Трансформер

Коган Александра БПМИ-182

Что? Зачем?

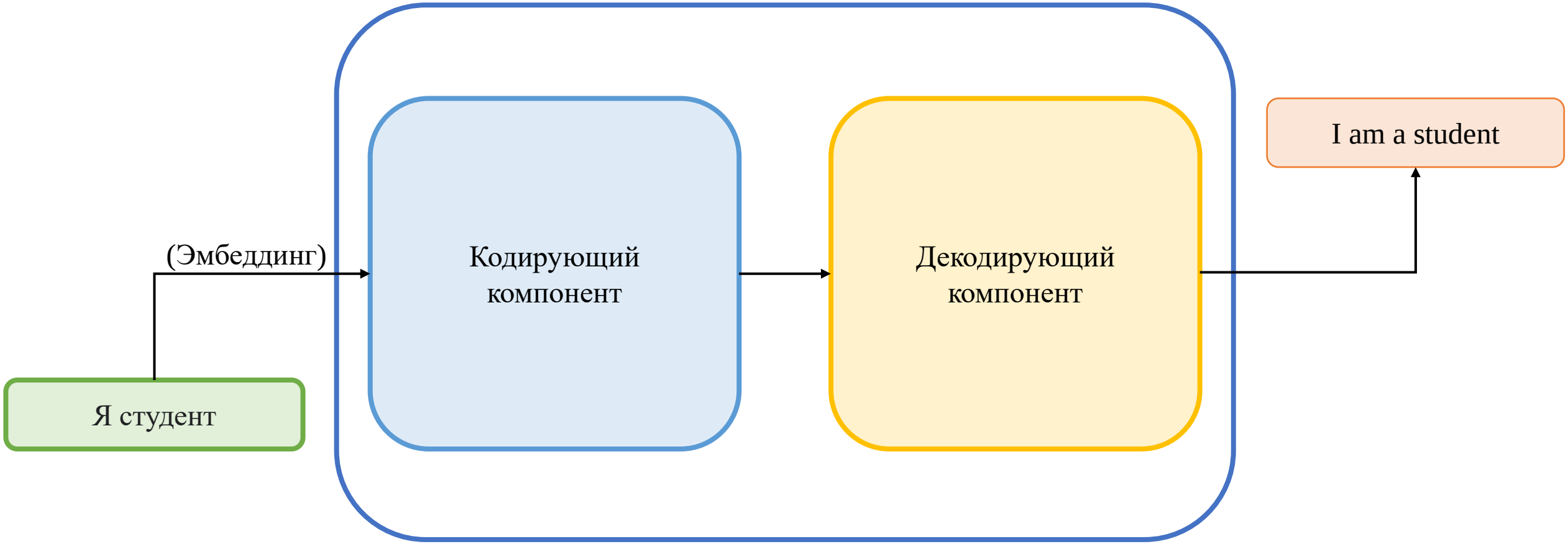
Рекуррентность, RNN

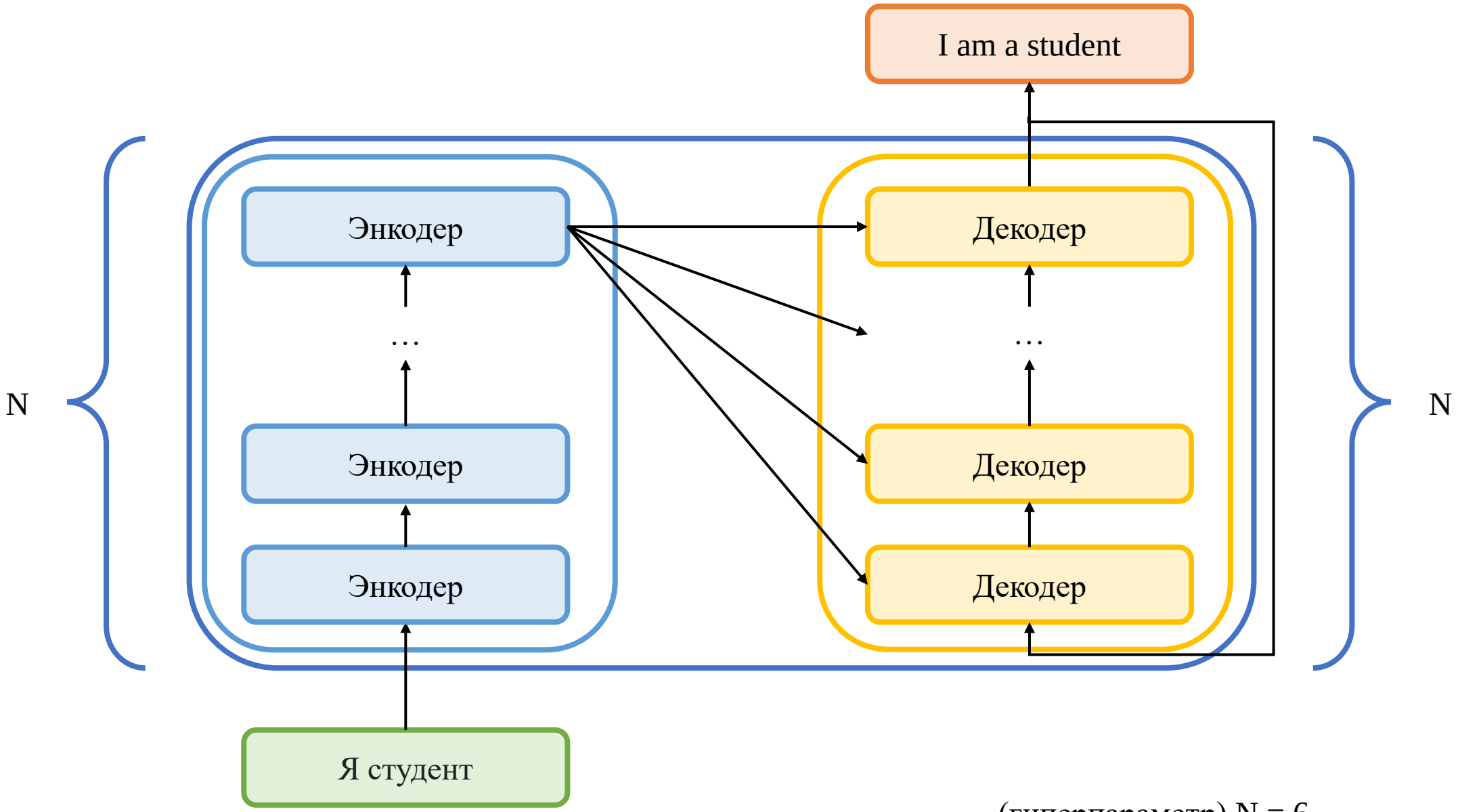
Внимание

- Быстрее обучается
- Можно параллелить

Трансформер

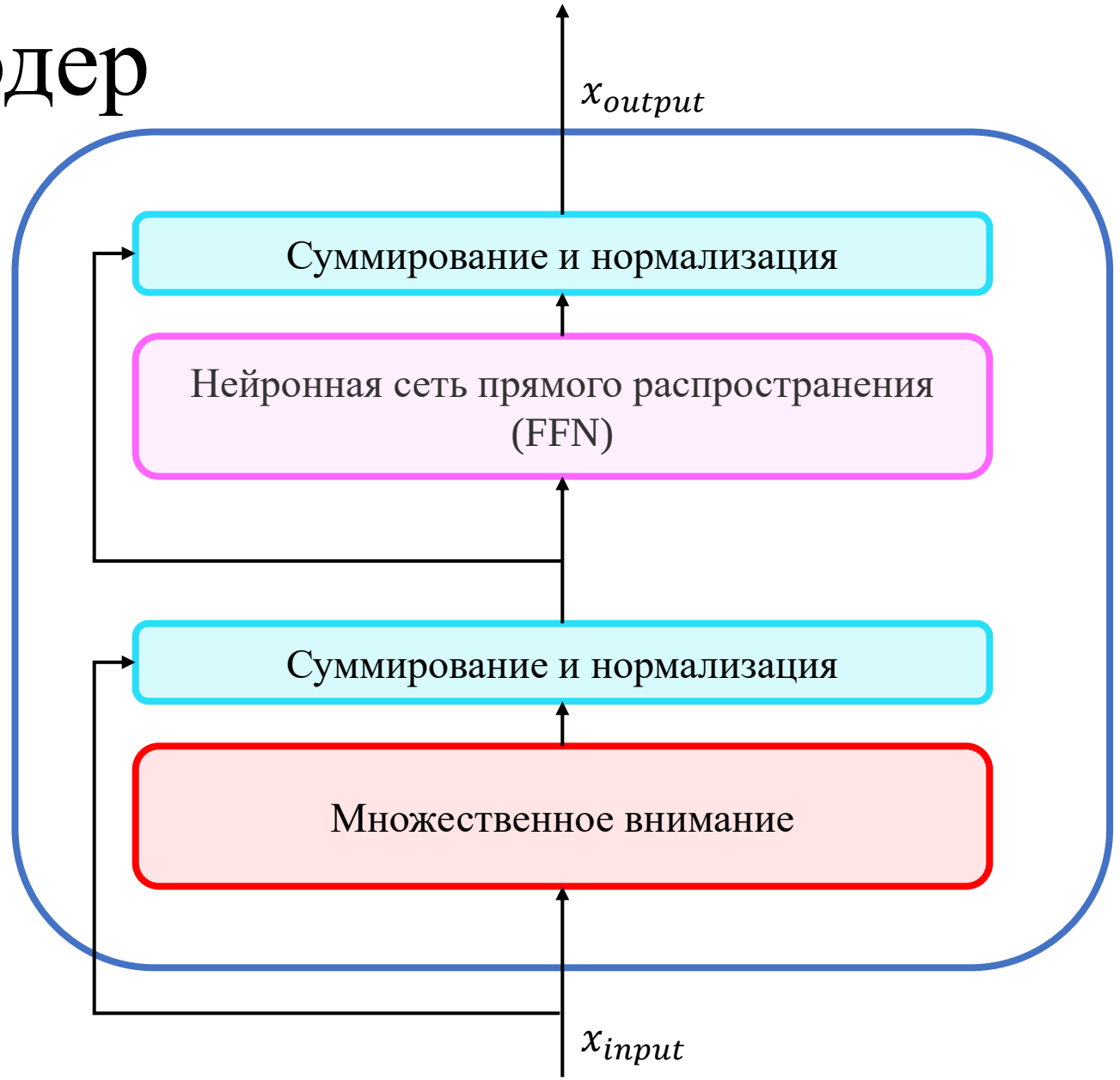

Я студент
Трансформер
I am a student


(Эмбеддинг)
Кодирующий компонент
Декодирующий компонент
I am a student
Я студент



(гиперпараметр) N = 6

Энкодер



LayerNorm(x + Sublayer(x))

x – вход подуровня
Sublayer(x) – выход подуровня

LayerNorm(x + Sublayer(x))

$\dim x_{input} = \dim x_{output} = d_{model}$

(гиперпараметр) $d_{model} = 512$

Нейронная сеть прямого распространения (FFN)

$$FFN(x) = \max(0, xW_1 + b_1)\, W_2 + b_2$$

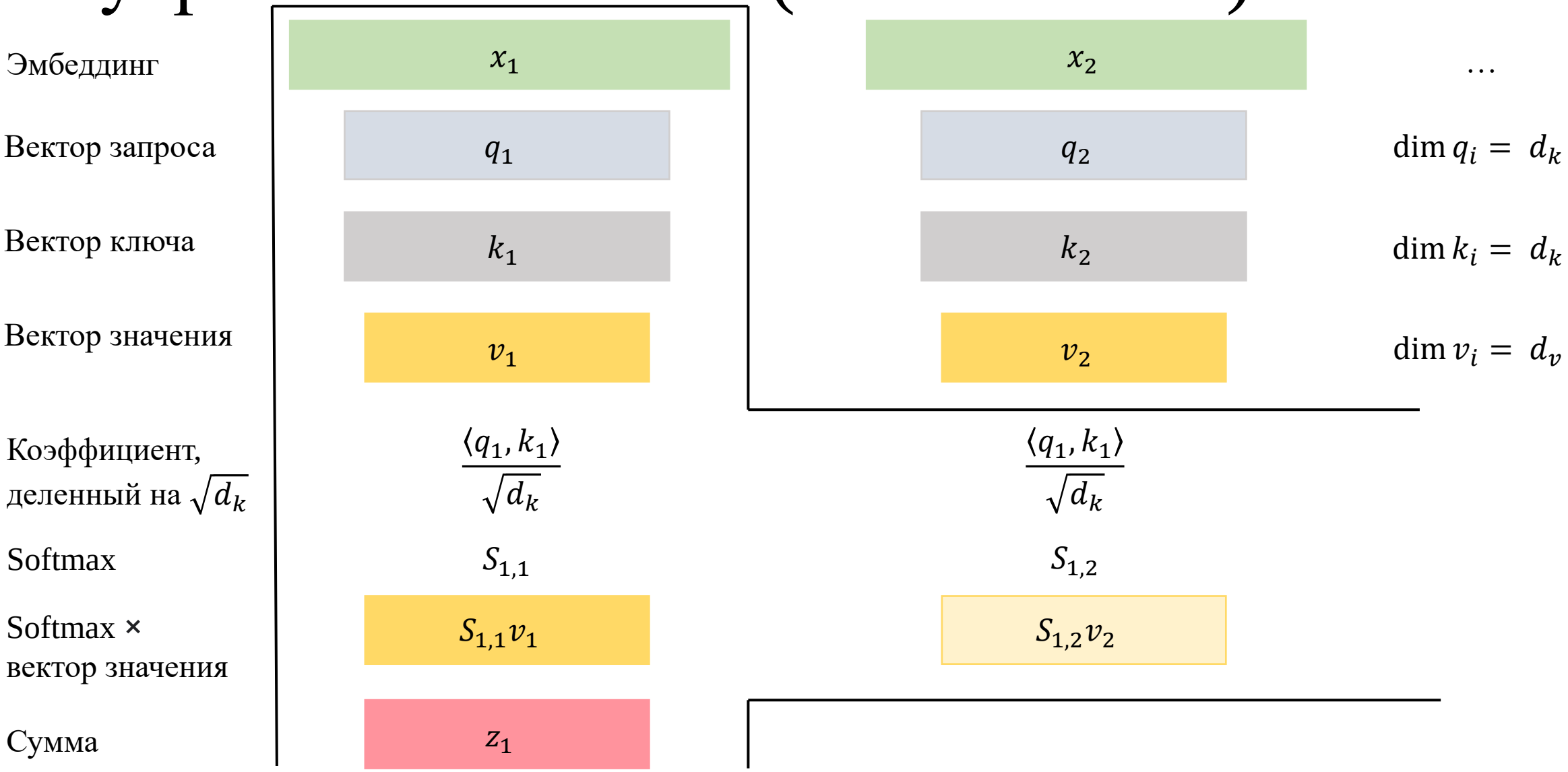
$$\dim x = \dim FFN(x) = d_{model}$$

$$\dim hidden\ layer = d_{ff}$$

$W_1 \in \mathbb{R}^{d_{model} \times d_{ff}}$, $W_2 \in \mathbb{R}^{d_{ff} \times d_{model}}$, $b_1 \in \mathbb{R}^{d_{ff}}$, $b_2 \in \mathbb{R}^{d_{model}}$
обучаются

(гиперпараметр) $d_{ff} = 2048$

Внутреннее внимание (Self-attention)



Внутреннее внимание (Scaled Dot-Product Attention)

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

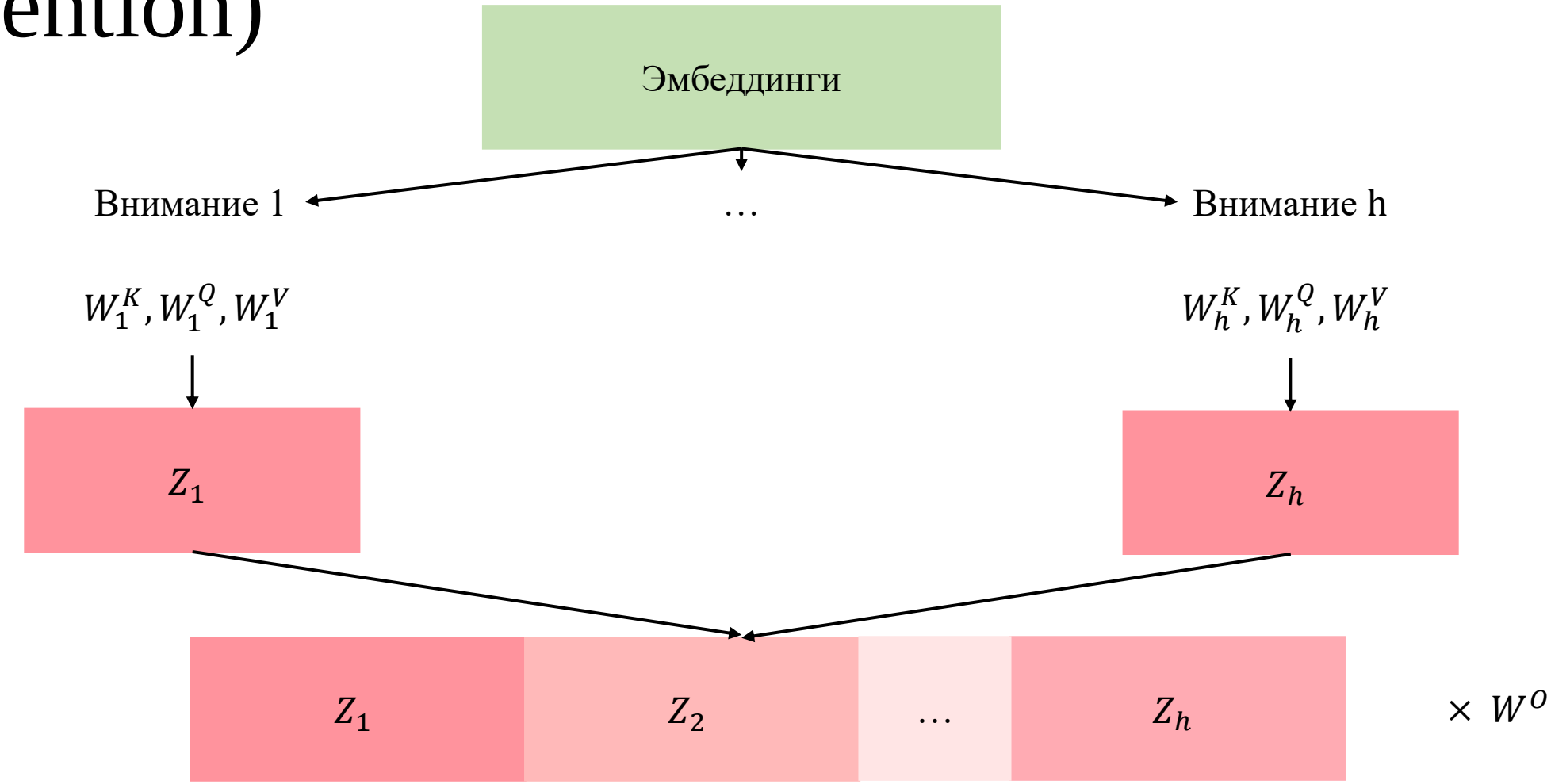
Q – матрица запросов
K – матрица ключей
V – матрица значений

$$Q = X\,W^Q$$
$$K = X\,W^K$$
$$V = X\,W^V$$

$W^K \in \mathbb{R}^{d_{model} \times d_k}, W^Q \in \mathbb{R}^{d_{model} \times d_k}, W^V \in \mathbb{R}^{d_{model} \times d_v}$ обучаются

(гиперпараметр) $d_k = 64$
(гиперпараметр) $d_v = 64$

Множественное внимание (Multi-Head Attention)



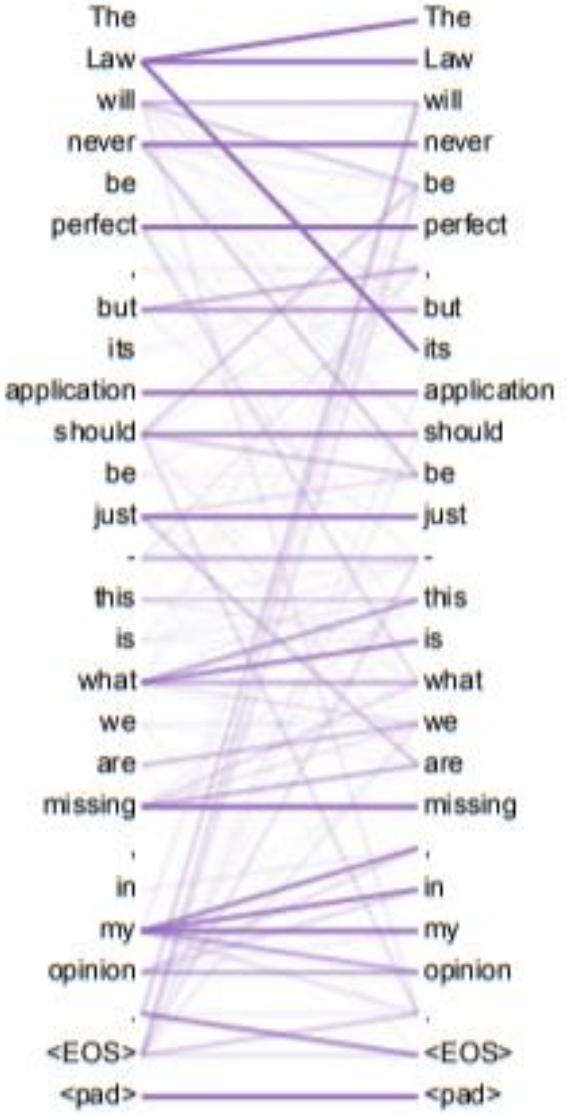
Множественное внимание

$$MultiHead(X) = Concat(Z_1, \dots, Z_h)W^O$$

$$Z_i = Attention(XW_i^Q, X\,W_i^K, XW_i^V)$$

$W^O \in \mathbb{R}^{hd_v \times d_{model}}$ обучается

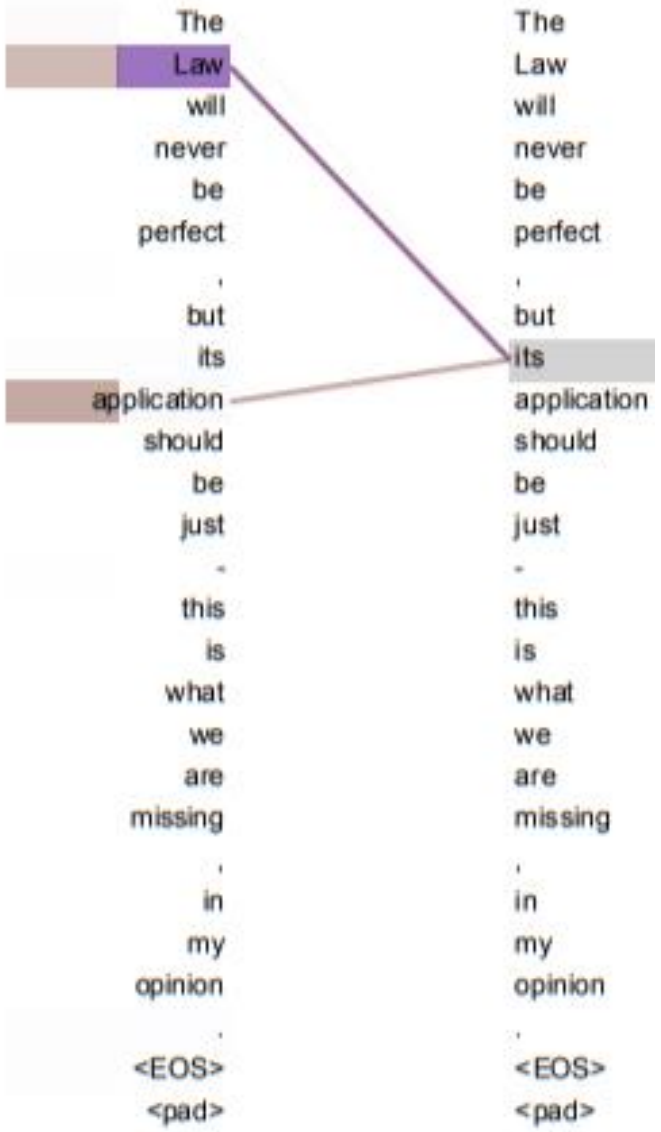
(гиперпараметр) h = 8



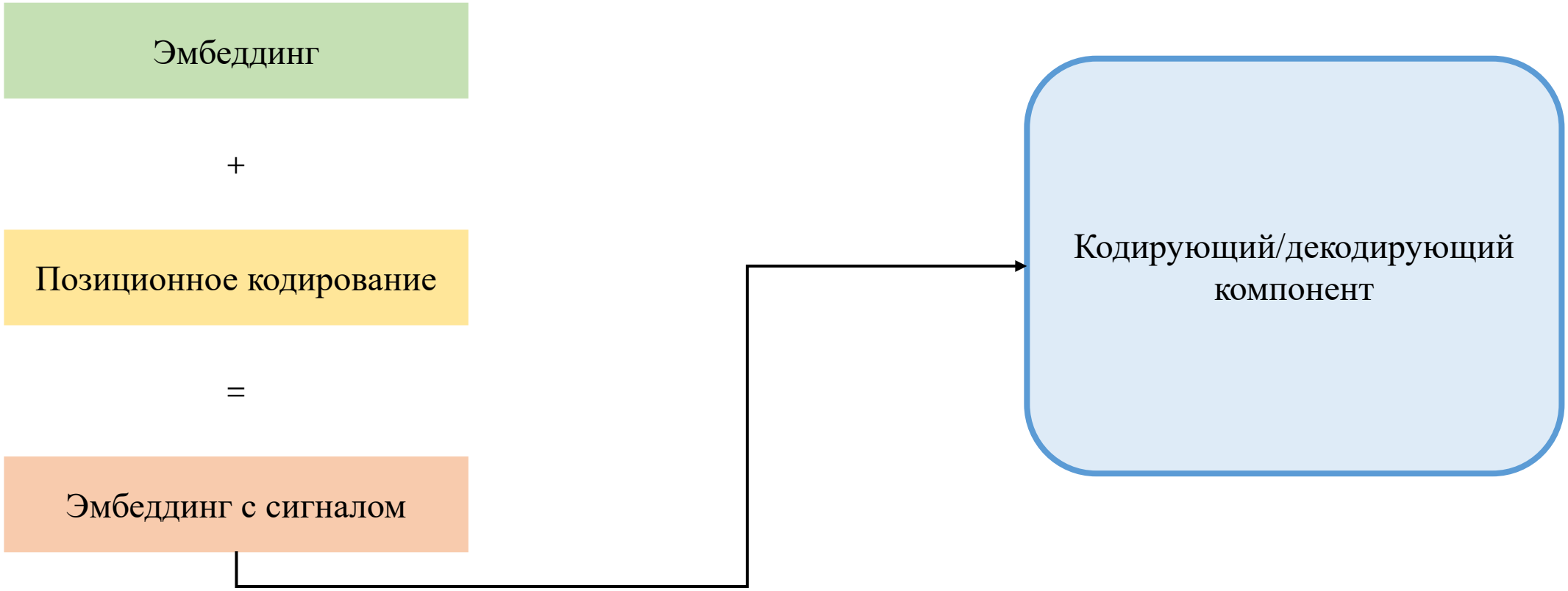
Энкодер 5

Все внимание для головы 5

Изолированное внимание для слова “its” Головы 5,6



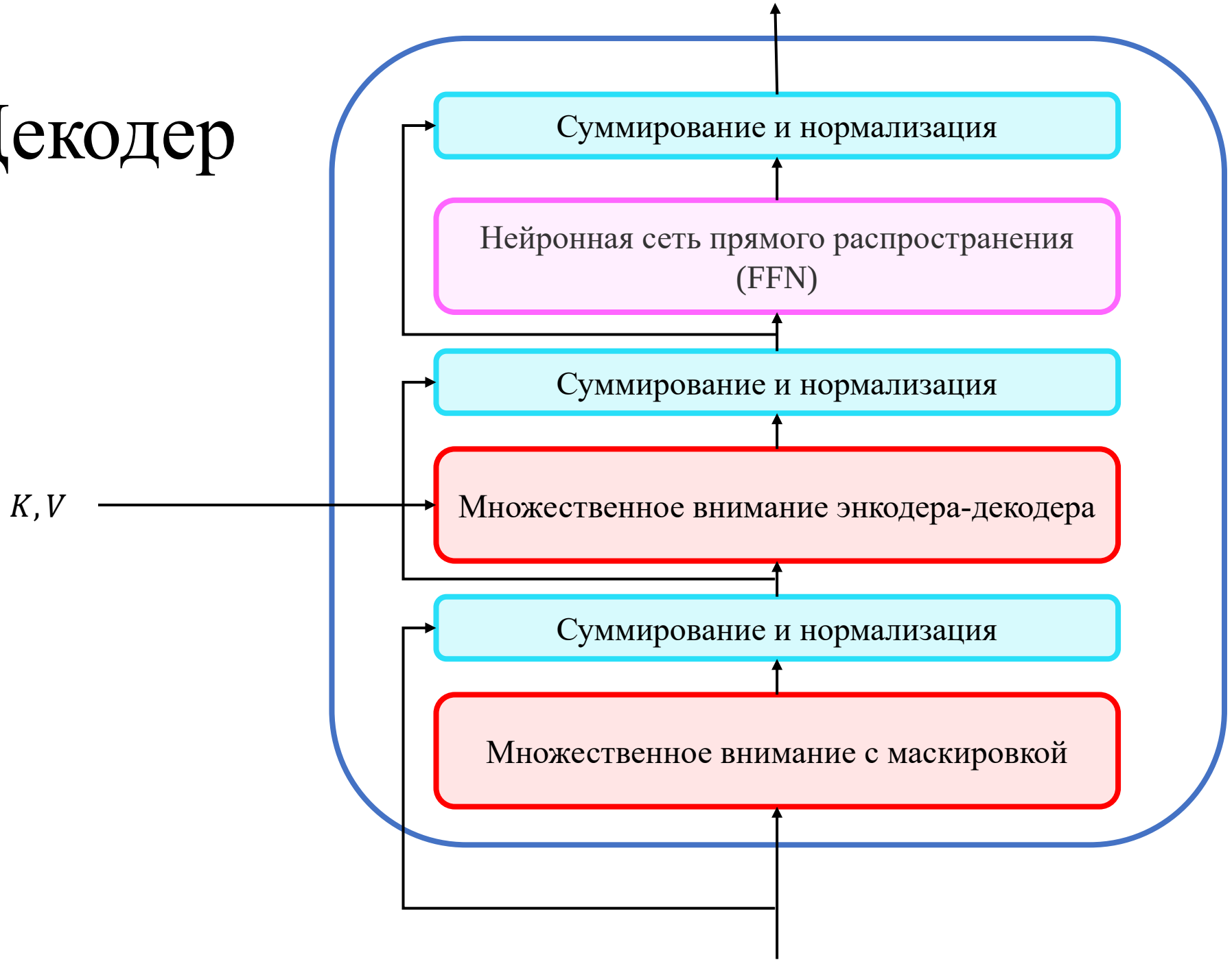
Позиционное кодирование



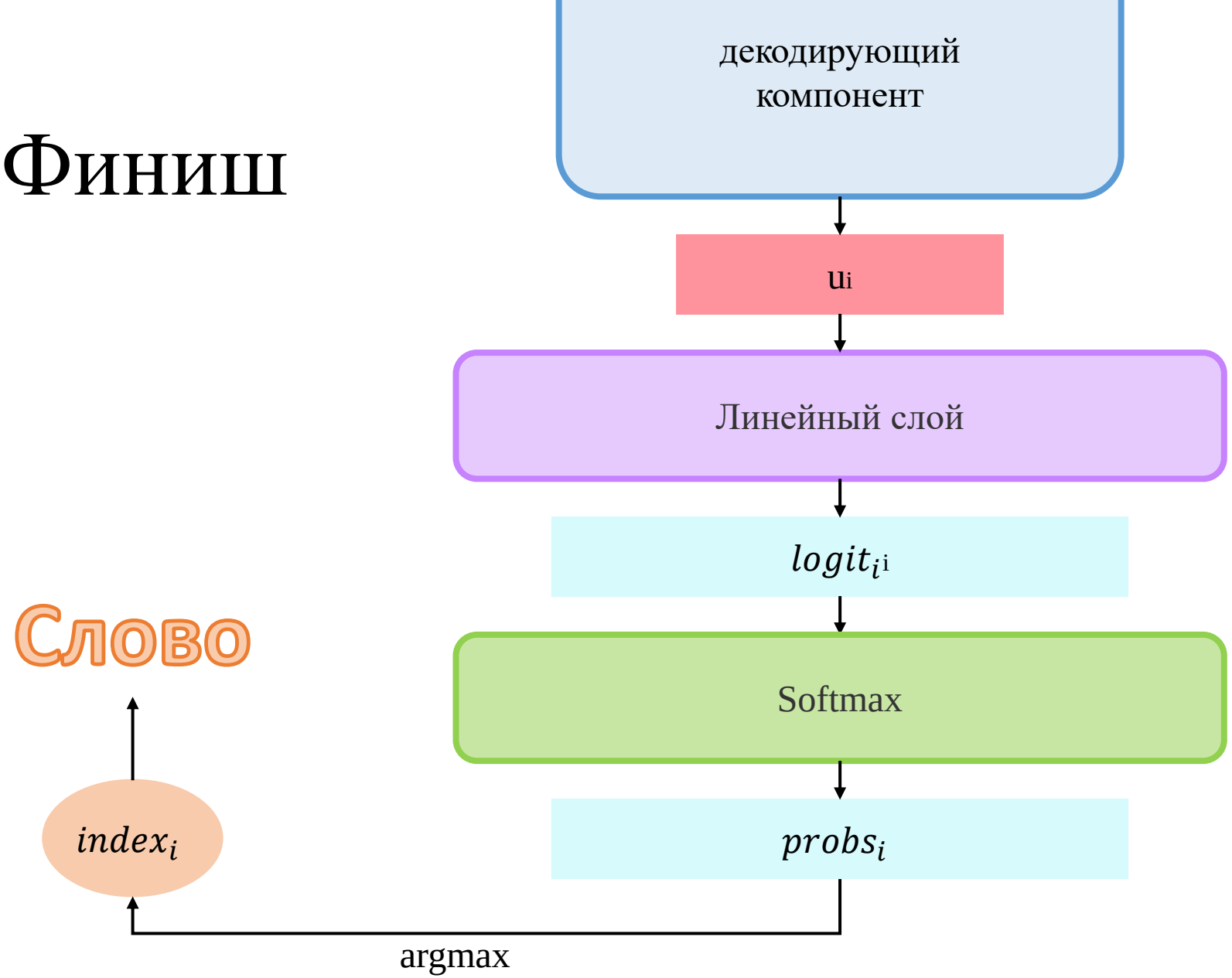
Позиционное кодирование

$$PE(t)^{(i)} = \begin{cases} \sin \frac{t}{(10000)^{2k/d_{model}}}, i = 2k, k \epsilon \mathbb{N} \cup \{0\} \\ \cos \frac{t}{(10000)^{2k/d_{model}}}, i = 2k+1, k \epsilon \mathbb{N} \cup \{0\} \end{cases}$$

Декодер


Суммирование и нормализация
Нейронная сеть прямого распространения (FFN)
Суммирование и нормализация
K, V
Множественное внимание энкодера-декодера
Суммирование и нормализация
Множественное внимание с маскировкой

Финиш



$vocab\_size$ – размер словаря

$\dim u_i = d_{model}$

Логиты

$\dim logit_i = vocab\_size$

Вероятности

$\dim probs_i = vocab\_size$

Оптимизации/Регуляризации

Adam $\beta_1, = 0.9\ \beta_2 = 0.98, \epsilon = \ 10^{-9},$

$learning\ rate = \ d_{model}^{-0.5} \min(k^{-0.5}, k\ \times warmup\_steps^{-1.5})$ k – номер шага

Использовалось $\text{warmup\_steps} = 4000$

Residual Dropout (гиперпараметр) $P_{drop} = 0.1$

Label Smoothing (гиперпараметр) $\epsilon_{ls} = 0.1$

Обучение

WMT 2014 английско-немецкий набор данных:

~ 4.5 миллиона пар предложений

Словарь ~ 37000 токенов (лексем)

WMT 2014 английско-французский набор данных:

~ 36 миллионов предложений

Словарь ~ 32000 токенов(лексем)

8 NVIDIA P100 GPUs.

Базовая модель с описанными гиперпараметрами:

- Шаг обучения – 0.4 секунды
- (гиперпараметр) количество шагов = 100 000
- 12 часов

Большая модель:

- Шаг обучения – 1.0 секунды
- 300 000 шагов
- 3.5 дней

+Beam search $\alpha = 0.6, beam\_size = 4$

Модель	BLEU		Цена обучения (FLOPs)	
	Английский - немецкий	Английский-французский	Английский - немецкий	Английский-французский
ByteNet	23.75			
Deep-Att + PosUnk		39.2		$1.0 \cdot 10^{20}$
GNMT + RL	24.6	39.92	$2.3 \cdot 10^{19}$	$1.4 \cdot 10^{20}$
ConvS2S	25.16	40.46	$9.6 \cdot 10^{18}$	$1.5 \cdot 10^{20}$
MoE	26.03	40.56	$2.2 \cdot 10^{19}$	$1.02 \cdot 10^{20}$
Deep-Att + PosUnk Ensemble		40.4		$8.0 \cdot 10^{20}$
GNMT + RL Ensemble	26.3	41.16	$18 \cdot 10^{20}$	$1.1 \cdot 10^{21}$
ConvS2S Ensemble	26.36	41.29	$7.7 \cdot 10^{19}$	$1.2 \cdot 10^{21}$
Трансформер (базовая модель)	**27.3**	38.1	$3.3 \cdot 10^{18}$	
Трансформер (большой)	**28.4**	**41.8**	$2.3 \cdot 10^{19}$	

	N	d_{model}	d_{ff}	h	d_k	d_v	P_{drop}	ϵ_{ls}	К-во шагов	BLEU	К-во параметров $\cdot 10^6$
Базовая	6	512	2048	8	64	64	0.1	0.1	100000	25.8	65
(A)				1	512	512				24.9	
				4	128	128				25.5	
				16	32	32				25.8	
				32	16	16				25.4	
(B)					16					25.1	58
					32					25.4	60
(C)	2									23.7	36
	4									25.3	50
	8									25.5	80
		256			32	32				24.5	28
		1024			128	128				26.0	168
			1024							25.4	53
			4096							26.2	90
(D)							0.0			24.6	
							0.2			25.5	
								0.0		25.3	
								0.2		25.7	
(E)	Обучаемое позиционное кодирование									25.7	
Большая		1024	4096	16			0.3		300000	26.4	213

Английско-немецкий development set, newstest2013

Constituency Parsing

N= 4
$d_{model} = 1024$

Wall Street Journal (WSJ) :

~ 40 000 предложений.

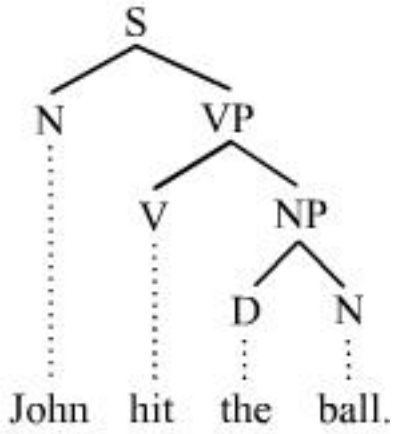
Словарь ~ 37000 токенов

Berkley Parser корпус (Berkley Parser -учитель):

~ 17 000 000 предложений.

Словарь ~ 37000 токенов

+Beam search $\alpha = 0.3, beam\_size = 21$



Парсер	Обучение	WSJ 23 F1
Vinyals & Kaiser el al. (2014)	WSJ	88.3
Petrov et al. (2006)	WSJ	90.4
Zhu et al. (2013)	WSJ	90.4
Dyer et al. (2016)	WSJ	91.7
Трансформер	WSJ	91.3
Zhu et al. (2013)	С учителем	91.3
Huang & Harper (2009)	С учителем	91.3
McClosky et al. (2006)	С учителем	92.1
Vinyals & Kaiser el al. (2014)	С учителем	92.1
Трансформер	С учителем	92.7
Luong et al. (2015)	Мульти-задачное	93.0
Dyer et al. (2016)	Генеративное	93.3

Итог

- Архитектура трансформера:
  - Энкодер
  - Декодер
  - Множественное внимание
  - Позиционное кодирование
- Обучение трансформера
  - Оптимизация
  - Регуляризация
- Сравнение с другими моделями
- Вариация гиперпараметров
- Использование трансформера для Constituency Parsing

Источники

- https://arxiv.org/pdf/1706.03762.pdf
- https://arxiv.org/pdf/1412.6980.pdf
- https://kazemnejad.com/blog/transformer_architecture_positional_encoding/
- https://habr.com/ru/post/486358/
- http://jalammar.github.io/illustrated-transformer/