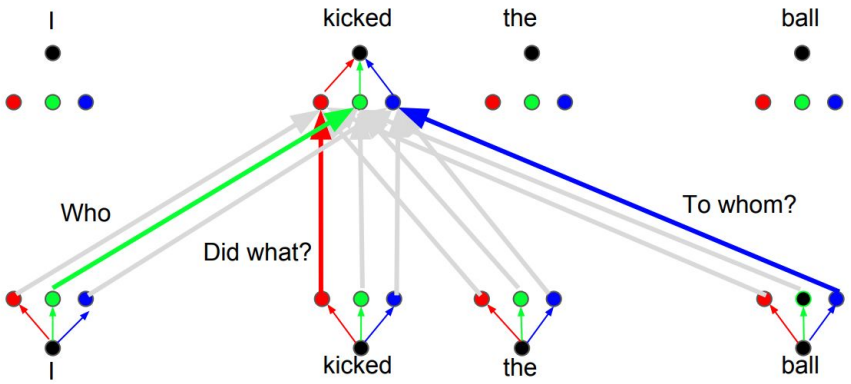
Sparse attention transformers

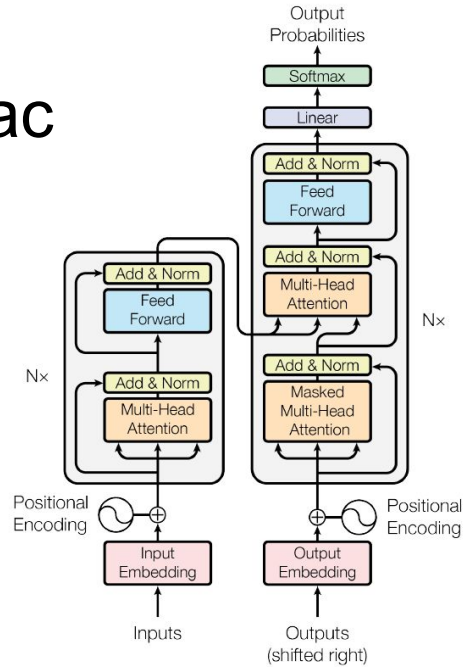
Шишков Алексей
ПМИ 192

Как есть сейчас

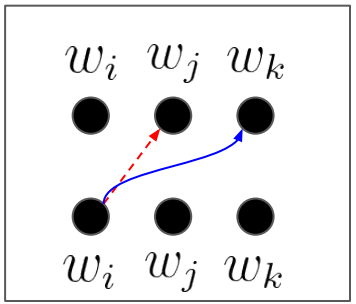
Сложность на одном слое — $O(n^2)$:

Для каждого токена сопоставляем пары ключ-значение для каждого токена предыдущего слоя

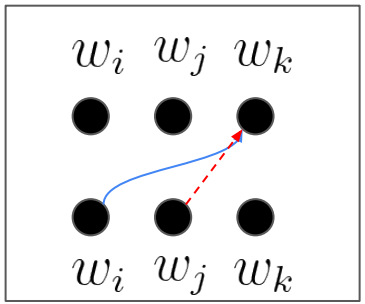




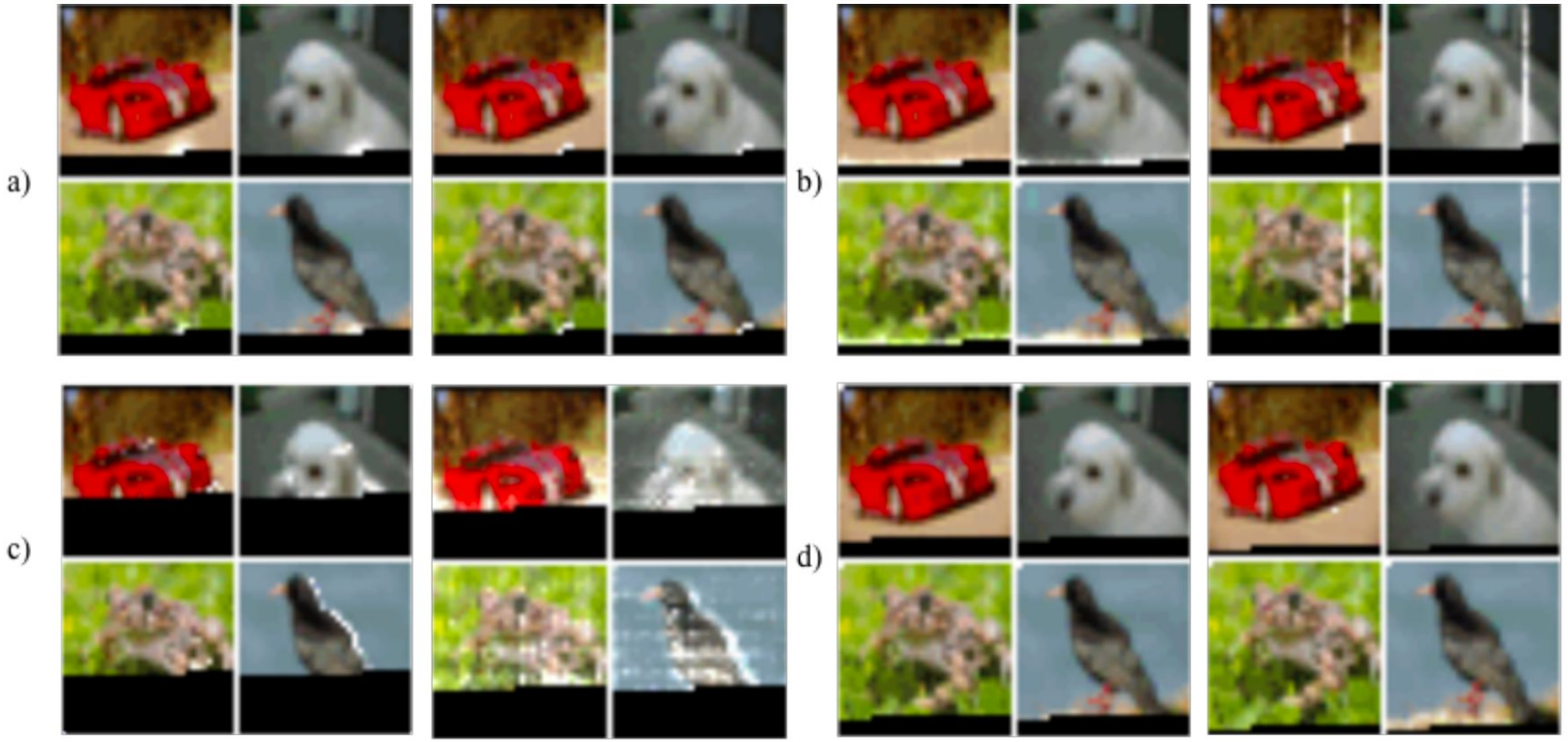
Слой t



Слой t+1



Откуда пришла идея?



Внимание… Спасибо за внимание

- $\mathcal{S}$ — шаблон связности. $\mathcal{S} = \{S_1, \ldots, S_n\}$
- $\text{Attend}(\mathbf{X}, \mathcal{S}) = \Big(a(\mathbf{x}_i, S_i)\Big)_{i \in \{1,\ldots,L\}}$,

$$a(\mathbf{x}_i, S_i) = \text{softmax}\Big(\frac{(\mathbf{x}_i \mathbf{W}^q)(\mathbf{x}_j \mathbf{W}^k)^\top_{j \in S_i}}{\sqrt{d_k}}\Big)(\mathbf{x}_j \mathbf{W}^v)_{j \in S_i}$$

- Разбиваем $\mathcal{S}$ на непересекающиеся подмножества:

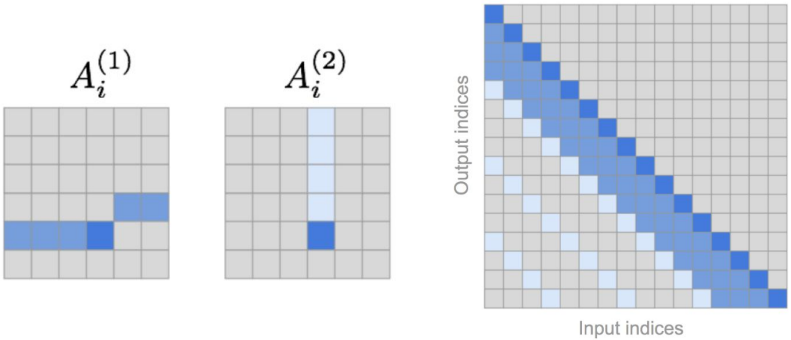
$$A_i^{(m)} \subset S_i, m = 1, \ldots, p$$

- Если $(j, a, b, c, \ldots, i)$ — путь, то $j \in A_a^{(1)}, a \in A_b^{(2)}, b \in A_c^{(3)}, \ldots$

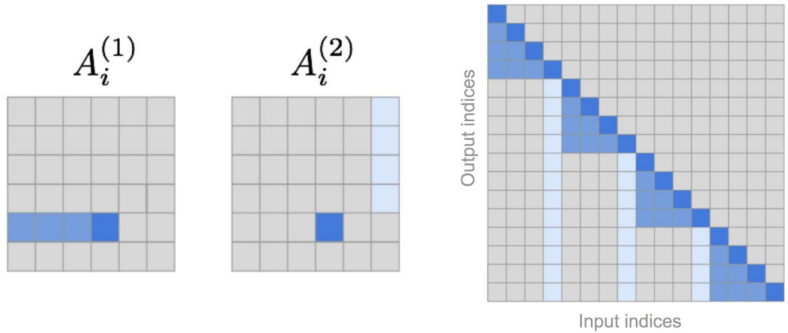
~~Love~~ ~~Attention~~ Sparse Attention Is All You Need

Sparse attention

Strided attention

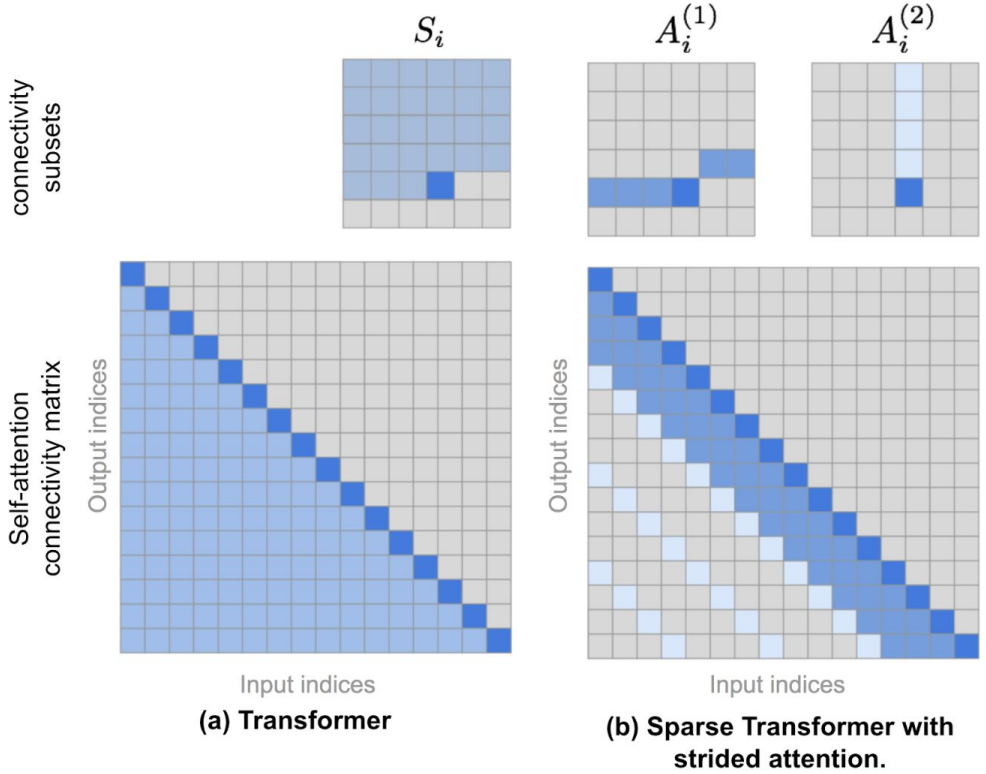


Fixed attention



Strided attention

- $\ell \sim \sqrt{n}$
- Смотрит на предыдущие ℓ токенов в порядке C-memory-order и на пиксели в той же колонке



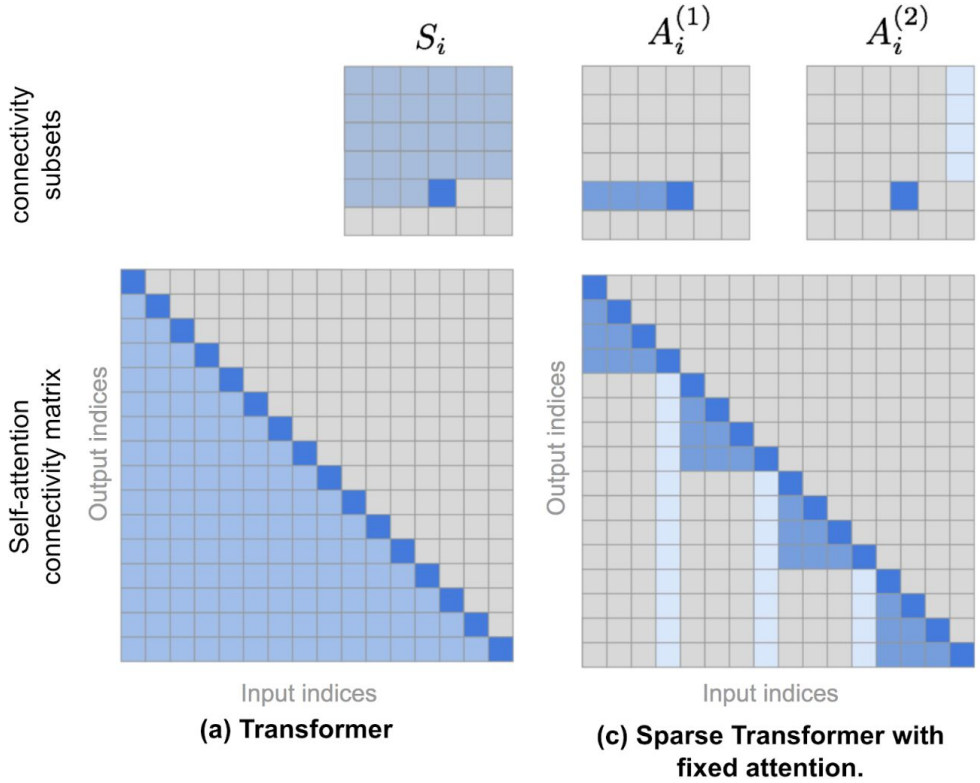
$$A_i^{(1)} = \{t, t+1, \ldots, i\}, \text{ where } t = \max(0, i - \ell)$$

$$A_i^{(2)} = \{j : (i - j) \mod \ell = 0\}$$

Fixed attention

- Смотрит до начала строки и на c колонок справа

- $c \in \{8, 16, 32\}, \ell \in \{128, 256\}$



(a) Transformer

(c) Sparse Transformer with fixed attention.

$$A_i^{(1)} = \{j : \lfloor \frac{j}{\ell} \rfloor = \lfloor \frac{i}{\ell} \rfloor\}$$

$$A_i^{(2)} = \{j : j \mod \ell \in \{\ell - c, \ldots, \ell - 1\}\}$$

Как строить модель?

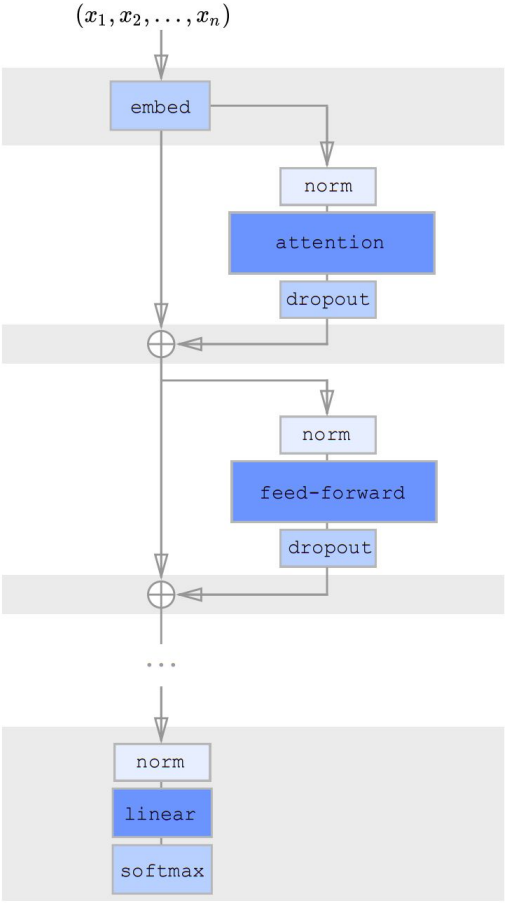
1. Один тип sparse attention на каждый слой, с каждым новым слоем чередовать

$$\text{attention}(\mathbf{X}) = \text{Attend}(\mathbf{X}, A^{(n \pmod p)})\mathbf{W}^o$$

2. Одна голова, область внимания — объединение областей

$$\text{attention}(\mathbf{X}) = \text{Attend}(\mathbf{X}, \cup_{m=1}^{p} A^{(m)})\mathbf{W}^o$$

3. Multi-head, головы выбираются как в первых двух пунктах



Итоги

Model	Bits per byte
CIFAR-10	
PixelCNN (Oord et al., 2016)	3.03
PixelCNN++ (Salimans et al., 2017)	2.92
Image Transformer (Parmar et al., 2018)	2.90
PixelSNAIL (Chen et al., 2017)	2.85
Sparse Transformer 59M (strided)	**2.80**
Enwik8	
Deeper Self-Attention (Al-Rfou et al., 2018)	1.06
Transformer-XL 88M (Dai et al., 2018)	1.03
Transformer-XL 277M (Dai et al., 2018)	**0.99**
Sparse Transformer 95M (fixed)	**0.99**
ImageNet 64x64	
PixelCNN (Oord et al., 2016)	3.57
Parallel Multiscale (Reed et al., 2017)	3.7
Glow (Kingma & Dhariwal, 2018)	3.81
SPN 150M (Menick & Kalchbrenner, 2018)	3.52
Sparse Transformer 152M (strided)	**3.44**
Classical music, 5 seconds at 12 kHz	
Sparse Transformer 152M (strided)	**1.97**

Model	Bits per byte	Time/Iter
Enwik8 (12,288 context)		
Dense Attention	1.00	1.31
Sparse Transformer (Fixed)	**0.99**	0.55
Sparse Transformer (Strided)	1.13	0.35
CIFAR-10 (3,072 context)		
Dense Attention	2.82	0.54
Sparse Transformer (Fixed)	2.85	0.47
Sparse Transformer (Strided)	**2.80**	0.38

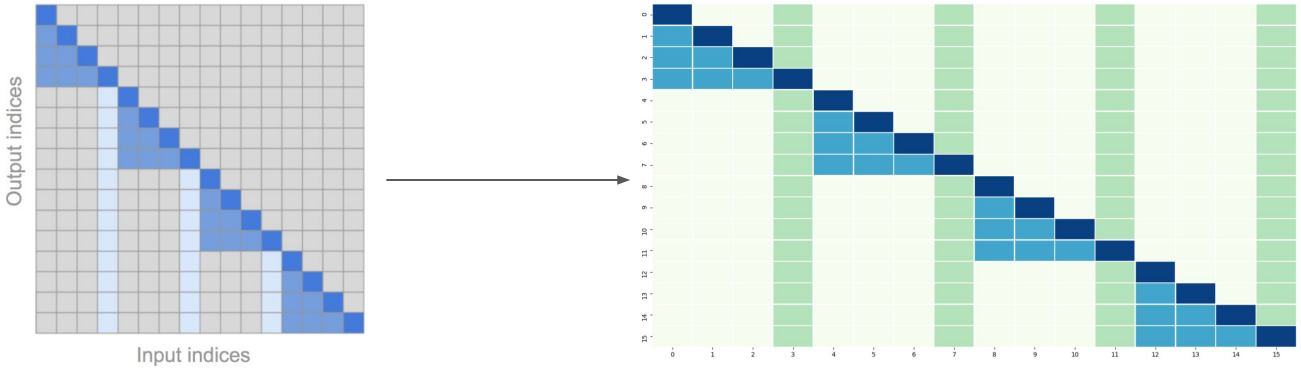
Table 3. We observe increased compression of Enwik8 with longer contexts, suggesting the Sparse Transformer can effectively incorporate long-term dependencies.

Minimum context length during evaluation	Bits per byte
6,144 tokens	0.9952
9,216 tokens	0.9936
10,752 tokens	0.9932
11,904 tokens	0.9930
12,096 tokens	0.9922
12,160 tokens	**0.9908**

Неужели всё идеально?

- Не Bi-directional

  Решение:



- Насколько хорошо это будет работать в конкретной задаче?

Спасибо за внимание!

Надеюсь, оно не было слишком sparse

[послушать]

Литература

- https://arxiv.org/abs/1904.10509
- https://lilianweng.github.io/lil-log/2020/04/07/the-transformer-family.html
- https://www.geeksforgeeks.org/sparse-transformer-stride-and-fixed-factorized-attention
- https://www.youtube.com/watch?v=KwKr_e7xBQ4