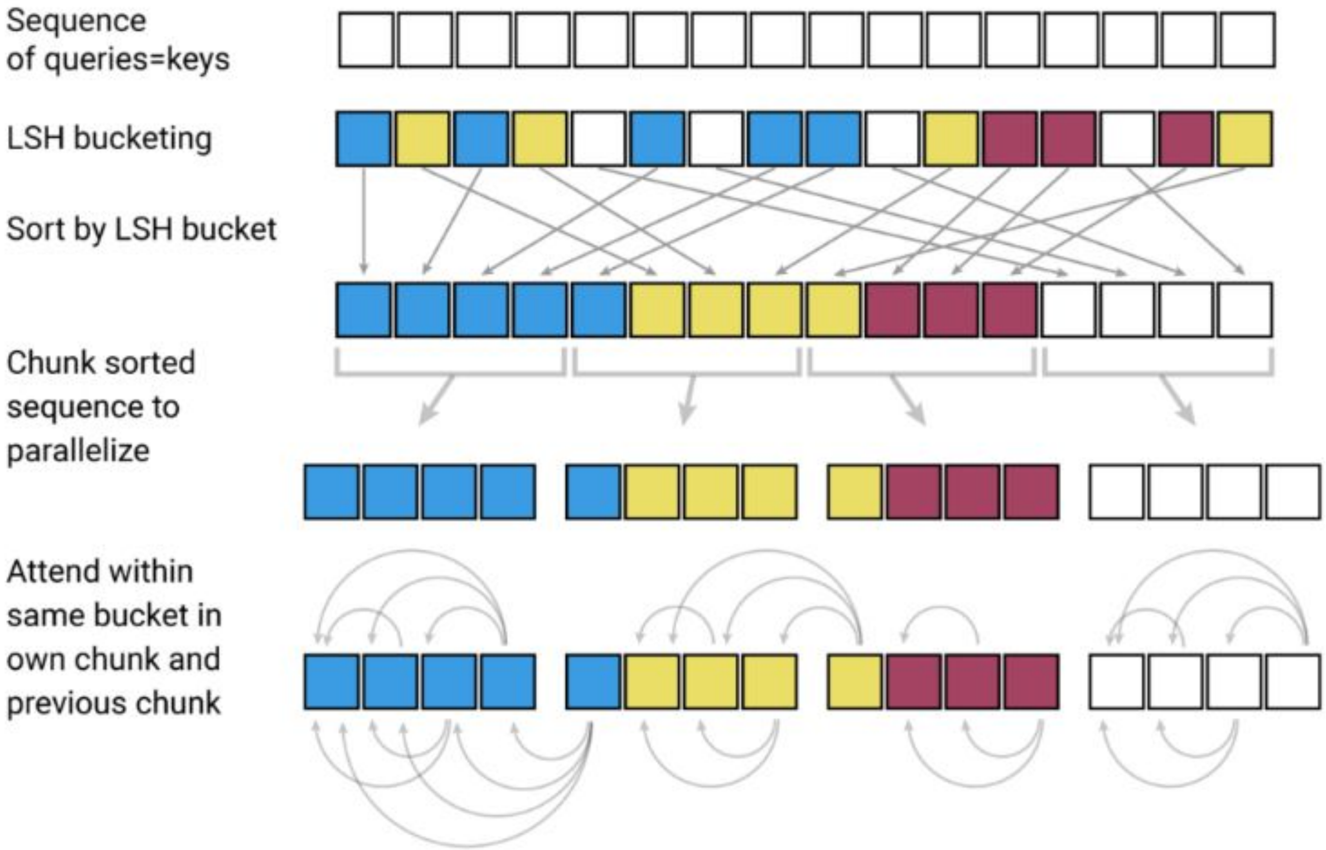
where $[\cdot\, ; \cdot]$ – concatenation, R is a random matrix,

R.shape = (d, b/2), b is a hyperparameter



(a) Normal (b) Bucketed (c) Q = K (d) Chunked

Locality-Sensitive Hashing Attention



Reversible Residual Layers

normal residual layer: $x \mapsto y$,

$$y = x + F(x)$$

reversible residual layer:

$$(x_1, x_2) \mapsto (y_1, y_2)$$

$$y_1 = x_1 + F(x_2), \; y_2 = x_2 + G(y_1)$$

$$x_2 = y_2 - G(y_1), \; x_1 = y_1 - F(x_2)$$

in transformers:

$$Y_1 = X_1 + \text{Attention}(X_2), \; Y_2 = X_2 + \text{FeedForward}(Y_1)$$

Reformer:

Locality-Sensitive Hashing Attention + Reversible Residual Layers

Table 3: Memory and time complexity of Transformer variants. We write d_{model} and d_{ff} for model depth and assume $d_{ff} \geq d_{model}$; b stands for batch size, l for length, n_l for the number of layers. We assume $n_c = l/32$ so $4l/n_c = 128$ and we write $c = 128^2$.

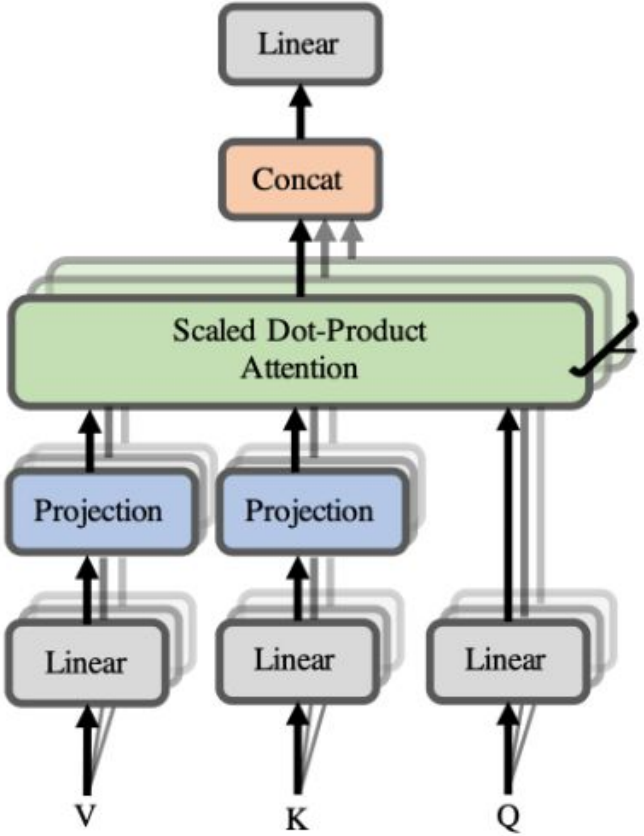
Model Type	Memory Complexity	Time Complexity
Transformer	$\max(bld_{ff}, bn_h l^2)n_l$	$(bld_{ff} + bn_h l^2)n_l$
Reversible Transformer	$\max(bld_{ff}, bn_h l^2)$	$(bn_h ld_{ff} + bn_h l^2)n_l$
Chunked Reversible Transformer	$\max(bld_{model}, bn_h l^2)$	$(bn_h ld_{ff} + bn_h l^2)n_l$
LSH Transformer	$\max(bld_{ff}, bn_h ln_r c)n_l$	$(bld_{ff} + bn_h n_r lc)n_l$
Reformer	$\max(bld_{model}, bn_h ln_r c)$	$(bld_{ff} + bn_h n_r lc)n_l$

Low-Rank Attention

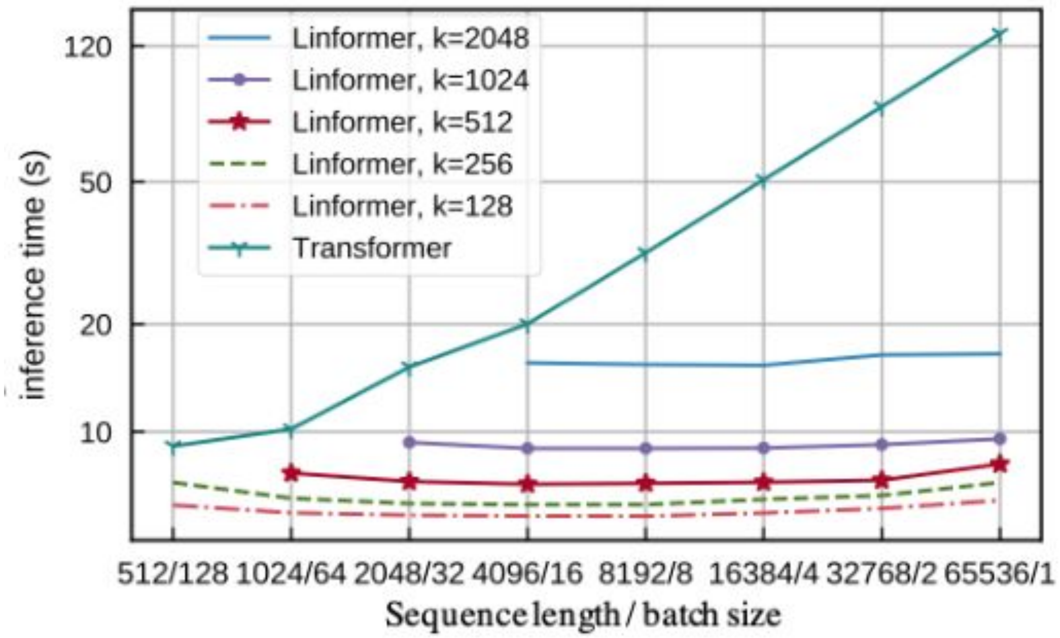
linear projections:

$$\mathbf{E}_i, \mathbf{F}_i \in \mathbb{R}^{L \times k}$$

$$\begin{aligned}\overline{\mathrm{head}_i} &= \mathrm{attn}(\mathbf{X}_q\mathbf{W}_i^q, \mathbf{E}_i\mathbf{X}_k\mathbf{W}_i^k, \mathbf{F}_i\mathbf{X}_v\mathbf{W}_i^v) \\ &= \underbrace{\mathrm{softmax}\left(\frac{\mathbf{X}_q\mathbf{W}_i^q(\mathbf{E}_i\mathbf{X}_k\mathbf{W}_i^k)^\top}{\sqrt{d}}\right)}_{\text{low rank attention matrix } \bar{A} \in \mathbb{R}^{k \times d}}\mathbf{F}_i\mathbf{X}_v\mathbf{W}_i^v\end{aligned}$$



Low-Rank Attention



Summing Up

Efficient Attention:

- Sparse Attention: Fixed Local Context
- Content-based Attention: Locality-Sensitive Hashing Attention and Reversible Residual Layers
- Low-Rank Attention

Source:

https://lilianweng.github.io/posts/2023-01-27-the-transformer-family-v2/