

Полная история семейства GPT: -1, 1, 2, 3...



Котенков Игорь



@stm

Наш путь: 4 модуля



Как появилась GPT-1



Чем занимались
OpenAI до GPT-4



Как мы пришли к GPT-3



ChatGPT и GPT-4

Модуль 2, часть 1. План:

- Данные для обучения GPT-2
- Архитектурные изменения между поколениями
- Метрики на задачах языкового моделирования
- Зачатки промптинга для решения задач

Детали тренировки GPT-2: Датасет



The crawl archive for December 2017 is now available! The archive is located in the `commoncrawl` bucket at [crawl-data/CC-MAIN-2017-51/](#). It contains 2.9 billion web pages and over 240 TiB of uncompressed content.

Data Type	File List	#Files	Total Size Compressed (TiB)
Segments	segment.paths.gz	100	
WARC	warc.paths.gz	80000	61.20
WAT	wat.paths.gz	80000	19.38
WET	wet.paths.gz	80000	8.41
Robots.txt files	robotstxt.paths.gz	80000	0.13
Non-200 responses	non200responses.paths.gz	80000	1.47
URL index files	cc-index.paths.gz	302	0.21
Columnar URL index files	cc-index-table.paths.gz	900	0.24

Детали тренировки GPT-2: Датасет



The crawl archive for December 2017 is now available! The archive is located in the `commoncrawl` bucket at [crawl-data/CC-MAIN-2017-51/](#). It contains 2.9 billion web pages and over 240 TiB of uncompressed content.

Data Type	File List	#Files	Total Size Compressed (TiB)
Segments	segment.paths.gz	100	
WARC	warc.paths.gz	80000	61.20
WAT	wat.paths.gz	80000	19.38
WET	wet.paths.gz	80000	8.41
Robots.txt files	robotstxt.paths.gz	80000	0.13
Non-200 responses	non200responses.paths.gz	80000	1.47
URL index files	cc-index.paths.gz	302	0.21
Columnar URL index files	cc-index-table.paths.gz	900	0.24

большое количество документов,
«содержание которых по
большей части непонятно»

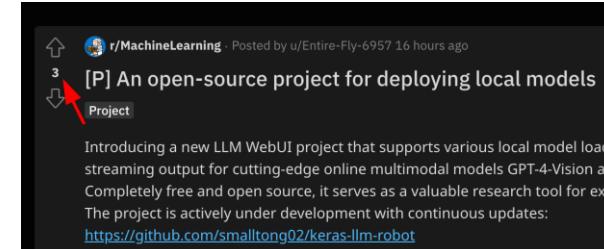
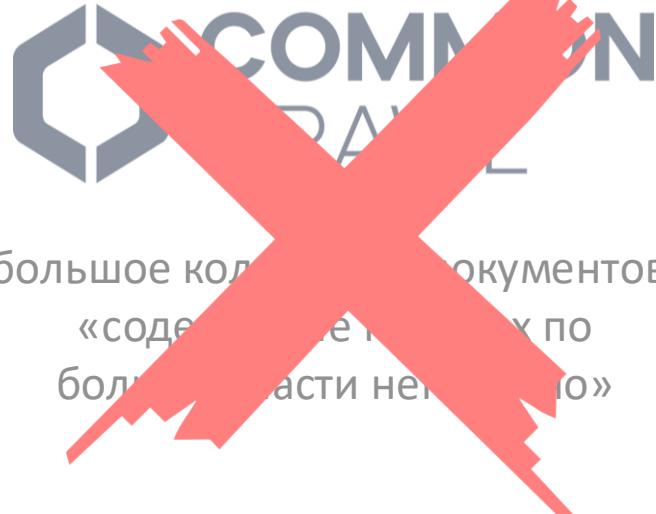
Детали тренировки GPT-2: Датасет



большое количество документов,
«содержащие миллионы по
большинству неподходящих»



Детали тренировки GPT-2: Датасет



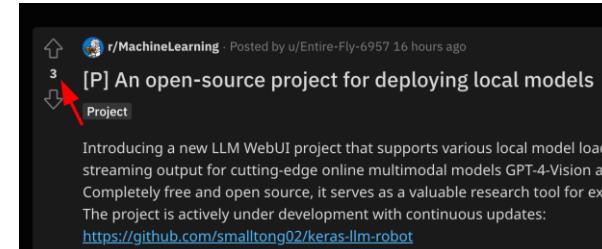
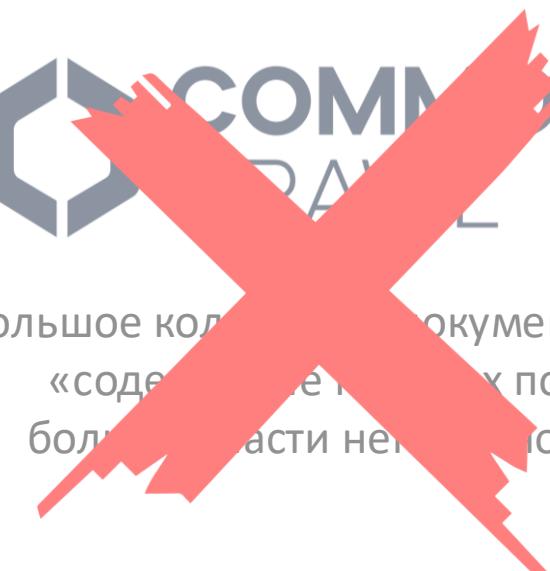
Датасет «WebText»

1. Собрать 45М ссылок
2. Извлечь текст из HTML

Детали тренировки GPT-2: Датасет



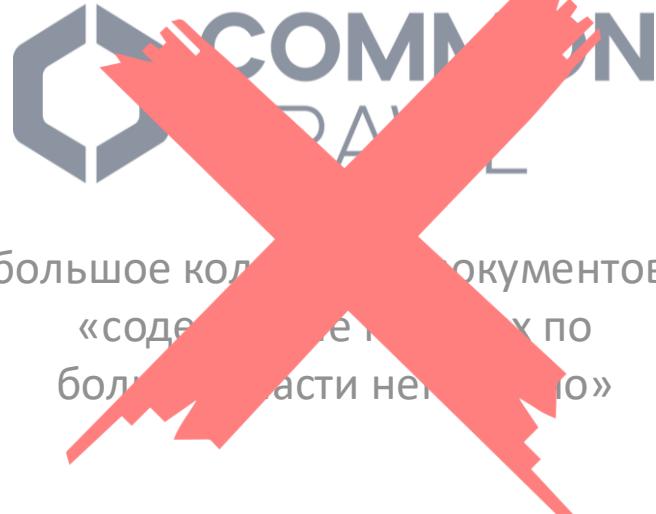
большое количество документов,
«содержащие различные по
большей части нетекстуально»



Датасет «WebText»

1. Собрать 45М ссылок
2. Извлечь текст из HTML
3. Отфильтровать Википедию
4. Отфильтровать всё, кроме английского
5. Дедуплицировать

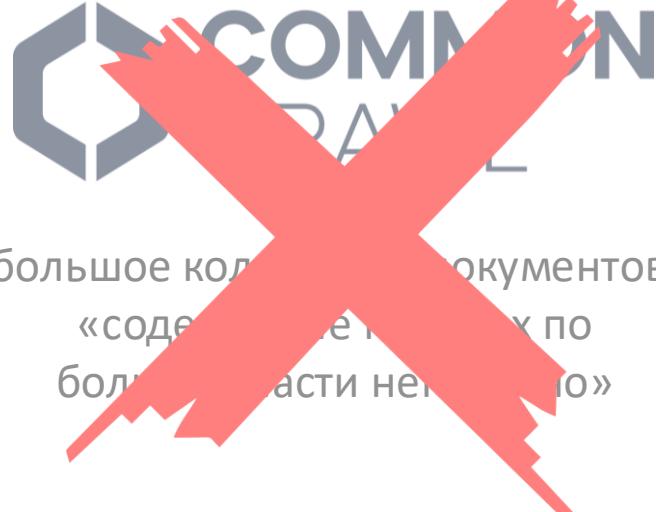
Детали тренировки GPT-2: Датасет



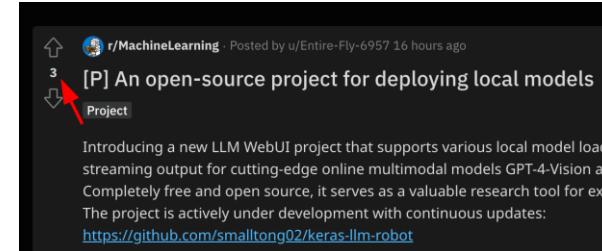
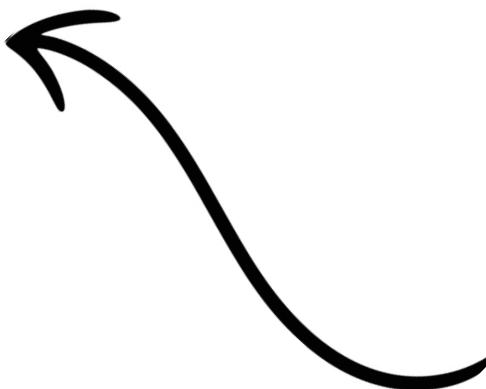
Датасет «WebText»

1. Собрать 45М ссылок
2. Извлечь текст из HTML
3. Отфильтровать Википедию
4. Отфильтровать всё, кроме английского
5. Дедуплицировать
6. Итог: 8М документов и 40GB текста

Детали тренировки GPT-2: Датасет



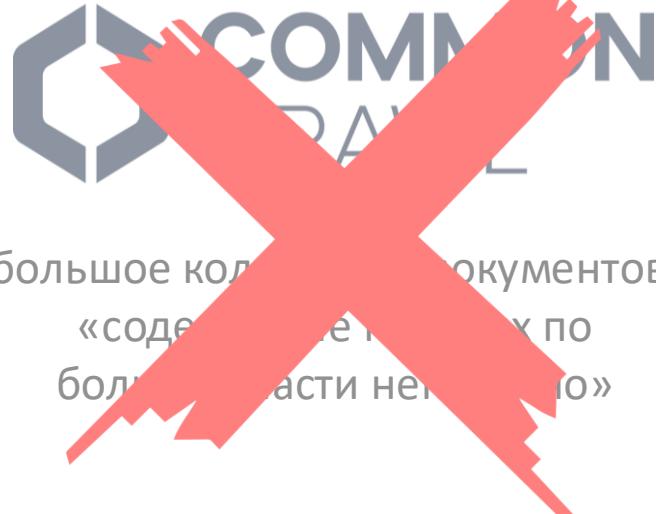
VPE (Unicode code points)



Датасет «WebText»

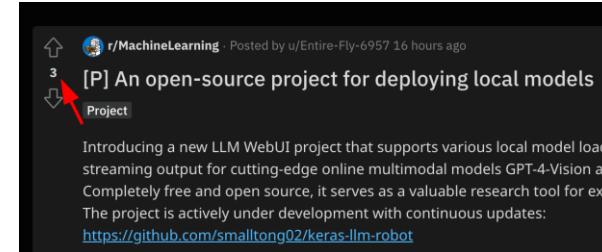
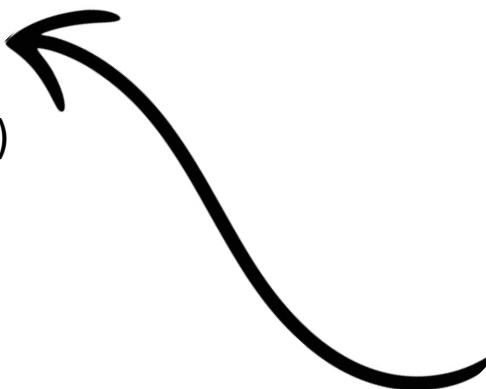
1. Собрать 45М ссылок
2. Извлечь текст из HTML
3. Отфильтровать Википедию
4. Отфильтровать всё, кроме английского
5. Дедуплицировать
6. Итог: 8М документов и 40GB текста

Детали тренировки GPT-2: Датасет



BPE (Unicode code points)

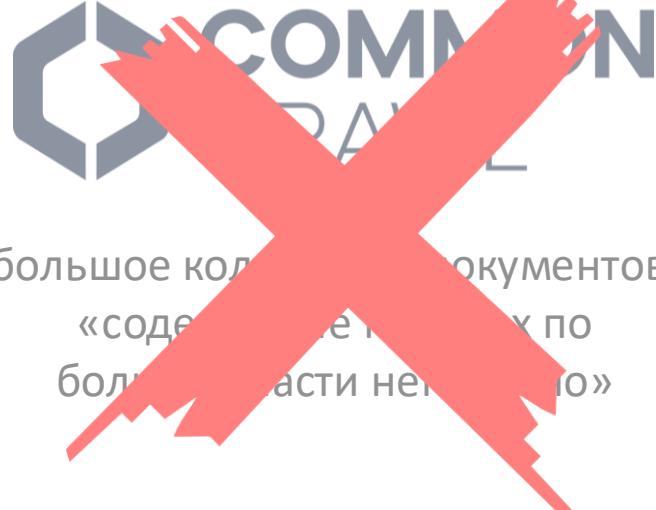
BBPE
(byte-level byte-pair encoding)



Датасет «WebText»

1. Собрать 45М ссылок
2. Извлечь текст из HTML
3. Отфильтровать Википедию
4. Отфильтровать всё, кроме английского
5. Дедуплицировать
6. Итог: 8М документов и 40GB текста

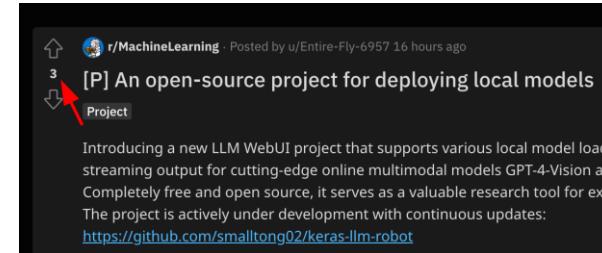
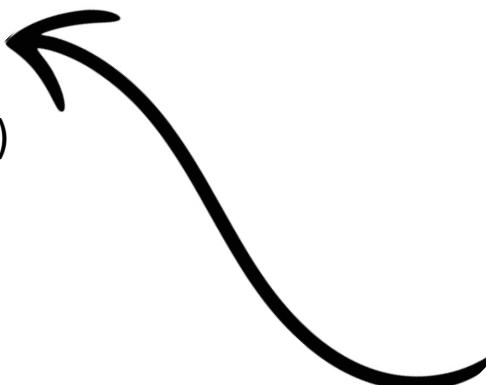
Детали тренировки GPT-2: Датасет



BPE (Unicode code points)

BBPE
(byte-level byte-pair encoding)

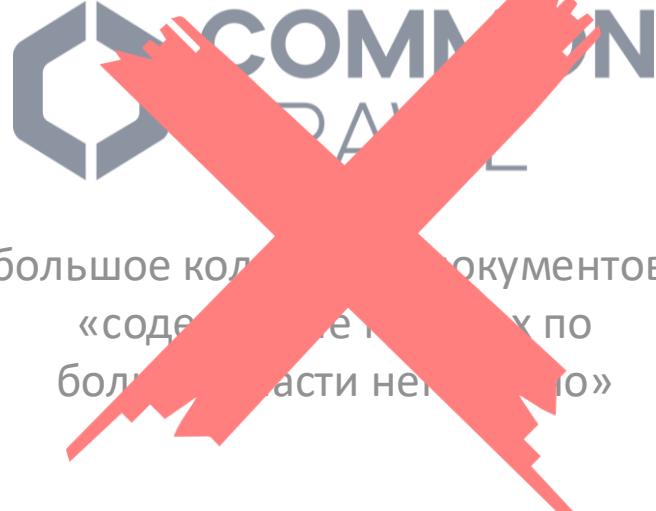
- dog.
- dog!
- dog?



Датасет «WebText»

1. Собрать 45М ссылок
2. Извлечь текст из HTML
3. Отфильтровать Википедию
4. Отфильтровать всё, кроме английского
5. Дедуплицировать
6. Итог: 8М документов и 40GB текста

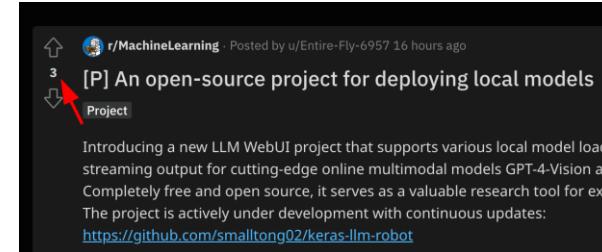
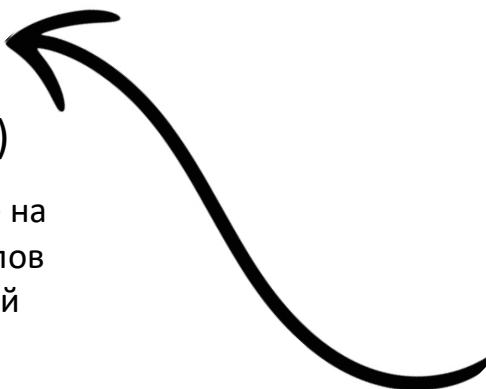
Детали тренировки GPT-2: Датасет



BPE (Unicode code points)

BBPE
(byte-level byte-pair encoding)

- dog. Вводим ограничение на объединение символов из разных категорий (кроме пробелов)
- dog!
- dog?



Датасет «WebText»

1. Собрать 45М ссылок
2. Извлечь текст из HTML
3. Отфильтровать Википедию
4. Отфильтровать всё, кроме английского
5. Дедуплицировать
6. Итог: 8М документов и 40GB текста

Детали тренировки GPT-2: Датасет



большое количество документов,
«содержащие текст по
большинству тематикам»

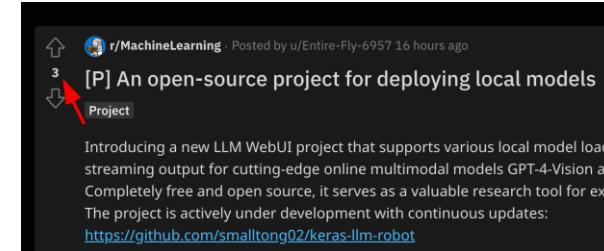
VPE (Unicode code points)

BBPE
(byte-level byte-pair encoding)

- dog. Вводим ограничение на объединение символов из разных категорий (кроме пробелов)
- dog!
- dog?

256 (бит) + 50'000 (мерджей) + 1 (спец. токен)

50257



Датасет «WebText»

1. Собрать 45М ссылок
2. Извлечь текст из HTML
3. Отфильтровать Википедию
4. Отфильтровать всё, кроме английского
5. Дедуплицировать
6. Итог: 8М документов и 40GB текста

Детали тренировки GPT-2: Размер

Parameters	Layers	d_{model}	Размер GPT-1 (датасет больше, мощности не ясны)
117M	12	768	←
345M	24	1024	
762M	36	1280	
1542M	48	1600	

Детали тренировки GPT-2: Размер

Parameters	Layers	d_{model}	
117M	12	768	← Размер GPT-1 (датасет больше, мощности не ясны)
345M	24	1024	
762M	36	1280	
1542M	48	1600	← 100 Volta GPU (V100 или Titan V) x 1 неделя (недотренировано на WebText)

Детали тренировки GPT-2: Размер

Parameters	Layers	d_{model}	
117M	12	768	← Размер GPT-1 (датасет больше, мощности не ясны)
345M	24	1024	
762M	36	1280	
1542M	48	1600	← 100 Volta GPU (V100 или Titan V) x 1 неделя (недотренировано на WebText)

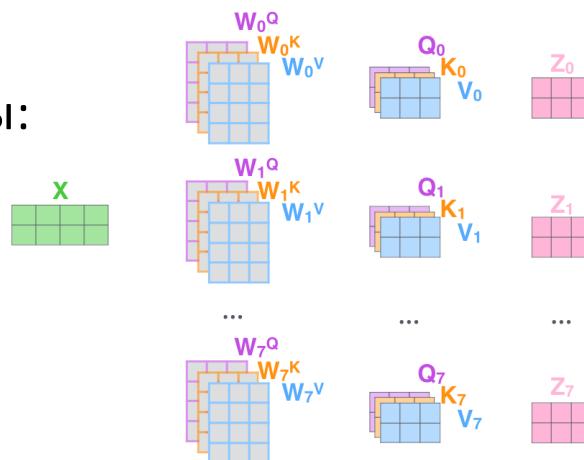
Размерность каждой головы:

$$768 / 12 = 64$$

$$1024 / 16 = 64$$

$$1280 / 20 = 64$$

$$1600 / 25 = 64$$



Детали тренировки GPT-2: Размер

Parameters	Layers	d_{model}	
117M	12	768	← Размер GPT-1 (датасет больше, мощности не ясны)
345M	24	1024	
762M	36	1280	
1542M	48	1600	← 100 Volta GPU (V100 или Titan V) x 1 неделя (недотренировано на WebText)

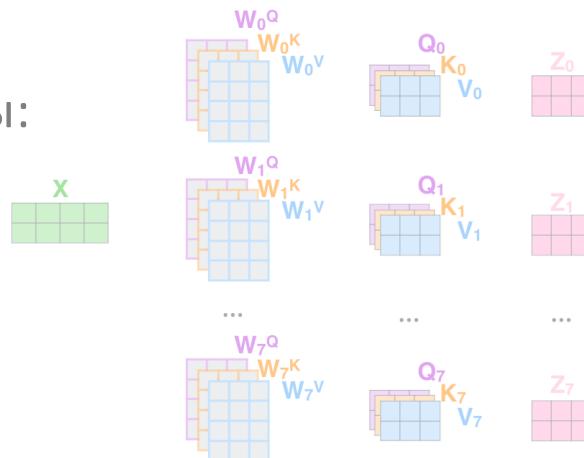
Размерность каждой головы:

$$768 / 12 = 64$$

$$1024 / 16 = 64$$

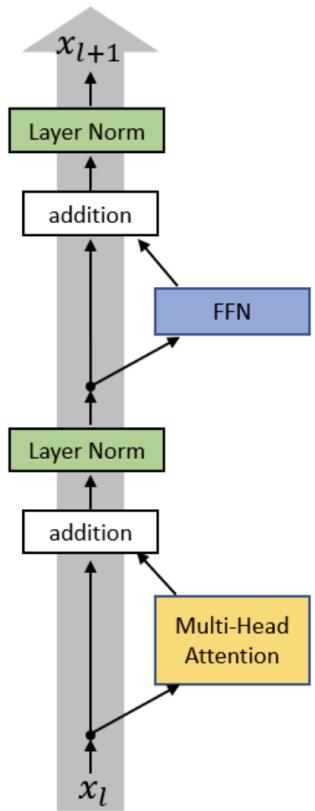
$$1280 / 20 = 64$$

$$1600 / 25 = 64$$



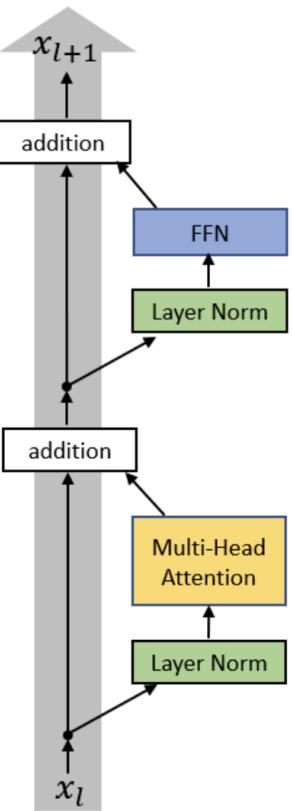
+ расширение контекста до **1024** токенов

Детали тренировки GPT-2: Архитектура



(a)

Post-LN

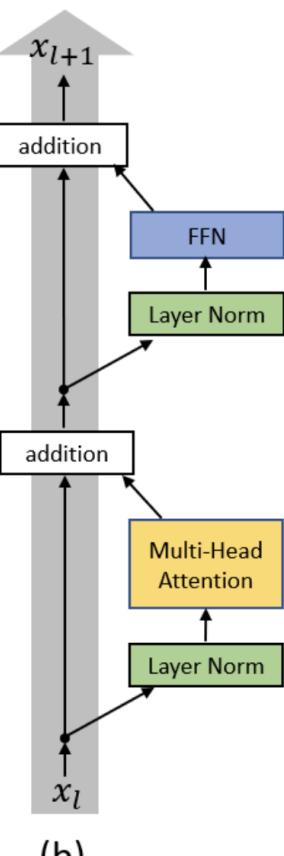
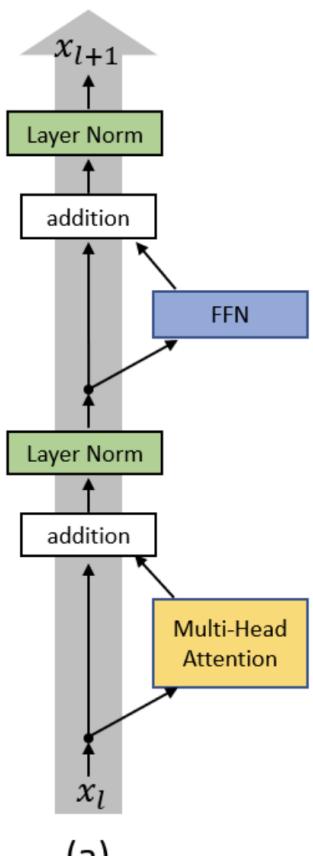


(b)

Pre-LN



Детали тренировки GPT-2: Архитектура



Post-LN

Pre-LN

GPT-1:

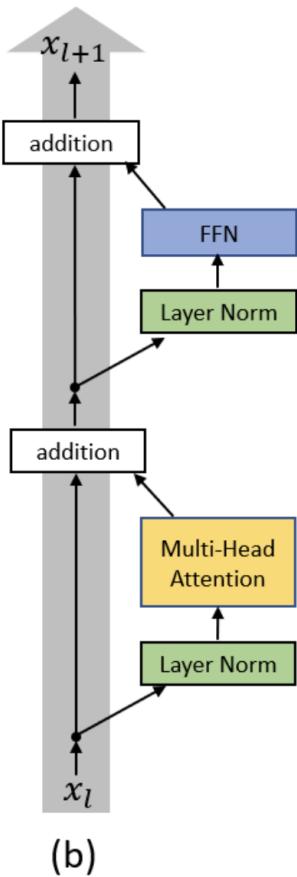
```
def block(x, scope, train=False, scale=False):
    with tf.variable_scope(scope):
        nx = shape_list(x)[-1]
        a = attn(x, 'attn', nx, n_head, train=train, scale=scale)
        n = norm(x+a, 'ln_1')
        m = mlp(n, 'mlp', nx*4, train=train)
        h = norm(n+m, 'ln_2')
    return h
```

GPT-2:

```
def block(x, scope, *, past, hparams):
    with tf.variable_scope(scope):
        nx = x.shape[-1].value
        a, present = attn(norm(x, 'ln_1'), 'attn', nx, past=past, hparams=hparams)
        x = x + a
        m = mlp(norm(x, 'ln_2'), 'mlp', nx*4, hparams=hparams)
        x = x + m
    return x, present
```



Детали тренировки GPT-2: Архитектура



```
[2] model = GPT2LMHeadModel.from_pretrained("gpt2-xl")
tokenizer = GPT2Tokenizer.from_pretrained("gpt2-xl")

prompt = "The future of artificial intelligence is"
inputs = tokenizer(prompt, return_tensors="pt")
```

```
[3] %%time
outputs_base = model.generate(**inputs, max_length=50, num_return_sequences=1)
decoded_output_base = tokenizer.decode(outputs_base[0], skip_special_tokens=True)

Setting `pad_token_id` to `eos_token_id`:50256 for open-end generation.
CPU times: user 39.4 s, sys: 28 ms, total: 39.4 s
Wall time: 40.3 s
```

```
[4] decoded_output_base
'The future of artificial intelligence is bright, but it\'s not going to be a silver bullet for humanity.\n\n"We\'re not going to be a do," said Dr. Andrew Ng,'
```

```
[5] # do the magic!
print(len(model.transformer.h))
del model.transformer.h[-12:]
print(len(model.transformer.h))

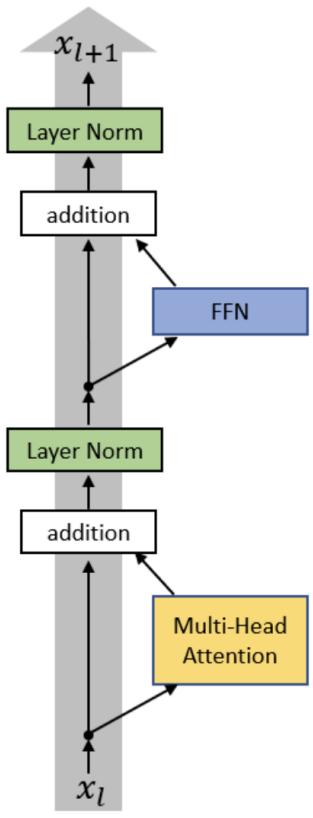
48
36
```

```
[6] %%time
outputs_cut = model.generate(**inputs, max_length=50, num_return_sequences=1)
decoded_output_cut = tokenizer.decode(outputs_cut[0], skip_special_tokens=True)

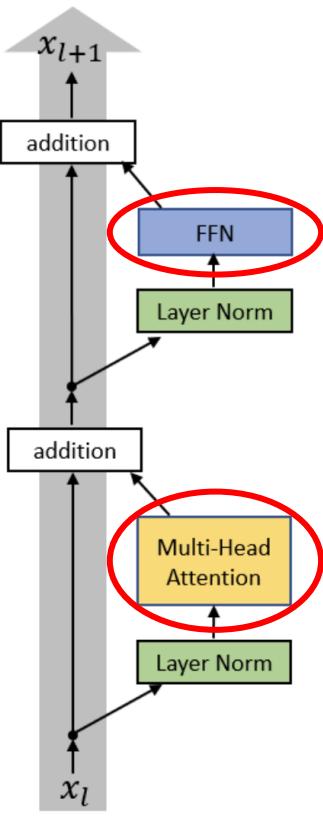
Setting `pad_token_id` to `eos_token_id`:50256 for open-end generation.
CPU times: user 30.1 s, sys: 29.9 ms, total: 30.1 s
Wall time: 31.1 s
```

```
[7] decoded_output_cut
'The future of artificial intelligence is uncertain. Artificial intelligence systems are increasingly sophisticated and capable of sop'
```

Детали тренировки GPT-2: Инициализация



Post-LN



Pre-LN

A modified initialization which accounts for the accumulation on the residual path with model depth is used. We scale the weights of residual layers at initialization by a factor of $1/\sqrt{N}$ where N is the number of residual layers.

```
def scaled_init_method_normal(sigma, num_layers):
    """Init method based on N(0, sigma/sqrt(2*num_layers))."""
    std = sigma / math.sqrt(2.0 * num_layers)

    def init_(tensor):
        return torch.nn.init.normal_(tensor, mean=0.0, std=std)

    return init_
```

GPT-2: Результаты

	LAMBADA (PPL)	LAMBADA (ACC)	CBT-CN (ACC)	CBT-NE (ACC)
SOTA	99.8	59.23	85.7	82.3
117M	35.13	45.99	87.65	83.4
345M	15.60	55.48	92.35	87.1
762M	10.87	60.12	93.45	88.0
1542M	8.63	63.24	93.30	89.05

PPL = перплексия
(меньше — лучше)

ACC = доля правильных угадываний
(больше — лучше)

GPT-2: Результаты

	LAMBADA (PPL)	LAMBADA (ACC)	CBT-CN (ACC)	CBT-NE (ACC)
SOTA	99.8	59.23	85.7	82.3
117M	35.13	45.99	87.65	83.4
345M	15.60	55.48	92.35	87.1
762M	10.87	60.12	93.45	88.0
1542M	8.63	63.24	93.30	89.05

PPL = перплексия
(меньше — лучше)

ACC = доля правильных угадываний
(больше — лучше)

Context: “Why?” “I would have thought you’d find him rather dry,” she said. “I don’t know about that,” said Gabriel.

“He was a great craftsman,” said Heather. “That he was,” said Flannery.

Target sentence: “And Polish, to boot,” said _____.

Target word: Gabriel

Context: Preston had been the last person to wear those chains, and I knew what I’d see and feel if they were slipped onto my skin—the Reaper’s unending hatred of me. I’d felt enough of that emotion already in the amphitheater. I didn’t want to feel anymore. “Don’t put those on me,” I whispered. “Please.”

Target sentence: Sergei looked at me, surprised by my low, raspy please, but he put down the _____.

Target word: chains

Пример заданий датасета LAMBADA

GPT-2: Результаты

	LAMBADA (PPL)	LAMBADA (ACC)	CBT-CN (ACC)	CBT-NE (ACC)	WikiText2 (PPL)	PTB (PPL)	enwik8 (BPB)	text8 (BPC)	WikiText103 (PPL)	1BW (PPL)
SOTA	99.8	59.23	85.7	82.3	39.14	46.54	0.99	1.08	18.3	21.8
117M	35.13	45.99	87.65	83.4	29.41	65.85	1.16	1.17	37.50	75.20
345M	15.60	55.48	92.35	87.1	22.76	47.33	1.01	1.06	26.37	55.72
762M	10.87	60.12	93.45	88.0	19.93	40.31	0.97	1.02	22.05	44.575
1542M	8.63	63.24	93.30	89.05	18.34	35.76	0.93	0.98	17.48	42.16

PPL = перплексия
(меньше — лучше)

ACC = доля правильных угадываний
(больше — лучше)

BPB = Bits-per-byte
(меньше — лучше)

Context: “Why?” “I would have thought you’d find him rather dry,” she said. “I don’t know about that,” said Gabriel.
“He was a great craftsman,” said Heather. “That he was,” said Flannery.

Target sentence: “And Polish, to boot,” said _____.

Target word: Gabriel

Context: Preston had been the last person to wear those chains, and I knew what I’d see and feel if they were slipped onto my skin—the Reaper’s unending hatred of me. I’d felt enough of that emotion already in the amphitheater. I didn’t want to feel anymore. “Don’t put those on me,” I whispered. “Please.”

Target sentence: Sergei looked at me, surprised by my low, raspy please, but he put down the _____.

Target word: chains

Пример заданий датасета LAMBADA

GPT-2: Результаты

	LAMBADA (PPL)	LAMBADA (ACC)	CBT-CN (ACC)	CBT-NE (ACC)	WikiText2 (PPL)	PTB (PPL)	enwik8 (BPB)	text8 (BPC)	WikiText103 (PPL)	1BW (PPL)
SOTA	99.8	59.23	85.7	82.3	39.14	46.54	0.99	1.08	18.3	21.8
117M	35.13	45.99	87.65	83.4	29.41	65.85	1.16	1.17	37.50	75.20
345M	15.60	55.48	92.35	87.1	22.76	47.33	1.01	1.06	26.37	55.72
762M	10.87	60.12	93.45	88.0	19.93	40.31	0.97	1.02	22.05	44.575
1542M	8.63	63.24	93.30	89.05	18.34	35.76	0.93	0.98	17.48	42.16

PPL = перплексия
(меньше — лучше)

ACC = доля правильных угадываний
(больше — лучше)

BPB = Bits-per-byte
(меньше — лучше)

	PTB	WikiText-2	enwik8	text8	Wikitext-103	1BW
Dataset train	2.67%	0.66%	7.50%	2.34%	9.09%	13.19%
WebText train	0.88%	1.63%	6.31%	3.94%	2.42%	3.75%

Table 6. Percentage of test set 8 grams overlapping with training sets.

Замер пересечения датасетов с
использованием фильтров Блума

GPT-2: Результаты

	LAMBADA (PPL)	LAMBADA (ACC)	CBT-CN (ACC)	CBT-NE (ACC)	WikiText2 (PPL)	PTB (PPL)	enwik8 (BPB)	text8 (BPC)	WikiText103 (PPL)	1BW (PPL)
SOTA	99.8	59.23	85.7	82.3	39.14	46.54	0.99	1.08	18.3	21.8
117M	35.13	45.99	87.65	83.4	29.41	65.85	1.16	1.17	37.50	75.20
345M	15.60	55.48	92.35	87.1	22.76	47.33	1.01	1.06	26.37	55.72
762M	10.87	60.12	93.45	88.0	19.93	40.31	0.97	1.02	22.05	44.575
1542M	8.63	63.24	93.30	89.05	18.34	35.76	0.93	0.98	17.48	42.16

PPL = перплексия
(меньше — лучше)

ACC = доля правильных угадываний
(больше — лучше)

BPB = Bits-per-byte
(меньше — лучше)

LAMBADA:

- 1.2% пересечения
- Пересчёт метрик с исключением общих примеров даёт:
 - PPL = 8.7 (**+0.07**)
 - ACC = 62.9 (**-0.34**)

	PTB	WikiText-2	enwik8	text8	Wikitext-103	1BW
Dataset train	2.67%	0.66%	7.50%	2.34%	9.09%	13.19%
WebText train	0.88%	1.63%	6.31%	3.94%	2.42%	3.75%

Table 6. Percentage of test set 8 grams overlapping with training sets.

Замер пересечения датасетов с использованием фильтров Блума

GPT-2: Результаты

Примеры из Winograd Schema Challenge:

- The trophy doesn't fit in the brown suitcase because it's too big. What is too big?

Answer 0: the trophy

Answer 1: the suitcase

- Joan made sure to thank Susan for all the help she had given. Who had given the help?

Answer 0: Joan

Answer 1: Susan

GPT-2: Результаты

Примеры из Winograd Schema Challenge:

- The trophy doesn't fit in the brown suitcase because it's too big. What is too big?

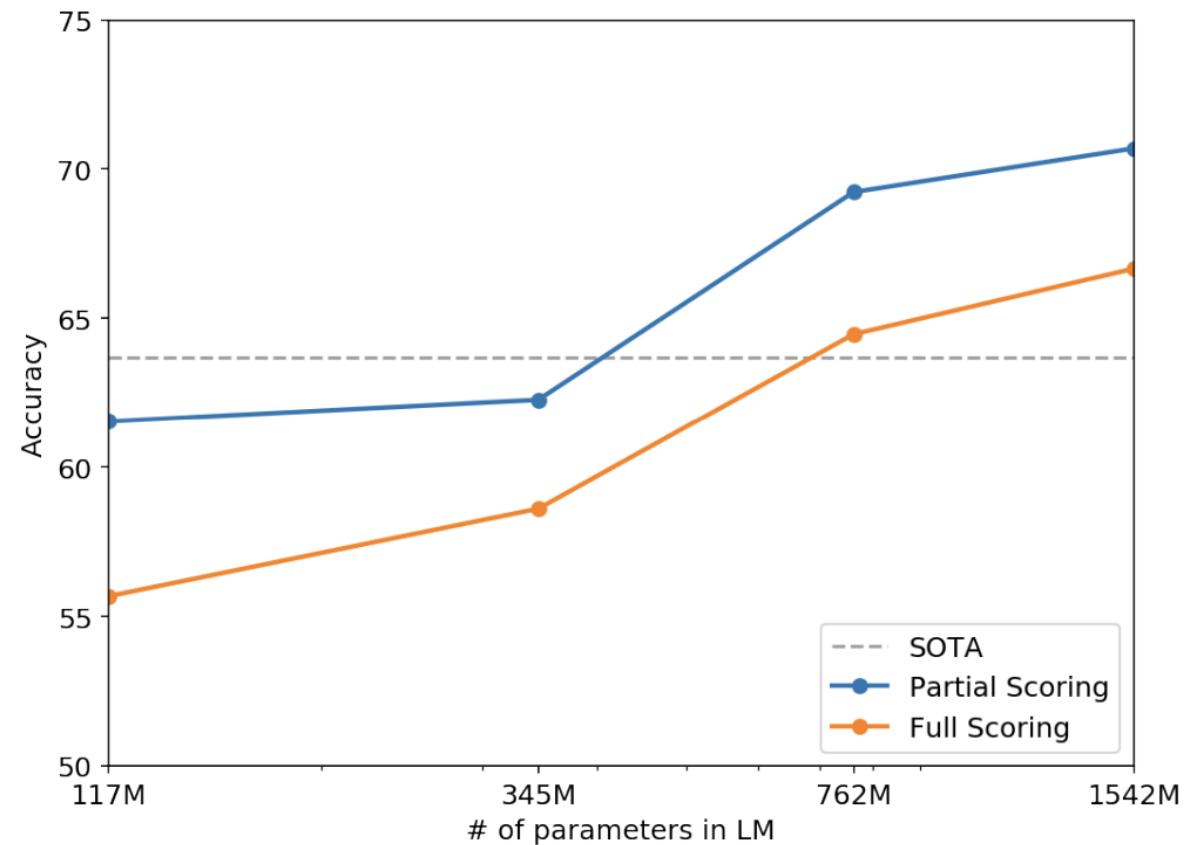
Answer 0: the trophy

Answer 1: the suitcase

- Joan made sure to thank Susan for all the help she had given. Who had given the help?

Answer 0: Joan

Answer 1: Susan



(в датасете всего 273 примера)

GPT-2: Результаты

Примеры из Winograd Schema Challenge:

- The trophy doesn't fit in the brown suitcase because it's too big. What is too big?

Answer 0: the trophy

Answer 1: the suitcase

- Joan made sure to thank Susan for all the help she had given. Who had given the help?

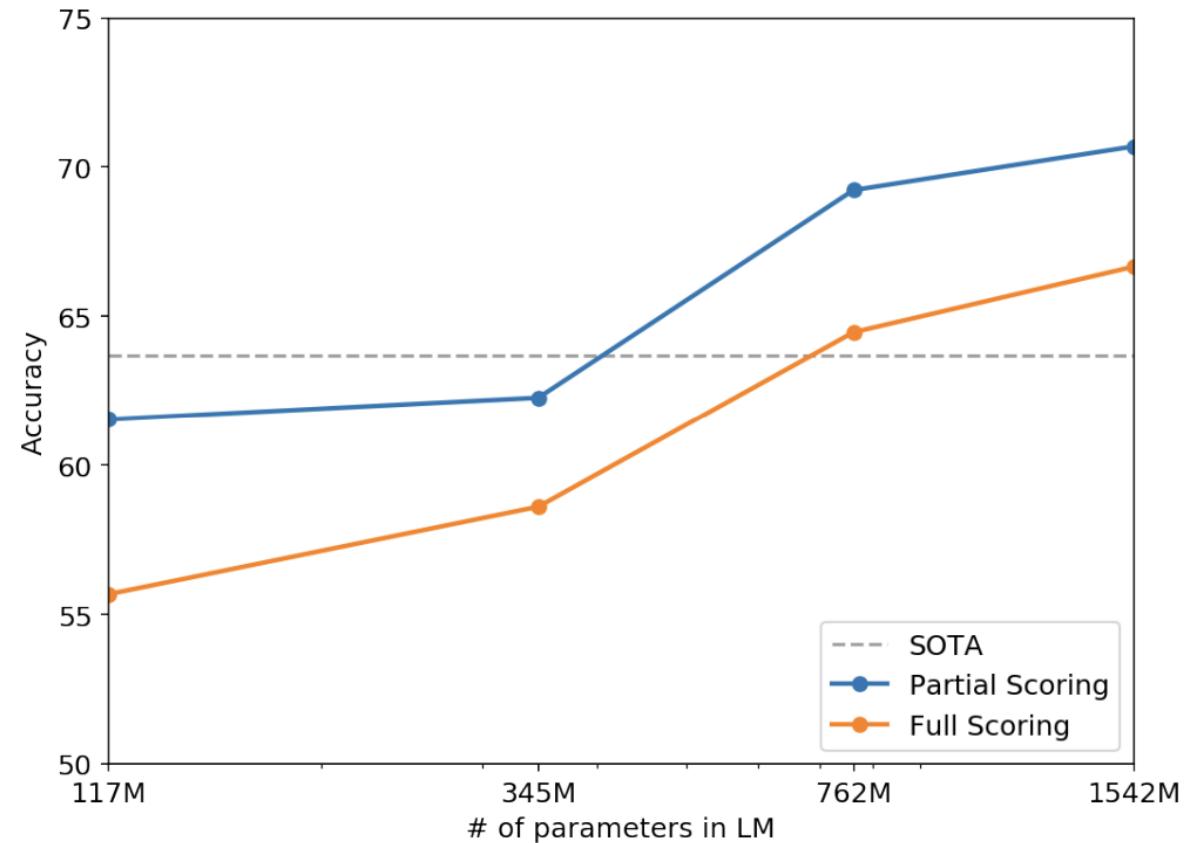
Answer 0: Joan

Answer 1: Susan

$$P(y_1, y_2, \dots, y_n) = P(y_1) \cdot P(y_2|y_1) \cdot P(y_3|y_1, y_2) \cdots$$

$S_{full} = P(\text{The trophy doesn't fit in the brown suitcase because } \underline{\text{it's too big}})$

$S_{partial} = P(\underline{\text{the suitcase is too big}} | \text{The trophy doesn't fit in the brown suitcase because})$



(в датасете всего 273 примера)

GPT-2: Неожиданные Результаты

	R-1	R-2	R-L	R-AVG
Bottom-Up Sum	41.22	18.68	38.34	32.75
Lede-3	40.38	17.66	36.62	31.55
Seq2Seq + Attn	31.33	11.81	28.83	23.99
GPT-2 TL; DR:	29.34	8.27	26.58	<u>21.40</u>
Random-3	28.78	8.63	25.52	20.98
GPT-2 no hint	21.58	4.03	19.47	<u>15.03</u>

После подачи новости:

- (опционально) добавляем «TL;DR: »
 - Встречается ~100k раз в датасете (+ «in summary»)
- генерируем 100 токенов (top-k sampling, k=2)
- берём первые 3 предложения

Обычно саммари концентрируются на последних частях новости, а также путает детали :(

GPT-2: Неожиданные Результаты

	R-1	R-2	R-L	R-AVG
Bottom-Up Sum	41.22	18.68	38.34	32.75
Lede-3	40.38	17.66	36.62	31.55
Seq2Seq + Attn	31.33	11.81	28.83	23.99
GPT-2 TL;DR:	29.34	8.27	26.58	<u>21.40</u>
Random-3	28.78	8.63	25.52	<u>20.98</u>
GPT-2 no hint	21.58	4.03	19.47	<u>15.03</u>

{english sentence_1} = {french sentence_2}
{english sentence_2} = {french sentence_2}
...
{english sentence_n} =

После подачи новости:

- (опционально) добавляем «TL;DR»
 - Встречается ~100k раз в датасете (+ «in summary»)
- генерируем 100 токенов (top-k sampling, k=2)
- берём первые 3 предложения

- WMT-14 English-French: 5 BLEU
- WMT-14 French-English: 11.5 BLEU (UL SotA 33.5 BLEU)
- В обоих случаях — greedy decoding

Обычно саммари концентрируются на последних частях новости, а также путает детали :(

GPT-2: Неожиданные Результаты

	R-1	R-2	R-L	R-AVG
Bottom-Up Sum	41.22	18.68	38.34	32.75
Lede-3	40.38	17.66	36.62	31.55
Seq2Seq + Attn	31.33	11.81	28.83	23.99
GPT-2 TL;DR:	29.34	8.27	26.58	<u>21.40</u>
Random-3	28.78	8.63	25.52	<u>20.98</u>
GPT-2 no hint	21.58	4.03	19.47	<u>15.03</u>

{english sentence_1} = {french sentence_2}
{english sentence_2} = {french sentence_2}
...
{english sentence_n} =

После подачи новости:

- (опционально) добавляем «TL;DR»
 - Встречается ~100k раз в датасете (+ «in summary»)
- генерируем 100 токенов (top-k sampling, k=2)
- берём первые 3 предложения

- WMT-14 English-French: 5 BLEU
- WMT-14 French-English: 11.5 BLEU (UL SotA 33.5 BLEU)
- В обоих случаях — greedy decoding
- Модель видела лишь **10МБ французского текста**
 - Это в **~500 раз меньше** корпуса для UL-систем!

Обычно саммари концентрируются на последних частях новости, а также путает детали :(

GPT-2: Неожиданные Результаты

	R-1	R-2	R-L	R-AVG
Bottom-Up Sum	41.22	18.68	38.34	32.75
Lede-3	40.38	17.66	36.62	31.55
Seq2Seq + Attn	31.33	11.81	28.83	23.99
GPT-2 TL;DR:	29.34	8.27	26.58	<u>21.40</u>
Random-3	28.78	8.63	25.52	<u>20.98</u>
GPT-2 no hint	21.58	4.03	19.47	<u>15.03</u>

{english sentence_1} = {french sentence_2}
{english sentence_2} = {french sentence_2}
...
{english sentence_n} =

После подачи новости:

- (опционально) добавляем «TL;DR»
 - Встречается ~100k раз в датасете (+ «in summary»)
- генерируем 100 токенов (top-k sampling, k=2)
- берём первые 3 предложения

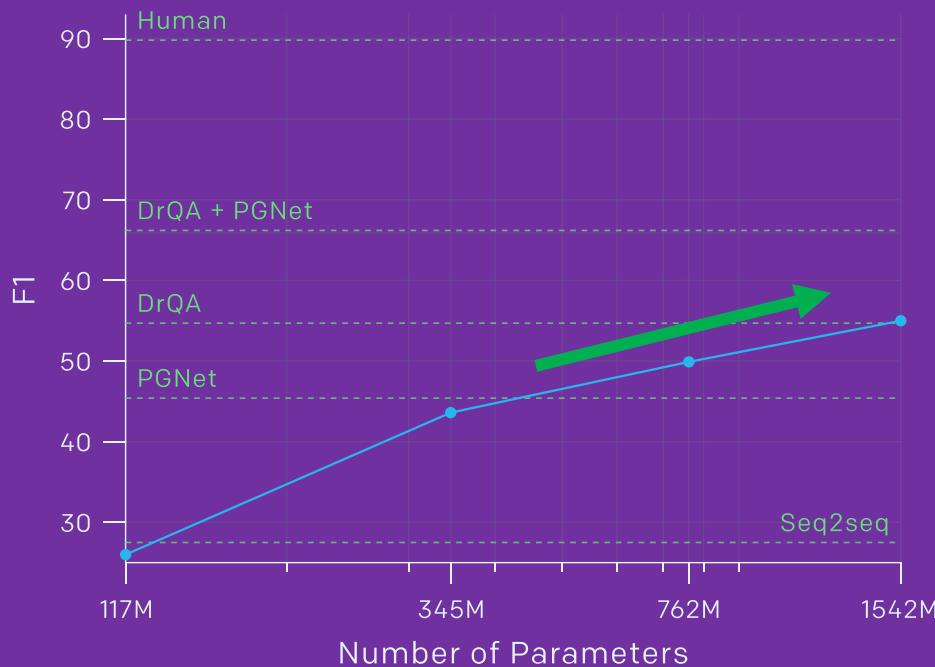
- WMT-14 English-French: 5 BLEU
- WMT-14 French-English: 11.5 BLEU (UL SotA 33.5 BLEU)
- В обоих случаях — greedy decoding
- Модель видела лишь 10МБ французского текста
 - Это в ~500 раз меньше корпуса для UL-систем!

Обычно саммари концентрируются на последних частях новости, а также путает детали :(

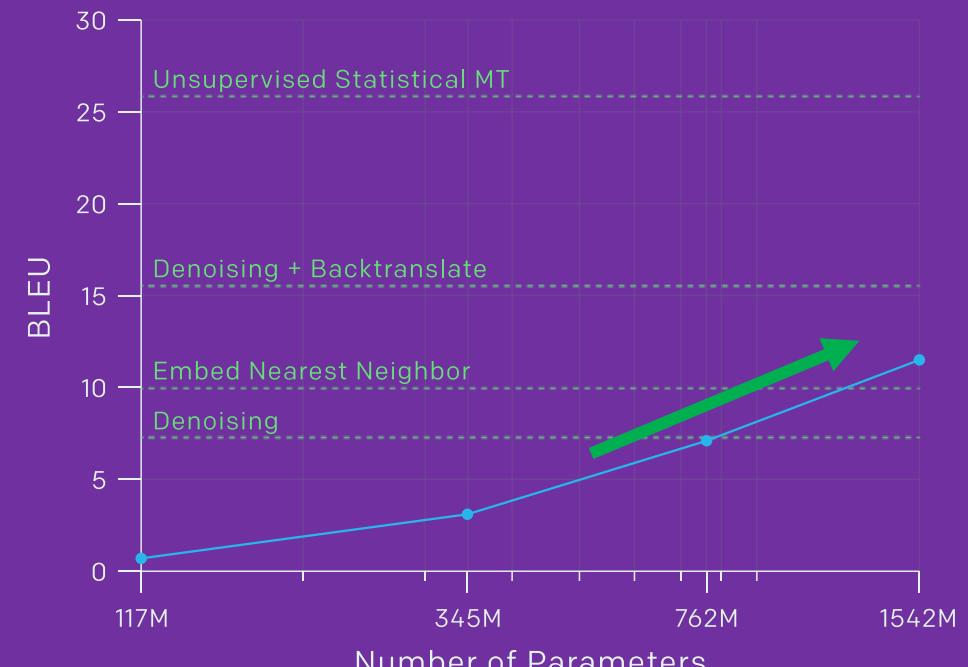
Ого, ZS/FS заработал!



GPT-2: Неожиданные Результаты



Качество на COQA
(вопрос-ответ по данному контексту)



Качество на WMT-14 Fr-En
(перевод)

ИТОГИ

- GPT-2 существенно отличается лишь размером модели и датасета
- Архитектурные изменения минимальны
 - pre-LN, инициализация, размер словаря, окно контекста

ИТОГИ

- GPT-2 существенно отличается лишь размером модели и датасета
- Архитектурные изменения минимальны
 - pre-LN, инициализация, размер словаря, окно контекста
- На задачах моделирования языка языковая модель...выигрывает
 - И при этом датасет не дублирует тестовые выборки!
- На задачах Commonsense Reasoning поставлен SotA результат

Итоги

- GPT-2 существенно отличается лишь размером модели и датасета
- Архитектурные изменения минимальны
 - pre-LN, инициализация, размер словаря, окно контекста
- На задачах моделирования языка языковая модель...выигрывает
 - И при этом датасет не дублирует тестовые выборки!
- На задачах Commonsense Reasoning поставлен SotA результат
- На нетривиальных задачах заметен интересный прогресс

Directed by
IGOR D. KOTENKOV

Полная история семейства GPT: -1, 1, 2, 3...



Котенков Игорь



@stm

Модуль 2, часть 2. План:

- Хорошо ли GPT-2 генерировала текст?
- Опубликовали ли OpenAI модель?
- Можно ли надёжно обнаруживать сгенерированный текст?

Генеративные навыки GPT-2

System Prompt (human-written)

In a shocking finding, scientist discovered a herd of unicorns living in a remote, previously unexplored valley, in the Andes Mountains. Even more surprising to the researchers was the fact that the unicorns spoke perfect English.

Model Completion (machine-written, 10 tries)

The scientist named the population, after their distinctive horn, Ovid's Unicorn. These four-horned, silver-white unicorns were previously unknown to science.

Now, after almost two centuries, the mystery of what sparked this odd phenomenon is finally solved.

Dr. Jorge Pérez, an evolutionary biologist from the University of La Paz, and several companions, were exploring the Andes Mountains when they found a small valley, with no other animals or humans. Pérez noticed that the valley had what appeared to be a natural fountain, surrounded by two peaks of rock and silver snow.

Pérez and the others then ventured further into the valley. "By the time we reached the top of one peak, the water looked blue, with some crystals on top," said Pérez.

Pérez and his friends were astonished to see the unicorn herd. These creatures could be seen from the air without having to move too much to see them - they were so close they could touch their horns.

Генеративные навыки GPT-2

System Prompt (human-written)

In a shocking finding, scientist discovered a herd of unicorns living in a remote, previously unexplored valley, in the Andes Mountains. Even more surprising to the researchers was the fact that the unicorns spoke perfect English.

Model Completion (machine-written, 10 tries)

The scientist named the population, after their distinctive horn, Ovid's Unicorn. These four-horned, silver-white unicorns were previously unknown to science.

Now, after almost two centuries, the mystery of what sparked this odd phenomenon is finally solved.

Dr. Jorge Pérez, an evolutionary biologist from the University of La Paz, and several companions, were exploring the Andes Mountains when they found a small valley, with no other animals or humans. Pérez noticed that the valley had what appeared to be a natural fountain, surrounded by two peaks of rock and silver snow.

Pérez and the others then ventured further into the valley. "By the time we reached the top of one peak, the water looked blue, with some crystals on top," said Pérez.

Pérez and his friends were astonished to see the unicorn herd. These creatures could be seen from the air without having to move too much to see them - they were so close they could touch their horns.

System Prompt (human-written)

We've trained a large language model called GPT-2 that generates realistic paragraphs of text, while also exhibiting zero shot generalization on tasks like machine translation, question answering, reading comprehension, and summarization - problems usually approached by using training datasets and models designed explicitly for these tasks.

A typical approach to language modeling is to learn the following task: predict the next word, given all of the previous words within some text. Last year, OpenAI's Generative Pre-trained Transformer (GPT) showed that language models trained on large amounts of data can be fine-tuned to specific tasks to achieve high performance. GPT-2 shows that much larger language models trained on a more diverse dataset derived from the internet begin to learn these NLP tasks without needing task-specific training data, instead learning from examples the system derives from the raw text. These systems also display a substantial qualitative jump in the realism and coherence of generated text.

Model Completion (machine-written, 10 tries)

Here you can see the most recent progress with Generative Pre-trained Transformer:

Figure 1: Generative Pre-trained Transformer training on several texts.

We are now preparing a collection of datasets for translation and machine translation in our language model. We will be using one of the large number of text samples provided by The New York Times.

We believe this project is the first step in the direction of developing large NLP systems without task-specific training data. That is, we are developing a machine language system in the generative style with no explicit rules for producing text.

Генеративные навыки GPT-2

Open Future | Guest comment: Art Intell

How to respond to climate change, if you are an algorithm

We ran our youth essay question through an artificial-intelligence system to produce an essay

«Какие фундаментальные экономические и политические изменения необходимы для эффективного реагирования на изменение климата?»



Oct 1st 2019

Share

Генеративные навыки GPT-2

Open Future | Guest comment: Art Intell

How to respond to climate change, if you are an algorithm

We ran our youth essay question through an artificial-intelligence system to produce an essay



Share

Oct 1st 2019

*«Какие фундаментальные
экономические и политические
изменения необходимы для
эффективного реагирования
на изменение климата?»*

Judge 1: No—It does not seem to offer a practical solution. The tone is hypothetical and abstract.

Judge 2: No—Doesn't get to the point quick enough; point isn't novel, too vague, excessive, high number of rhetorical questions.

Judge 3: No—The writing is quite good but the essay doesn't get to the point quickly enough, and in fact the single proposed change is not clearly defined at all.

Judge 4: Maybe—The solution ("alternative economy") isn't clear; a lot of giving context and asking questions but not a lot of answering them.

Judge 5: Maybe—It is strongly worded and backs up claims with evidence, but the idea is not incredibly original.

Judge 6: No—The essay does not fundamentally answer the question nor present a single novel idea, is not strongly argued and is not particularly well written/structured. In addition, I do not think it shows a strong understanding of existing climate policy nor of the scientific literature coming out of the IPCC.

Генеративные навыки GPT-2



BUSINESS INSIDER

An Elon Musk-backed AI firm is keeping a text generating tool under wraps amid fears it's too dangerous

Isobel Asher Hamilton Feb 15, 2019, 6:48 PM GMT+8

Share

Save

Злонамеренное использование:

- создание вводящих в заблуждение новостных статей
- подделка личности других людей в интернете
- массовое создание оскорбительного/фейкового контента
- загрязнение соц. сетей сгенерированным контентом
- автоматизация спама/фишинга

Генеративные навыки GPT-2



BUSINESS INSIDER

An Elon Musk-backed AI firm is keeping a text generating tool under wraps amid fears it's too dangerous

Isobel Asher Hamilton Feb 15, 2019, 6:48 PM GMT+8

Share

Save

Злонамеренное использование:

- создание вводящих в заблуждение новостных статей
- подделка личности других людей в интернете
- массовое создание оскорбительного/фейкового контента
- загрязнение соц. сетей сгенерированным контентом
- автоматизация спама/фишинга

Стратегия поэтапного релиза:

- сначала доступна самая маленькая модель
- нет данных и кода обучения, есть код генерации
- главная цель — создать прецедент и обсуждение

Генеративные навыки GPT-2



BUSINESS INSIDER

An Elon Musk-backed AI firm is keeping a text generating tool under wraps amid fears it's too dangerous

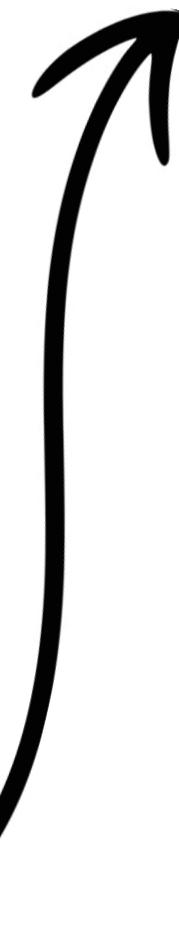
Isobel Asher Hamilton Feb 15, 2019, 6:48 PM GMT+8

Share

Save

Злонамеренное использование:

- создание вводящих в заблуждение новостных статей
- подделка личности других людей в интернете
- массовое создание оскорбительного/фейкового контента
- загрязнение соц. сетей сгенерированным контентом
- автоматизация спама/фишинга



В следующие 6 месяцев:

- опубликовано ещё 2 модели (345М и 774М)
- опубликован датасет

Стратегия поэтапного релиза:

- сначала доступна самая маленькая модель
- нет данных и кода обучения, есть код генерации
- главная цель — создать прецедент и обсуждение

Генеративные навыки GPT-2



BUSINESS INSIDER

An Elon Musk-backed AI firm is keeping a text generating tool under wraps amid fears it's too dangerous

Isobel Asher Hamilton Feb 15, 2019, 6:48 PM GMT+8

Share

Save

Злонамеренное использование:

- создание вводящих в заблуждение новостных статей
- подделка личности других людей в интернете
- массовое создание оскорбительного/фейкового контента
- загрязнение соц. сетей сгенерированным контентом
- автоматизация спама/фишинга

Стратегия поэтапного релиза:

- сначала доступна самая маленькая модель
- нет данных и кода обучения, есть код генерации
- главная цель — создать прецедент и обсуждение

В следующие 6 месяцев:

- опубликовано ещё 2 модели (345M и 774M)
- опубликован датасет
- опубликованы сотни тысяч генераций
- опубликован детектор (80-96% Accuracy)

Transferred model accuracy (nucleus samples)

Trained on ↓	Tested on →			
	Small (124M)	Medium (355M)	Large (774M)	XL (1.5B)
Small (124M)	99.3%	96.6%	90.9%	79.3%
Medium (355M)	99.0%	98.5%	96.9%	91.8%
Large (774M)	98.4%	97.9%	97.9%	95.7%
XL (1.5B)	96.9%	96.7%	96.6%	96.0%

Генеративные навыки GPT-2



BUSINESS INSIDER

An Elon Musk-backed AI firm is keeping a text generating tool under wraps amid fears it's too dangerous

Isobel Asher Hamilton Feb 15, 2019, 6:48 PM GMT+8

Share

Save

Злонамеренное использование:

- создание вводящих в заблуждение новостных статей
- подделка личности других людей в интернете
- массовое создание оскорбительного/фейкового контента
- загрязнение соц. сетей сгенерированным контентом
- автоматизация спама/фишинга

Стратегия поэтапного релиза:

- сначала доступна самая маленькая модель
- нет данных и кода обучения, есть код генерации
- главная цель — создать прецедент и обсуждение

В следующие 6 месяцев:

- опубликовано ещё 2 модели (345M и 774M)
- опубликован датасет
- опубликованы сотни тысяч генераций
- опубликован детектор (80-96% Accuracy)
- заключены партнёрства с 6+ организациями
 - им даже выдали доступ к 1.5B!
- опубликовано 2+ статьи на 100+ страниц

Transferred model accuracy (nucleus samples)

Trained on ↓	Tested on →			
	Small (124M)	Medium (355M)	Large (774M)	XL (1.5B)
Small (124M)	99.3%	96.6%	90.9%	79.3%
Medium (355M)	99.0%	98.5%	96.9%	91.8%
Large (774M)	98.4%	97.9%	97.9%	95.7%
XL (1.5B)	96.9%	96.7%	96.6%	96.0%

Что. Было. Дальше.



NVIDIA ADLR

GPT-2 Evaluation

To analyze the performance of training large language models, we compute perplexity on the [WikiText-103 dataset](#) and cloze-style prediction accuracy on the [LAMBADA dataset](#). Earlier we made sure to remove WikiText test set content to avoid leakage. Our resulting dataset has a WikiText 8-gram overlap of 10% which is similar to the 9% 8-gram overlap between the WikiText-103 train and test sets. As expected, the WikiText perplexity decreases and LAMBADA accuracy increases with the growth of the model size (Table 3). At a respective WikiText perplexity of 12.68 and 10.81 both our 2.5B and 8.3B models surpass the previous state of the art perplexity of 16.43 set by [Krause et. al.](#) Our models achieve 61.52% and 66.51% accuracy on LAMBADA even without any stopword filtration, surpassing [Radford et. al.](#) We describe our evaluation methodologies below; however, more details are available in [our github repository](#).

Model Size	Number of layers	Hidden Size	Attention heads	Wikitext (Perplexity .)	Lambada (Accuracy %)
345 M	24	1024	16	19.22	46.26
2.5 B	54	1920	20	12.68	61.52
8.3 B	72	3072	24	10.81	66.51
SOTA	—	—	—	16.43*	63.24**

Table 3: Zero-shot evaluation results for WikiText perplexity (lower is better) and LAMBADA

Реимплементация и масштабирование

Что. Было. Дальше.



NVIDIA ADLR

GPT-2 Evaluation

To analyze the performance of training large language models, we compute perplexity on the [WikiText-103 dataset](#) and cloze-style prediction accuracy on the [LAMBADA dataset](#). Earlier we made sure to remove WikiText test set content to avoid leakage. Our resulting dataset has a WikiText 8-gram overlap of 10% which is similar to the 9% 8-gram overlap between the WikiText-103 train and test sets. As expected, the WikiText perplexity decreases and LAMBADA accuracy increases with the growth of the model size (Table 3). At a respective WikiText perplexity of 12.68 and 10.81 both our 2.5B and 8.3B models surpass the previous state of the art perplexity of 16.43 set by [Krause et. al.](#) Our models achieve 61.52% and 66.51% accuracy on LAMBADA even without any stopword filtration, surpassing [Radford et. al.](#) We describe our evaluation methodologies below; however, more details are available in [our github repository](#).

Model Size	Number of layers	Hidden Size	Attention heads	Wikitext (Perplexity .)	Lambada (Accuracy %)
345 M	24	1024	16	19.22	46.26
2.5 B	54	1920	20	12.68	61.52
8.3 B	72	3072	24	10.81	66.51
SOTA	—	—	—	16.43*	63.24**

Table 3: Zero-shot evaluation results for WikiText perplexity (lower is better) and LAMBADA

Реимплементация и масштабирование

Ethical analysis of the open-sourcing of a state-of-the-art conversational AI



Clément Delangue · [Follow](#)

Published in HuggingFace · 4 min read · May 9, 2019

...

Also, we decided not to release the full GPT2 model but leveraged the smaller version of the model, which generates a conversation which — even if impressive- can still usually easily be told apart from a real human conversation, preventing most of the catfishing impact. We are aligning ourselves with OpenAI in not releasing a bigger model until they do.

Публичная дискуссия сообщества

Что. Было. Дальше.



NVIDIA ADLR

GPT-2 Evaluation

To analyze the performance of training large language models, we compute perplexity on the [WikiText-103 dataset](#) and cloze-style prediction accuracy on the [LAMBADA dataset](#). Earlier we made sure to remove WikiText test set content to avoid leakage. Our resulting dataset has a WikiText 8-gram overlap of 10% which is similar to the 9% 8-gram overlap between the WikiText-103 train and test sets. As expected, the WikiText perplexity decreases and LAMBADA accuracy increases with the growth of the model size (Table 3). At a respective WikiText perplexity of 12.68 and 10.81 both our 2.5B and 8.3B models surpass the previous state of the art perplexity of 16.43 set by [Krause et. al.](#) Our models achieve 61.52% and 66.51% accuracy on LAMBADA even without any stopword filtration, surpassing [Radford et. al.](#) We describe our evaluation methodologies below; however, more details are available in [our github repository](#).

Model Size	Number of layers	Hidden Size	Attention heads	Wikitext (Perplexity .)	Lambada (Accuracy %)
345 M	24	1024	16	19.22	46.26
2.5 B	54	1920	20	12.68	61.52
8.3 B	72	3072	24	10.81	66.51
SOTA	—	—	—	16.43*	63.24**

Table 3: Zero-shot evaluation results for WikiText perplexity (lower is better) and LAMBADA

Реимплементация и масштабирование



Yesterday I launched Talk to Transformer, a site where you can try out @OpenAI's new text-generating language model on your own custom text. talktotransformer.com One example is below. @AlecRad @gdb

Prompt: Before boarding your rocket to Mars, remember to pack these items:

Generated completion:

1. A good pair of gloves

In the event the airlocks fail due to high or low pressures, gloves can ensure survival for months or even years.
2. Two reusable oxygen bottles

Первые проекты

Ethical analysis of the open-sourcing of a state-of-the-art conversational AI



Clément Delangue · [Follow](#)

Published in HuggingFace · 4 min read · May 9, 2019

...

Also, we decided not to release the full GPT2 model but leveraged the smaller version of the model, which generates a conversation which — even if impressive- can still usually easily be told apart from a real human conversation, preventing most of the catfishing impact. We are aligning ourselves with OpenAI in not releasing a bigger model until they do.

Публичная дискуссия сообщества

Что. Было. Дальше.



NVIDIA ADLR

GPT-2 Evaluation

To analyze the performance of training large language models, we compute perplexity on the [WikiText-103 dataset](#) and cloze-style prediction accuracy on the [LAMBADA dataset](#). Earlier we made sure to remove WikiText test set content to avoid leakage. Our resulting dataset has a WikiText 8-gram overlap of 10% which is similar to the 9% 8-gram overlap between the WikiText-103 train and test sets. As expected, the WikiText perplexity decreases and LAMBADA accuracy increases with the growth of the model size (Table 3). At a respective WikiText perplexity of 12.68 and 10.81 both our 2.5B and 8.3B models surpass the previous state of the art perplexity of 16.43 set by [Krause et. al.](#) Our models achieve 61.52% and 66.51% accuracy on LAMBADA even without any stopword filtration, surpassing [Radford et. al.](#) We describe our evaluation methodologies below; however, more details are available in [our github repository](#).

Model Size	Number of layers	Hidden Size	Attention heads	Wikitext (Perplexity .)	Lambada (Accuracy %)
345 M	24	1024	16	19.22	46.26
2.5 B	54	1920	20	12.68	61.52
8.3 B	72	3072	24	10.81	66.51
SOTA	—	—	—	16.43*	63.24**

Table 3: Zero-shot evaluation results for WikiText perplexity (lower is better) and LAMBADA

Реимплементация и масштабирование

Adam King
@AdamDanielKing

Yesterday I launched Talk to Transformer, a site where you can try out @OpenAI's new text-generating language model on your own custom text. [talktotransformer.com](#) One example is below. @AlecRad @gdb

Prompt: Before boarding your rocket to Mars, remember to pack these items:

Generated completion:

1. A good pair of gloves

In the event the airlocks fail due to high or low pressures, gloves can ensure survival for months or even years.

2. Two reusable oxygen bottles

Первые проекты

Ethical analysis of the open-sourcing of a state-of-the-art conversational AI



Clément Delangue · [Follow](#)

Published in HuggingFace · 4 min read · May 9, 2019

...

Also, we decided not to release the full GPT2 model but leveraged the smaller version of the model, which generates a conversation which — even if impressive- can still usually easily be told apart from a real human conversation, preventing most of the catfishing impact. We are aligning ourselves with OpenAI in not releasing a bigger model until they do.

Публичная дискуссия сообщества



Ms. Danielle Citron

Mr. Jack Clark

OpenAI в Сенате США



Что. Было. Дальше.



NVIDIA ADLR

GPT-2 Evaluation

To analyze the performance of training large language models, we compute perplexity on the [WikiText-103 dataset](#) and cloze-style prediction accuracy on the [LAMBADA dataset](#). Earlier we made sure to remove WikiText test set content to avoid leakage. Our resulting dataset has a WikiText 8-gram overlap of 10% which is similar to the 9% 8-gram overlap between the WikiText-103 train and test sets. As expected, the WikiText perplexity decreases and LAMBADA accuracy increases with the growth of the model size (Table 3). At a respective WikiText perplexity of 12.68 and 10.81 both our 2.5B and 8.3B models surpass the previous state of the art perplexity of 16.43 set by [Krause et. al.](#) Our models achieve 61.52% and 66.51% accuracy on LAMBADA even without any stopword filtration, surpassing [Radford et. al.](#) We describe our evaluation methodologies below; however, more details are available in [our github repository](#).

Model Size	Number of layers	Hidden Size	Attention heads	Wikitext (Perplexity .)	Lambada (Accuracy %)
345 M	24	1024	16	19.22	46.26
2.5 B	54	1920	20	12.68	61.52
8.3 B	72	3072	24	10.81	66.51
SOTA	—	—	—	16.43*	63.24**

Table 3: Zero-shot evaluation results for WikiText perplexity (lower is better) and LAMBADA

Реимплементация и масштабирование



Yesterday I launched Talk to Transformer, a site where you can try out @OpenAI's new text-generating language model on your own custom text. [talktotransformer.com](#) One example is below. @AlecRad @gdb

Prompt: Before boarding your rocket to Mars, remember to pack these items:

Generated completion:

1. A good pair of gloves

In the event the airlocks fail due to high or low pressures, gloves can ensure survival for months or even years.

2. Two reusable oxygen bottles

Первые проекты

Ethical analysis of the open-sourcing of a state-of-the-art conversational AI



Clément Delangue · [Follow](#)

Published in HuggingFace · 4 min read · May 9, 2019

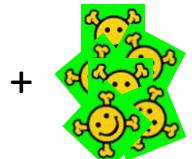
...

Also, we decided not to release the full GPT2 model but leveraged the smaller version of the model, which generates a conversation which — even if impressive- can still usually easily be told apart from a real human conversation, preventing most of the catfishing impact. We are aligning ourselves with OpenAI in not releasing a bigger model until they do.

Публичная дискуссия сообщества



Ms. Danielle Citron
Mr. Jack Clark



OpenAI в Сенате США

Итоги

- Модели впервые начали генерировать длинные тексты качественно
- OpenAI уже тогда поднимали вопросы и ходили в Сенат

Итоги

- Модели впервые начали генерировать длинные тексты качественно
- OpenAI уже тогда поднимали вопросы и ходили в Сенат
- Модели семейства GPT-2 не были опубликованы сразу
- За полгода разные компании смогли воспроизвести GPT-2
 - И даже увеличить — в случае Nvidia до 8.3B!

Итоги

- Модели впервые начали генерировать длинные тексты качественно
- OpenAI уже тогда поднимали вопросы и ходили в Сенат
- Модели семейства GPT-2 не были опубликованы сразу
- За полгода разные компании смогли воспроизвести GPT-2
 - И даже увеличить — в случае Nvidia до 8.3B!
- В настоящее время нет надёжных детекторов сгенерированного текста!

Directed by
IGOR D. KOTENKOV

Полная история семейства GPT: -1, 1, 2, 3...



Котенков Игорь



@stm

Модуль 2, часть 3. План:

- Характеристики GPT-3
- Соревнование по размерам GPU-кластера
- Тонкости сбора датасета
- Sparse Attention для уменьшения потребляемой памяти

GPT-3: Больше, выше, толще

	GPT-1	GPT-2	GPT-3
Количество параметров	117М	1.5В	175В
Количество слоёв	12	48	96

GPT-3: Больше, выше, толще

	GPT-1	GPT-2	GPT-3
Количество параметров	117M	1.5B	175B
Количество слоёв	12	48	96
Размер контекста	512	1024	2048

GPT-3: Больше, выше, толще

	GPT-1	GPT-2	GPT-3
Количество параметров	117M	1.5B	175B
Количество слоёв	12	48	96
Размер контекста	512	1024	2048
d_{model}	768	1600	12288

GPT-3: Больше, выше, толще

	GPT-1	GPT-2	GPT-3
Количество параметров	117М	1.5В	175В
Количество слоёв	12	48	96
Размер контекста	512	1024	2048
d_{model}	768	1600	12288
n_{heads}	12 ($d_{head} = 64$)	25 ($d_{head} = 64$)	96 ($d_{head} = 128$)

GPT-3: Больше, выше, толще

	GPT-1	GPT-2	GPT-3
Количество параметров	117М	1.5В	175В
Количество слоёв	12	48	96
Размер контекста	512	1024	2048
d_{model}	768	1600	12288
n_{heads}	12 ($d_{head} = 64$)	25 ($d_{head} = 64$)	96 ($d_{head} = 128$)
Размер батча	64	512	$\sim 3.2M$

GPT-3: Больше, выше, толще

	GPT-1	GPT-2	GPT-3
Количество параметров	117М	1.5В	175В
Количество слоёв	12	48	96
Размер контекста	512	1024	2048
d_{model}	768	1600	12288
n_{heads}	12 ($d_{\text{head}} = 64$)	25 ($d_{\text{head}} = 64$)	96 ($d_{\text{head}} = 128$)
Размер батча	64	512	$\sim 3.2\text{M}$

Model Name	n_{params}	n_{layers}	d_{model}	n_{heads}	d_{head}	Batch Size	Learning Rate
GPT-3 Small	125М	12	768	12	64	0.5М	6.0×10^{-4}
GPT-3 Medium	350М	24	1024	16	64	0.5М	3.0×10^{-4}
GPT-3 Large	760М	24	1536	16	96	0.5М	2.5×10^{-4}
GPT-3 XL	1.3B	24	2048	24	128	1M	2.0×10^{-4}
GPT-3 2.7B	2.7B	32	2560	32	80	1M	1.6×10^{-4}
GPT-3 6.7B	6.7B	32	4096	32	128	2M	1.2×10^{-4}
GPT-3 13B	13.0B	40	5140	40	128	2M	1.0×10^{-4}
GPT-3 175B or “GPT-3”	175.0B	96	12288	96	128	3.2M	0.6×10^{-4}

GPT-3: Больше, выше, толще

	GPT-1	GPT-2		GPT-3		
Количество параметров	117М	1.5В			175В	
Количество слоёв	12	48			96	
Размер контекста	512	1024			2048	
d_{model}	768	1600			12288	
n_{heads}	12 ($d_{\text{head}} = 64$)	25 ($d_{\text{head}} = 64$)			96 ($d_{\text{head}} = 128$)	
Размер батча	64	32k	512	524k	~1536	~ 3.2М

Model Name	n_{params}	n_{layers}	d_{model}	n_{heads}	d_{head}	Batch Size	Learning Rate
GPT-3 Small	125М	12	768	12	64	0.5М	6.0×10^{-4}
GPT-3 Medium	350М	24	1024	16	64	0.5М	3.0×10^{-4}
GPT-3 Large	760М	24	1536	16	96	0.5М	2.5×10^{-4}
GPT-3 XL	1.3B	24	2048	24	128	1М	2.0×10^{-4}
GPT-3 2.7B	2.7B	32	2560	32	80	1М	1.6×10^{-4}
GPT-3 6.7B	6.7B	32	4096	32	128	2М	1.2×10^{-4}
GPT-3 13B	13.0B	40	5140	40	128	2М	1.0×10^{-4}
GPT-3 175B or “GPT-3”	175.0B	96	12288	96	128	3.2М	0.6×10^{-4}

GPT-3: Больше, выше, толще

	GPT-1	GPT-2	GPT-3	
Количество параметров	117М	1.5В	175В	
Количество слоёв	12	48	96	
Размер контекста	512	1024	2048	
d_{model}	768	1600	12288	
n_{heads}	12 ($d_{head} = 64$)	25 ($d_{head} = 64$)	96 ($d_{head} = 128$)	
Размер батча	64	32k	512	524k
			~1536	~ 3.2М

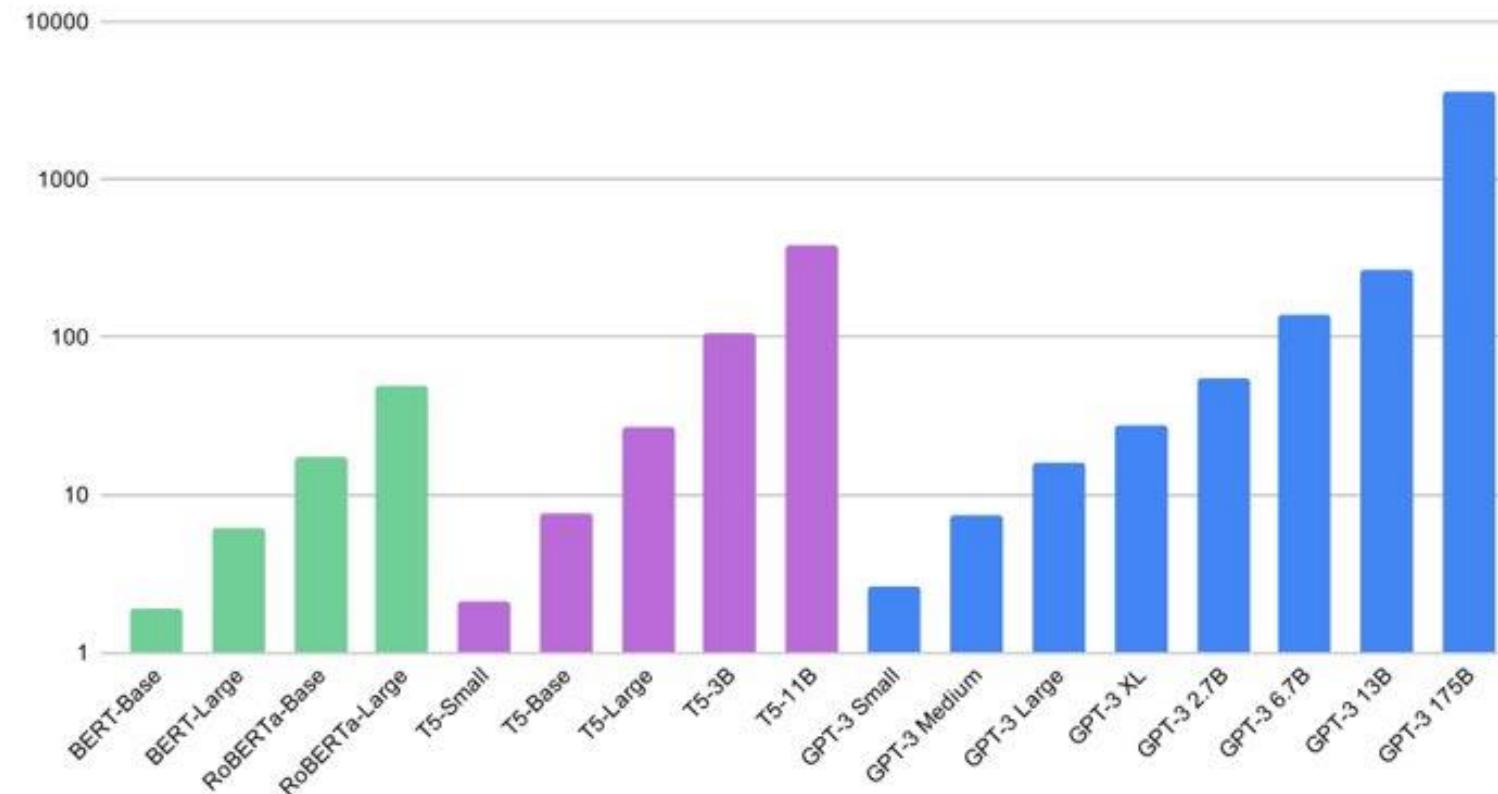
→ Context Dim (T=10)

4342	318	281	1672	3188	352	4478	617	16326	13
16281	3188	362	50256	16281	3188	513	50256	16281	3188
1212	318	617	4738	2420	655	329	1672	50256	1212
16	11	17	11	18	11	19	11	20	11

Batch Dim (B=4) ↓

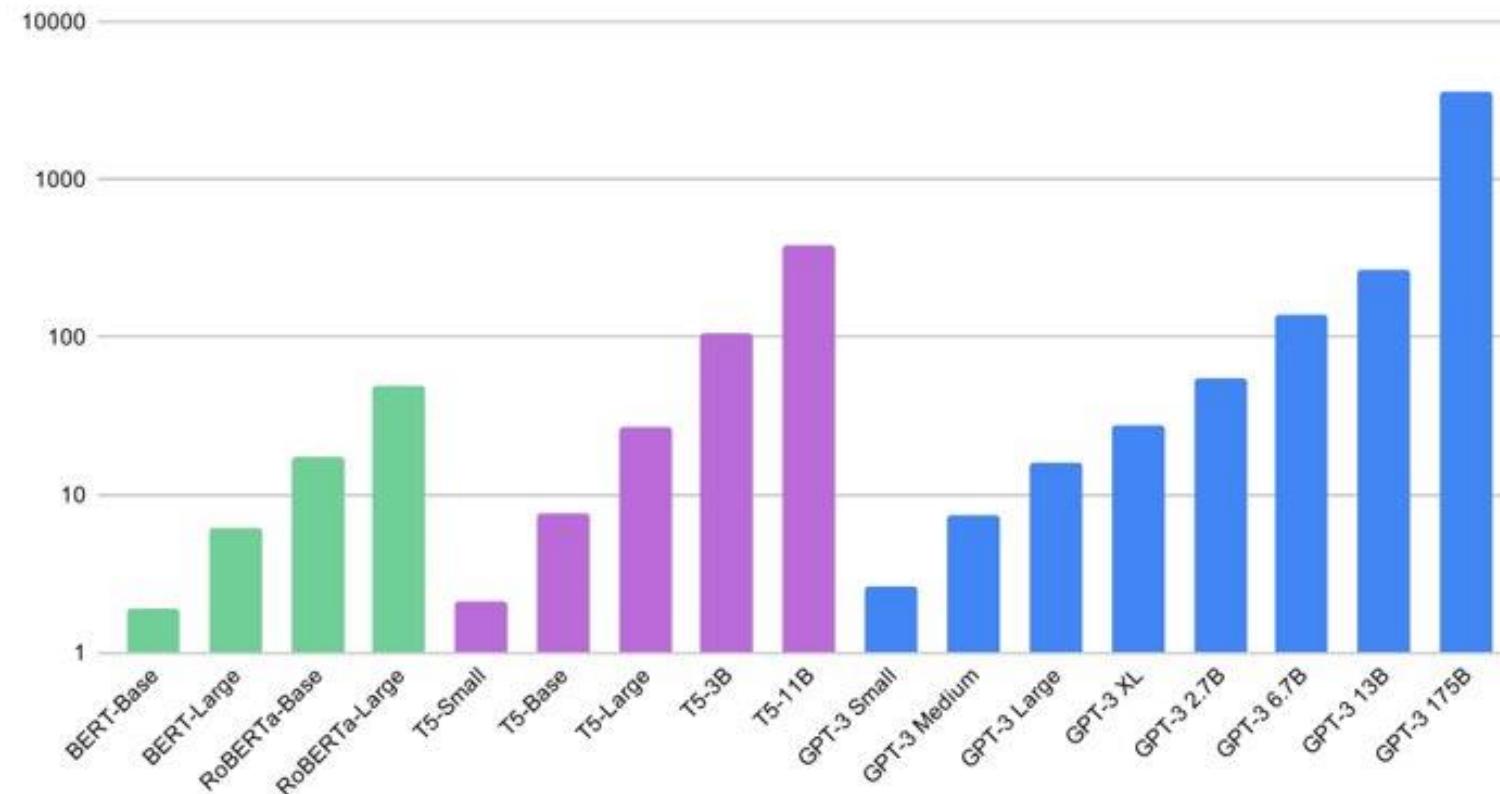
Компьютера – да побольше!

Total Compute Used During Training



Компьютера – да побольше!

Total Compute Used During Training



Exploring Transfer Learning with T5: the Text-To-Text Transfer Transformer

MONDAY, FEBRUARY 24, 2020

Posted by Adam Roberts, Staff Software Engineer and Colin Raffel, Senior Research Scientist, Google

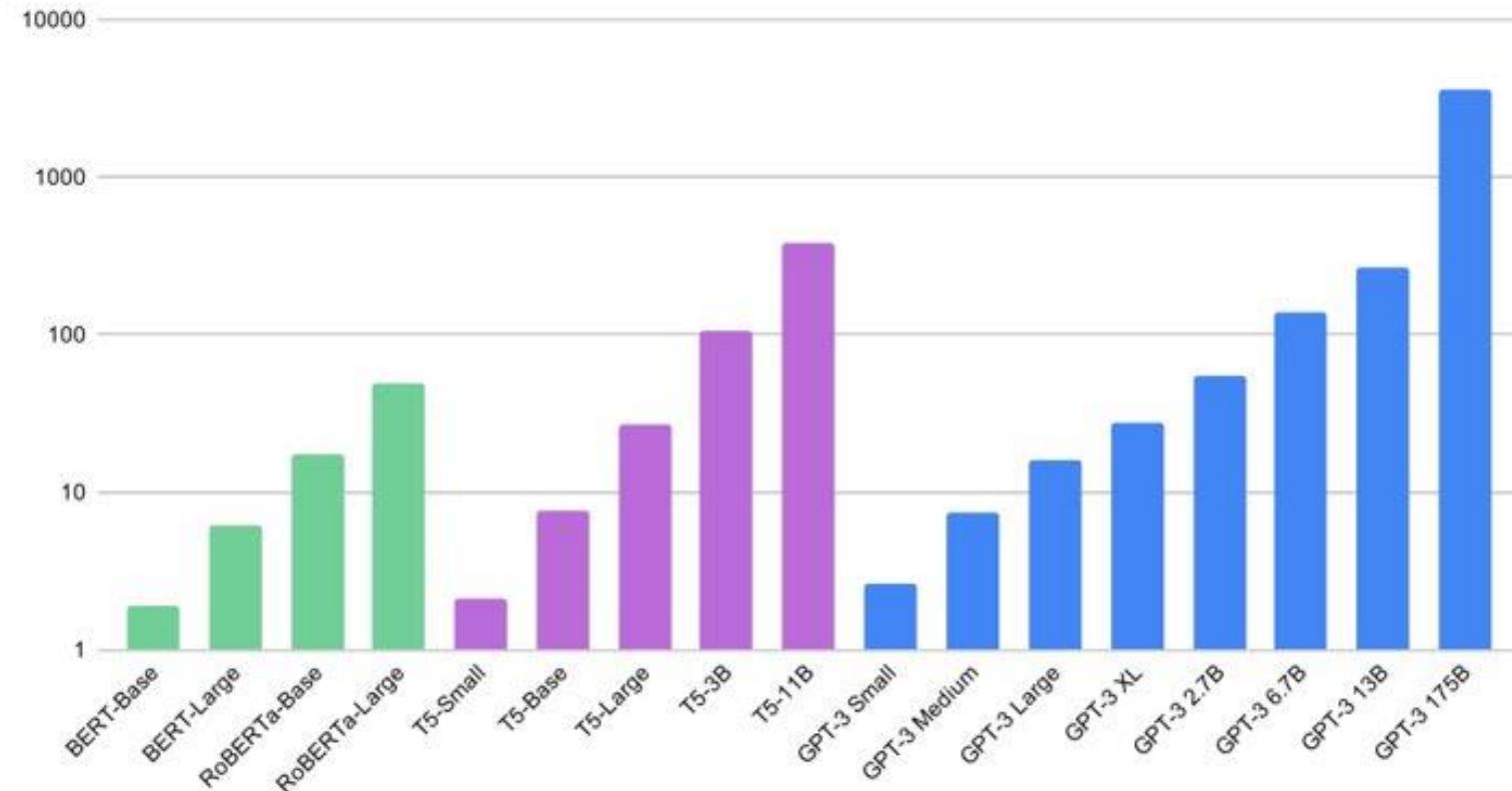
Turing-NLG: A 17-billion-parameter language model by Microsoft

Published February 13, 2020

By Corby Rosset, Applied Scientist

Компьютера – да побольше!

Total Compute Used During Training



Model	Total train compute (PF-days)	Total train compute (flops)
T5-3B	1.04E+02	9.00E+21
T5-11B	3.82E+02	3.30E+22
GPT-3 13B	2.68E+02	2.31E+22
GPT-3 175B	3.64E+03	3.14E+23

Exploring Transfer Learning with T5: the Text-To-Text Transfer Transformer

MONDAY, FEBRUARY 24, 2020

Posted by Adam Roberts, Staff Software Engineer and Colin Raffel, Senior Research Scientist, Google

Turing-NLG: A 17-billion-parameter language model by Microsoft

Published February 13, 2020

By Corby Rosset, Applied Scientist

Но где брать данные?

Dataset	Quantity (tokens)	Weight in training mix	Epochs elapsed when training for 300B tokens
Common Crawl (filtered)	410 billion	60%	0.44
WebText2	19 billion	22%	2.9
Books1	12 billion	8%	1.9
Books2	55 billion	8%	0.43
Wikipedia	3 billion	3%	3.4

Но где брать данные?

Dataset	Quantity (tokens)	Weight in training mix	Epochs elapsed when training for 300B tokens
Common Crawl (filtered)	410 billion	60%	0.44
WebText2	19 billion	22%	2.9
Books1	12 billion	8%	1.9
Books2	55 billion	8%	0.43
Wikipedia	3 billion	3%	3.4

←
Только
английский

Улучшение
датасета
GPT-2 →

Но где брать данные?

Dataset	Quantity (tokens)	Weight in training mix	Epochs elapsed when training for 300B tokens
<u>Common Crawl (filtered)</u>	410 billion	60%	0.44
WebText2	19 billion	22%	2.9
Books1	12 billion	8%	1.9
Books2	55 billion	8%	0.43
<u>Wikipedia</u>	3 billion	3%	3.4

← «As is» всё
ещё грязный

Только
английский

Улучшение
датасета
GPT-2

Но где брать данные?

41 шард за 2016-2019
(45 TB compressed plaintext)

Dataset	Quantity (tokens)	Weight in training mix	Epochs elapsed when training for 300B tokens
<u>Common Crawl (filtered)</u>	410 billion	60%	0.44
WebText2	19 billion	22%	2.9
Books1	12 billion	8%	1.9
Books2	55 billion	8%	0.43
<u>Wikipedia</u>	3 billion	3%	3.4

«As is» всё
ещё грязный

Только
английский

Улучшение
датасета
GPT-2

Но где брать данные?

570GB после
фильтрации и дедупликации

41 шард за 2016-2019
(45 TB compressed plaintext)

«As is» всё ещё грязный

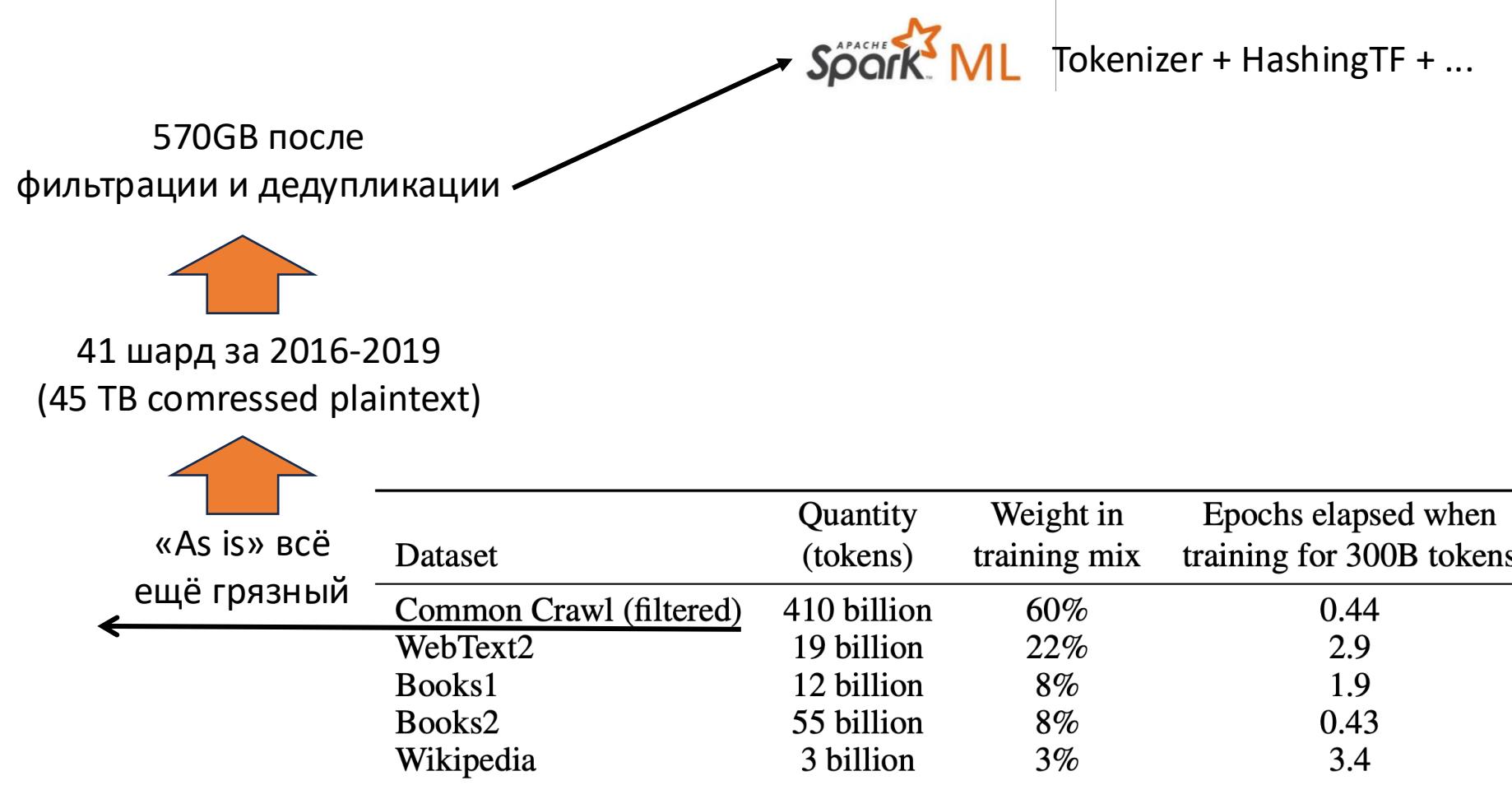
The diagram illustrates the evolution of GPT-2 datasets. It features a horizontal timeline with arrows pointing from left to right, indicating the progression. On the far left, there is a small orange square icon. To its right, the text "«As is» всё" and "ещё грязный" is displayed above a double-headed arrow. This arrow points to the first dataset in the table, "Common Crawl (filtered)". The table itself has four columns: "Dataset", "Quantity (tokens)", "Weight in training mix", and "Epochs elapsed when training for 300B tokens". The data rows are as follows:

Dataset	Quantity (tokens)	Weight in training mix	Epochs elapsed when training for 300B tokens
Common Crawl (filtered)	410 billion	60%	0.44
WebText2	19 billion	22%	2.9
Books1	12 billion	8%	1.9
Books2	55 billion	8%	0.43
Wikipedia	3 billion	3%	3.4

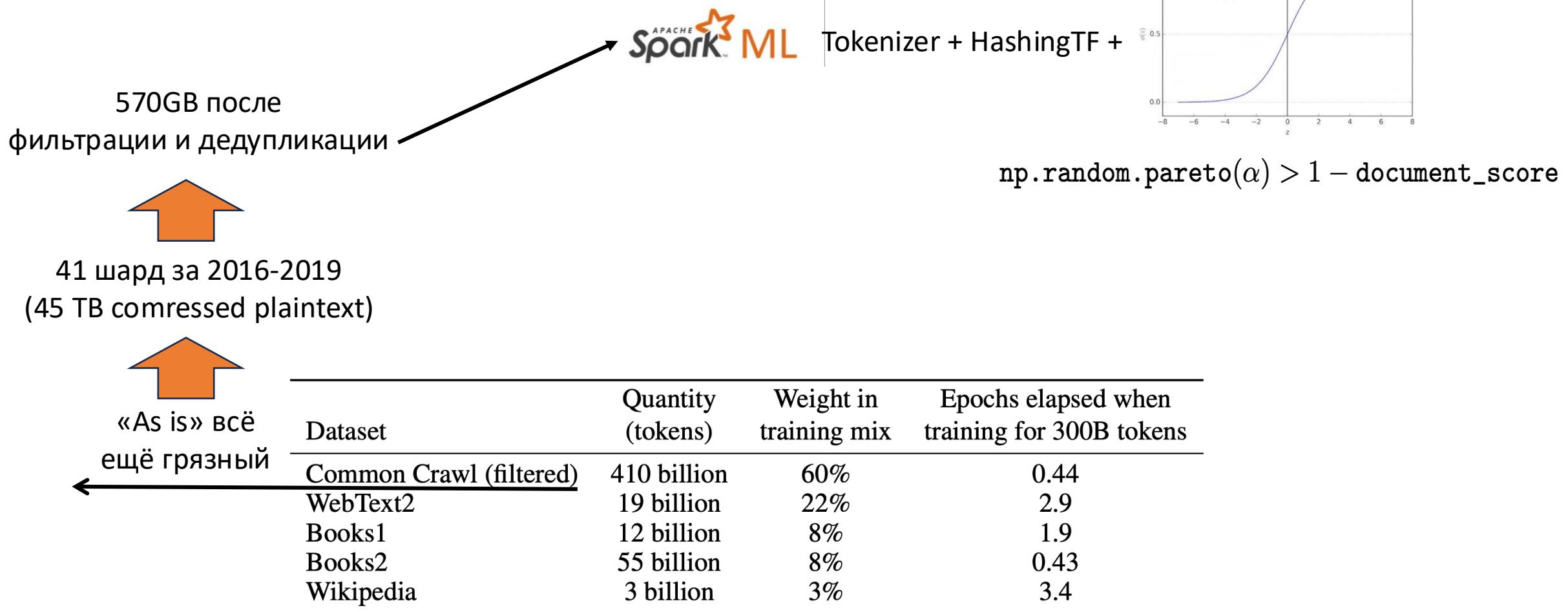
To the right of the table, another double-headed arrow points from the last dataset to the right, labeled "Улучшение датасета GPT-2".

At the bottom left, a double-headed arrow points from the first dataset to the right, labeled "Только английский".

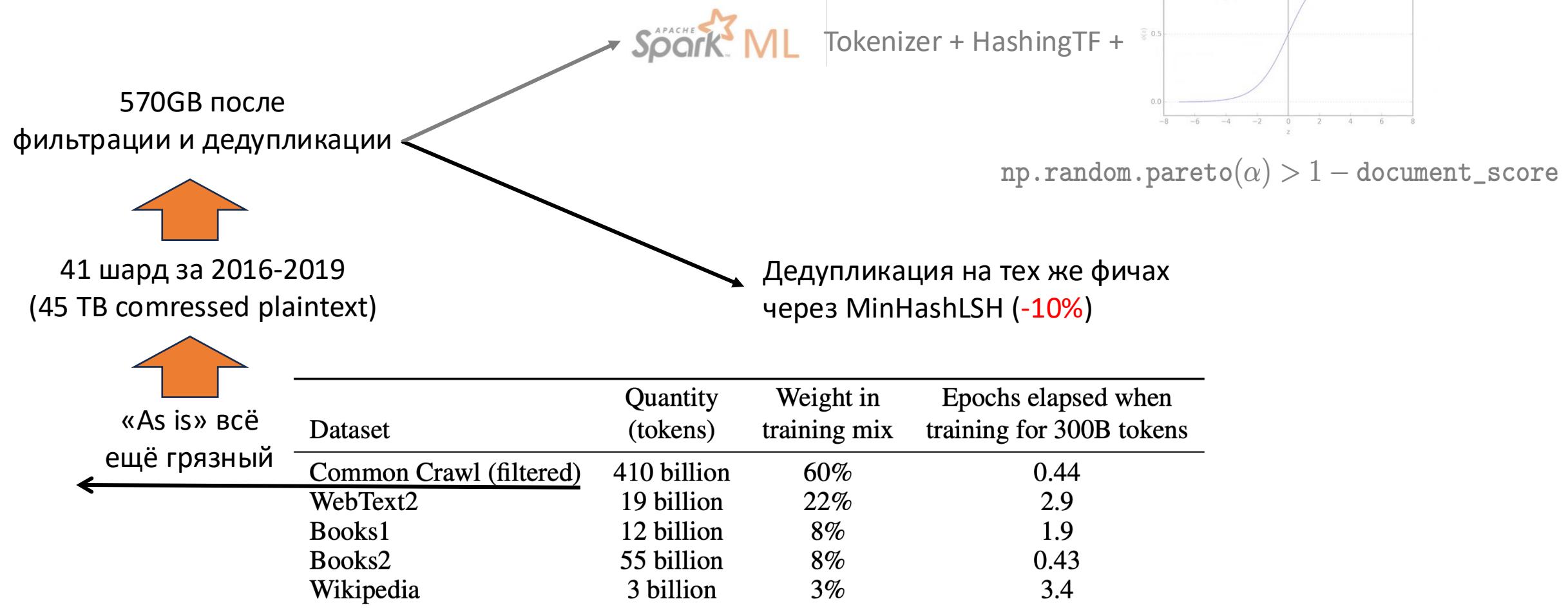
Но где брать данные?



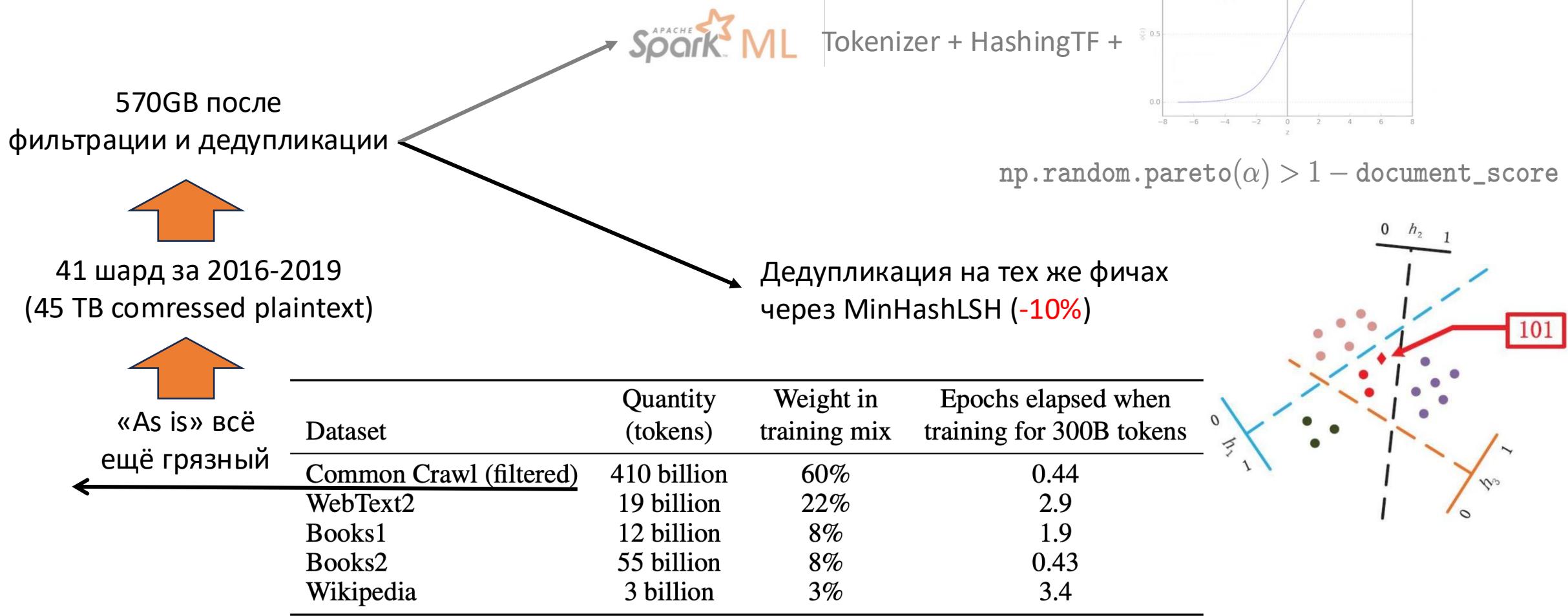
Но где брать данные?



Но где брать данные?

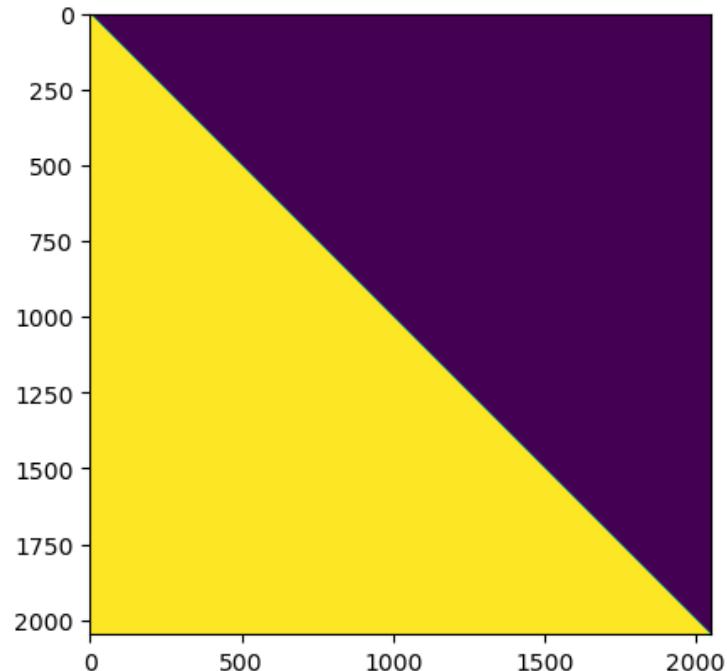


Но где брать данные?



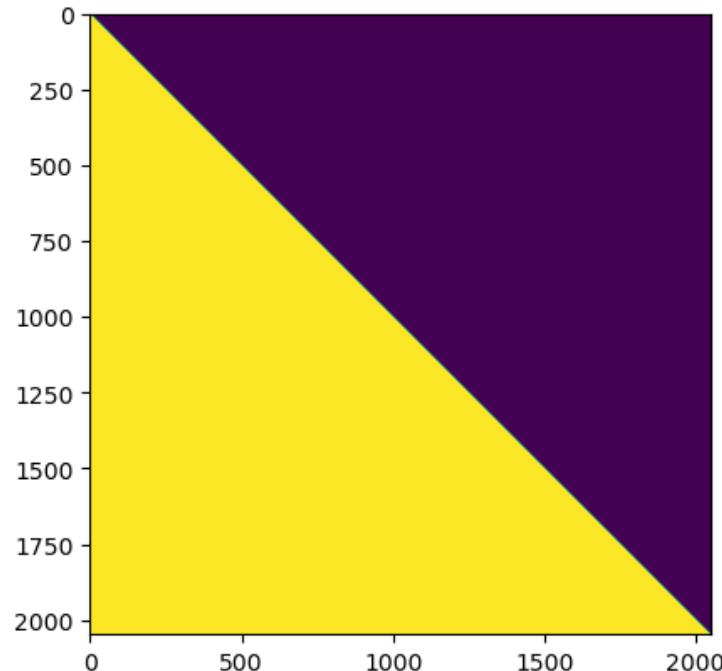
И снова про внимание

Dense attention map:



И снова про внимание

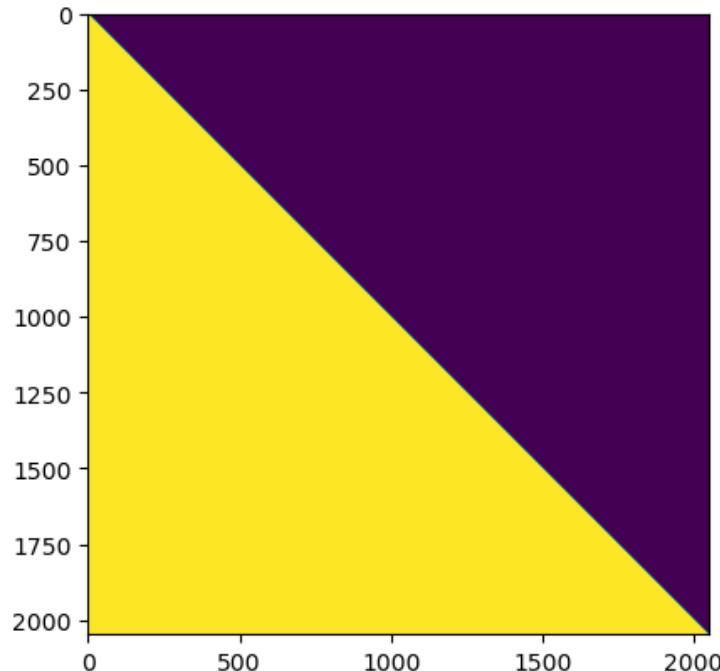
Dense attention map:



Layer Type	Complexity per Layer
Self-Attention	$O(n^2 \cdot d)$
Recurrent	$O(n \cdot d^2)$

И снова про внимание

Dense attention map:



Layer Type	Complexity per Layer
------------	----------------------

Self-Attention

$$O(n^2 \cdot d)$$

d ожидание: 12288

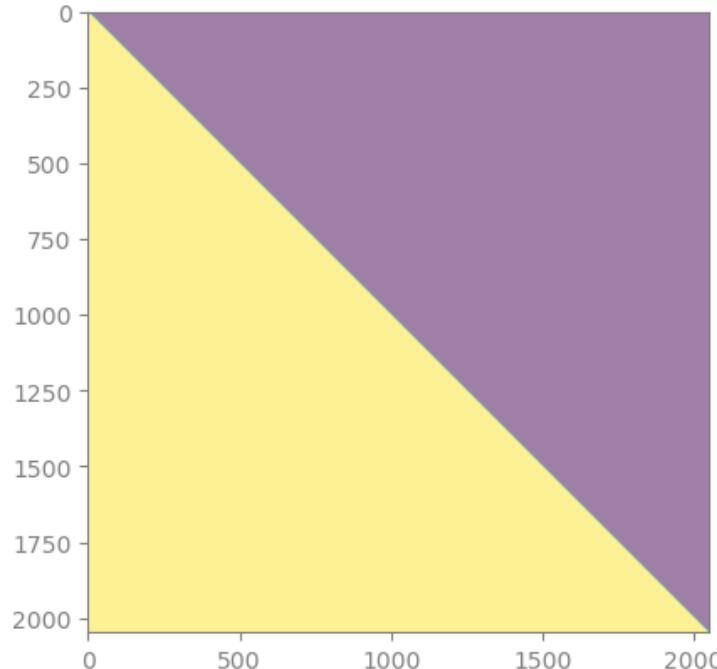
Recurrent

$$O(n \cdot d^2)$$

d реальность: 128

И снова про внимание

Dense attention map:



Решение: перейдём к sparse attention

- проанализируем паттерны в обученной модели

Layer Type

Complexity per Layer

Self-Attention

$$O(n^2 \cdot d)$$

d ожидание: 12288

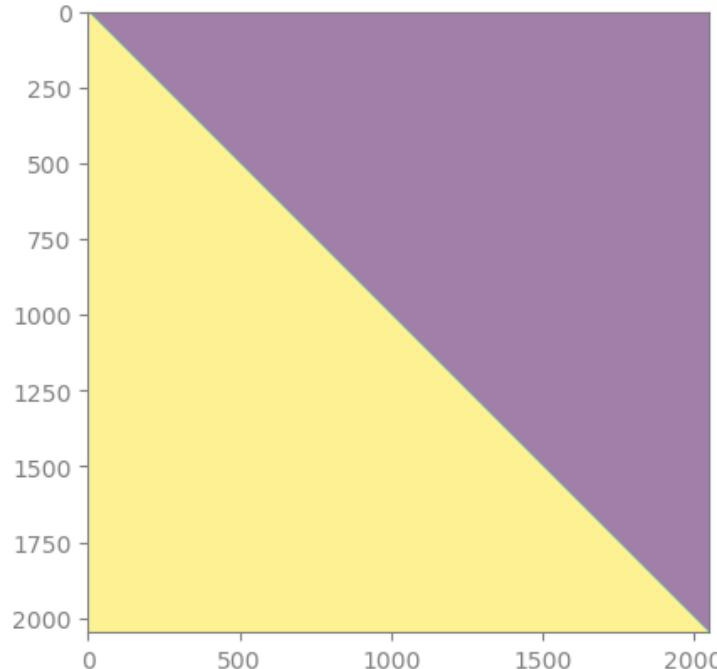
Recurrent

$$O(n \cdot d^2)$$

d реальность: 128

И снова про внимание

Dense attention map:



Решение: перейдём к **sparse** attention

- проанализируем паттерны в обученной модели
- факторизуем на p компонент
- при $p = 2$ введём следующую логику:
 - один фактор отвечает за local фичи
 - второй — за global

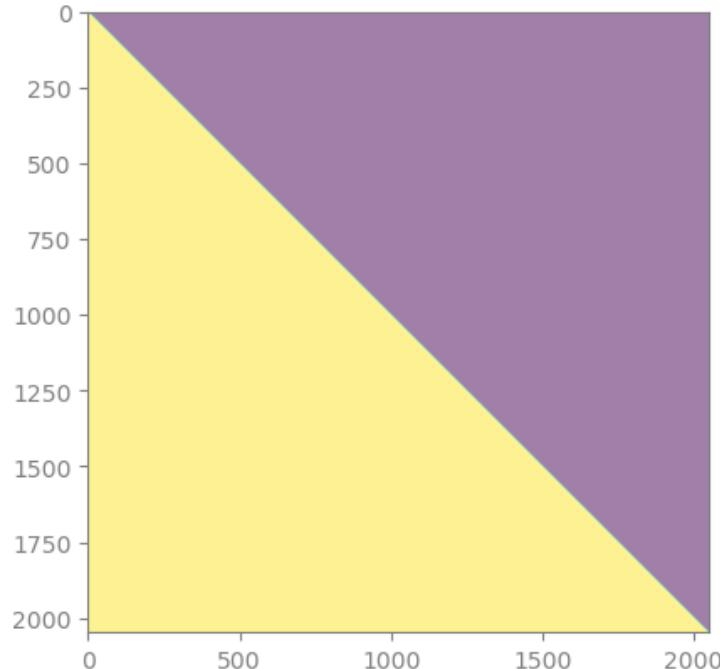
Layer Type	Complexity per Layer
------------	----------------------

Self-Attention	$O(n^2 \cdot d)$
Recurrent	$O(n \cdot d^2)$

d ожидание: 12288
d реальность: 128

И снова про внимание

Dense attention map:



Решение: перейдём к **sparse** attention

- проанализируем паттерны в обученной модели
- факторизуем на p компонент
- при $p = 2$ введём следующую логику:
 - один фактор отвечает за local фичи
 - второй — за global

$$A_i^{(1)} = \{j : (\lfloor j/l \rfloor = \lfloor i/l \rfloor)\}$$

$$A_i^{(2)} = \{j : j \bmod l \in \{t, t+1, \dots, l\}\}$$

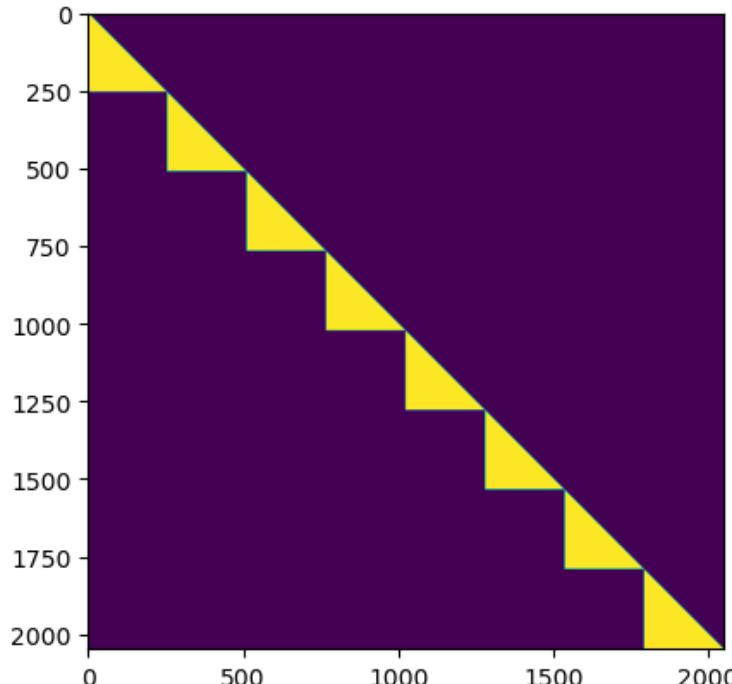
Layer Type	Complexity per Layer
------------	----------------------

Self-Attention
Recurrent

$O(n^2 \cdot d)$
 $O(n \cdot d^2)$

d ожидание: 12288
 d реальность: 128

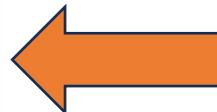
И снова про внимание



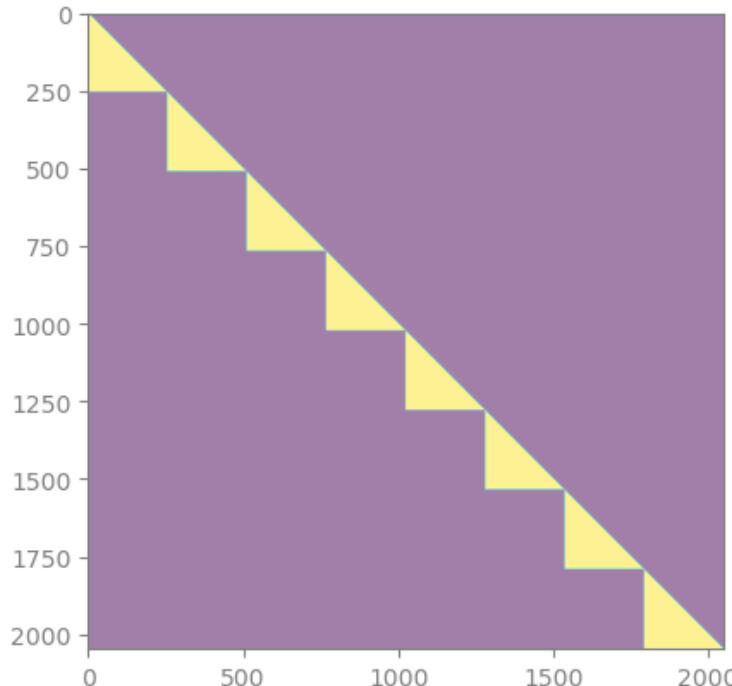
Решение: перейдём к **sparse** attention

- проанализируем паттерны в обученной модели
- факторизуем на p компонент
- при $p = 2$ введём следующую логику:
 - один фактор отвечает за local фичи
 - второй — за global

$$A_i^{(1)} = \{j : (\lfloor j/l \rfloor = \lfloor i/l \rfloor)\}$$



И снова про внимание

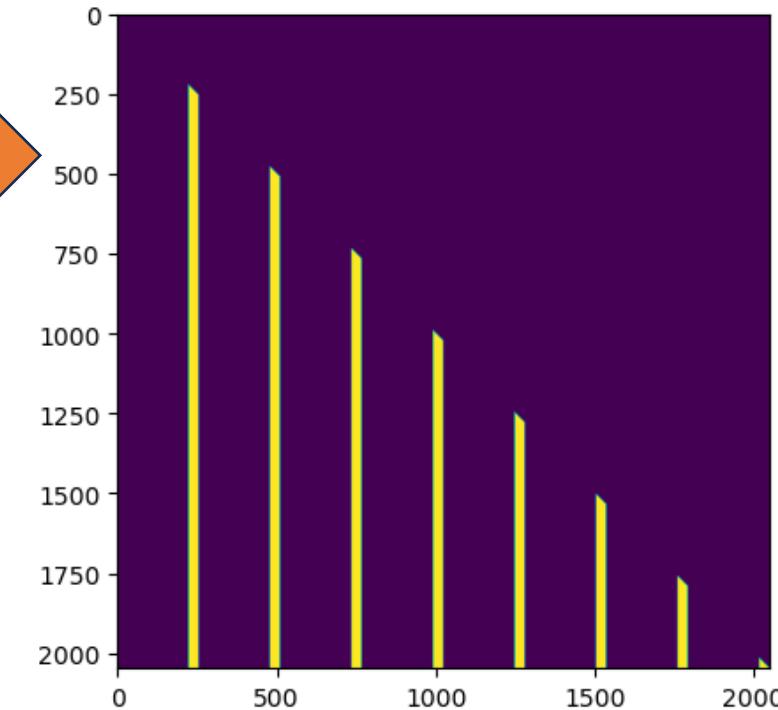


Решение: перейдём к **sparse** attention

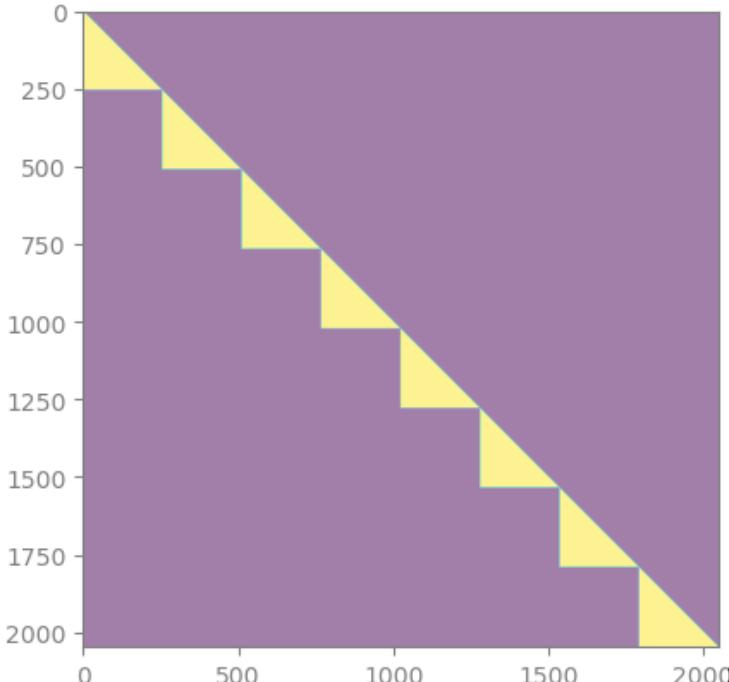
- проанализируем паттерны в обученной модели
- факторизуем на p компонент
- при $p = 2$ введём следующую логику:
 - один фактор отвечает за local фичи
 - второй — за global

$$A_i^{(1)} = \{j : (\lfloor j/l \rfloor = \lfloor i/l \rfloor)\}$$

$$A_i^{(2)} = \{j : j \bmod l \in \{t, t+1, \dots, l\}\}$$



И снова про внимание



Sparse Transformer:

- stride = 128
- c = 32

Решение: перейдём к **sparse** attention

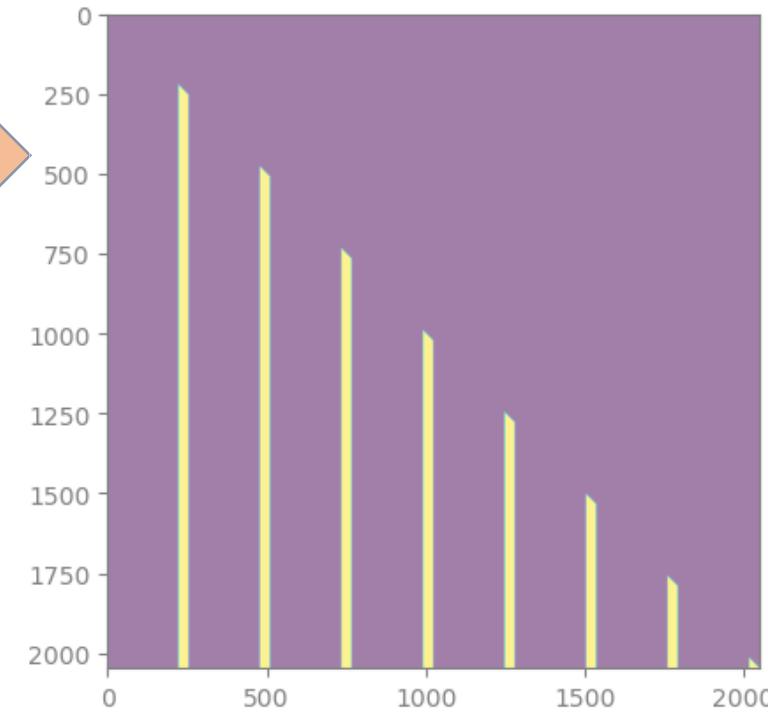
- проанализируем паттерны в обученной модели
- факторизуем на p компонент
- при $p=2$ введём следующую логику:
 - один фактор отвечает за local фичи
 - второй — за global

$$A_i^{(1)} = \{j : (\lfloor j/l \rfloor = \lfloor i/l \rfloor)\}$$
$$A_i^{(2)} = \{j : j \bmod l \in \{t, t+1, \dots, l\}\}$$

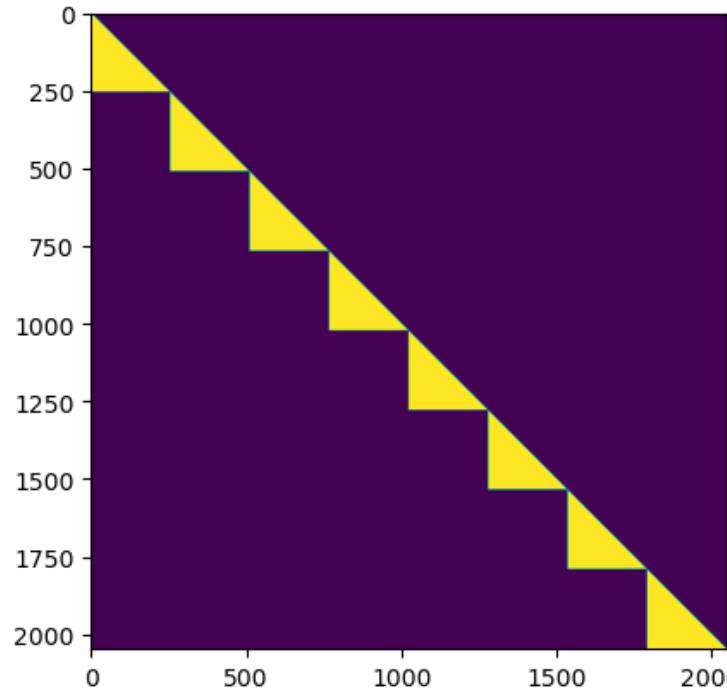
```
ctx_len = 2048
stride_l = 256
c_offset = 32

attn_mask = []

for cur_pos_idx_i in range(ctx_len):
    cur_attn_mask = []
    for j in range(0, cur_pos_idx_i+1):
        # pattern 1
        if int(j/stride_l) == int(cur_pos_idx_i/stride_l):
            cur_attn_mask.append(1)
        # pattern 2
        elif j % stride_l in set(range(stride_l-c_offset, stride_l+1)):
            cur_attn_mask.append(1)
        else:
            cur_attn_mask.append(0)
    cur_attn_mask.extend([0]*(ctx_len-cur_pos_idx_i-1))
    attn_mask.append(cur_attn_mask)
```

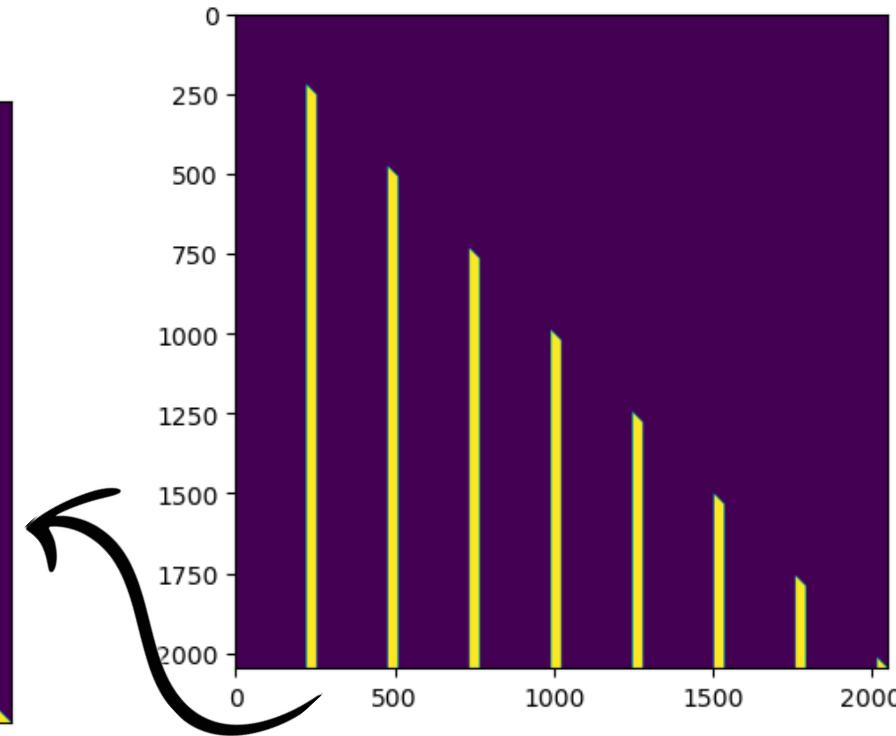
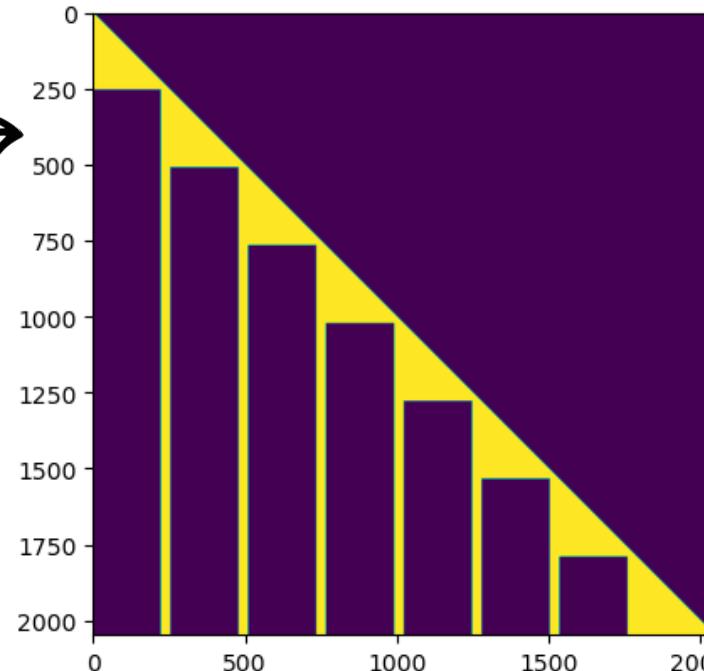


И снова про внимание

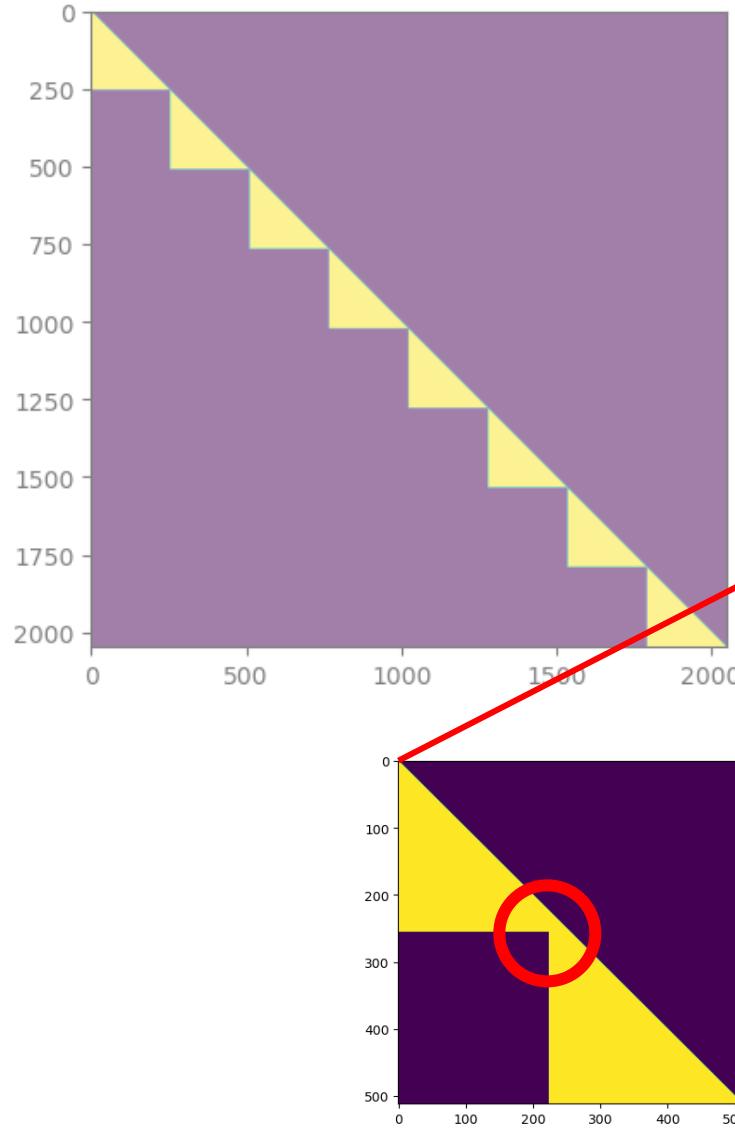


Решение: перейдём к *sparse* attention

- проанализируем паттерны в обученной модели
- факторизуем на p компонент
- при $p = 2$ введём следующую логику:
 - один фактор отвечает за local фичи
 - второй — за global

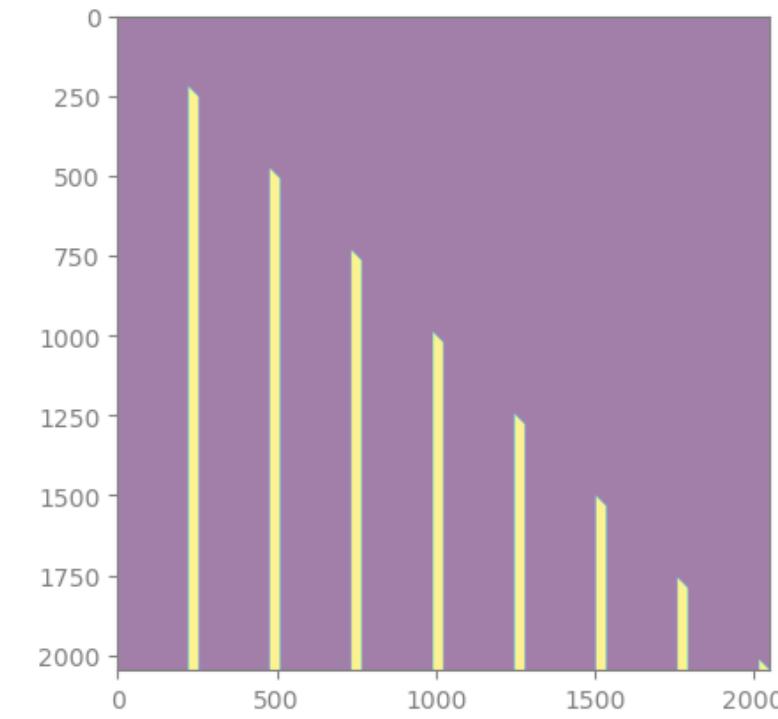
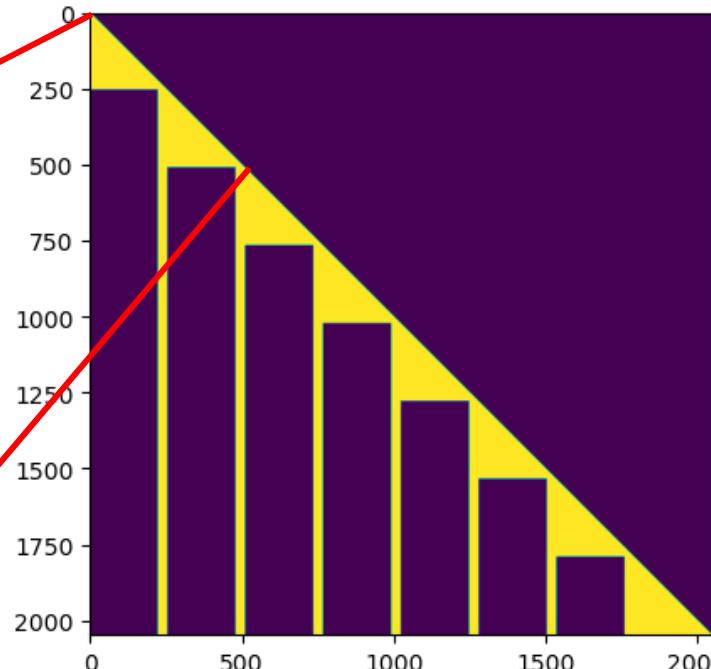


И снова про внимание

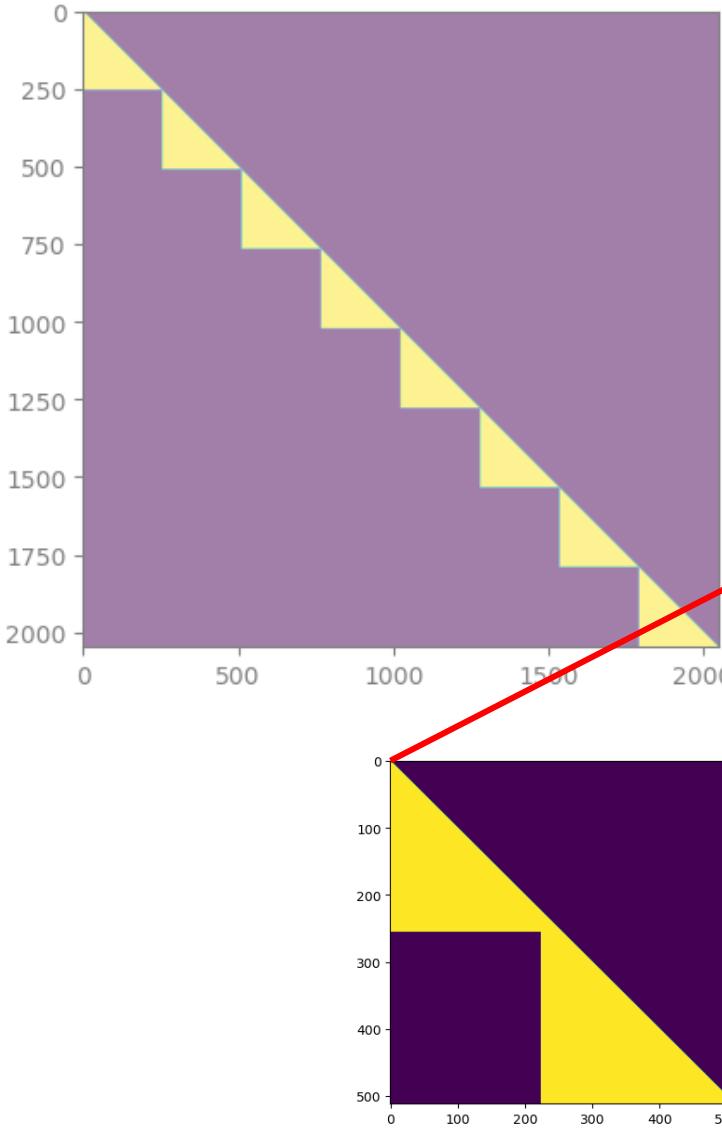


Решение: перейдём к **sparse** attention

- проанализируем паттерны в обученной модели
- факторизуем на p компонент
- при $p = 2$ введём следующую логику:
 - один фактор отвечает за local фичи
 - второй — за global

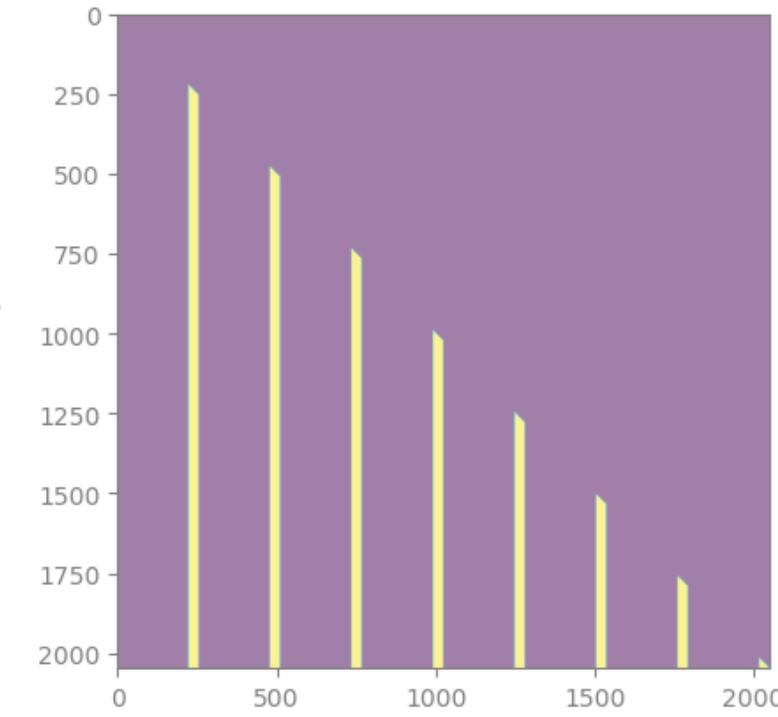
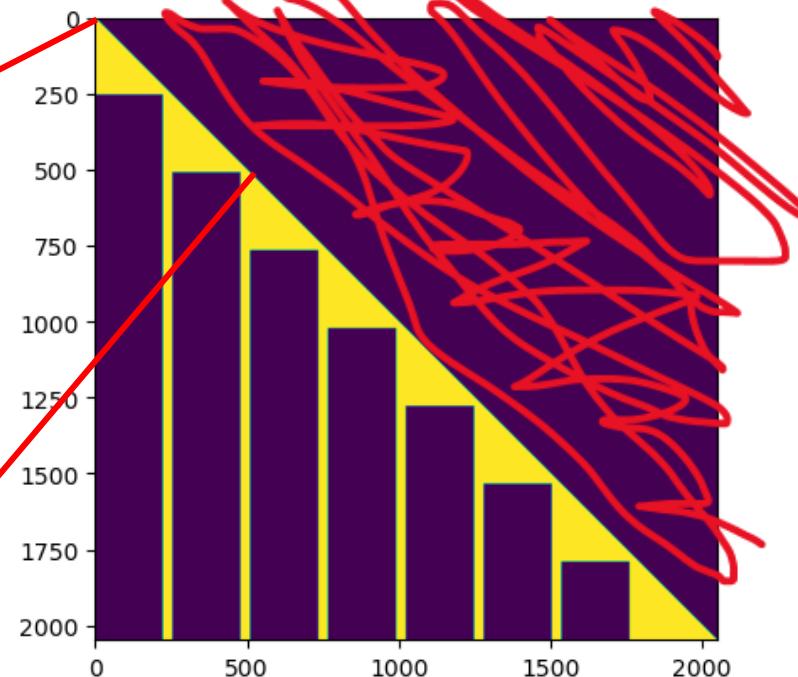


И снова про внимание



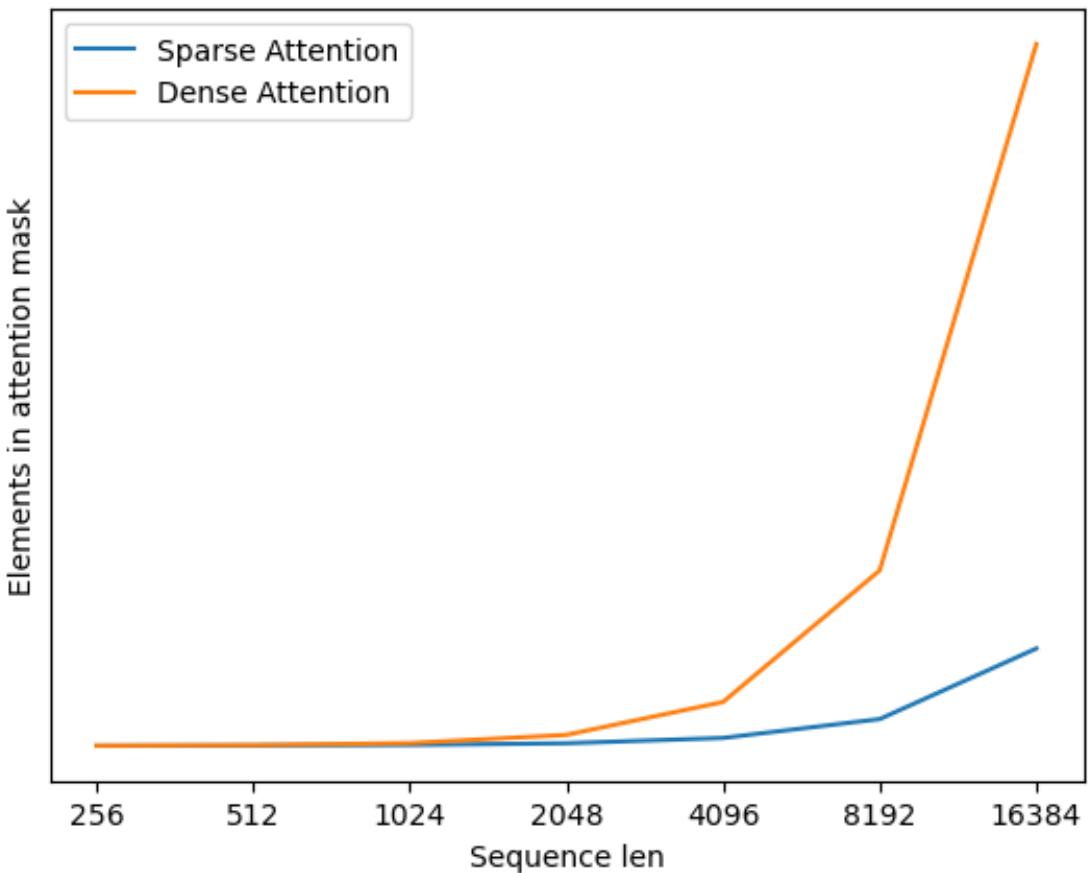
Решение: перейдём к **sparse** attention

- проанализируем паттерны в обученной модели
- факторизуем на p компонент
- при $p = 2$ введём следующую логику:
 - один фактор отвечает за local фичи
 - второй — за global



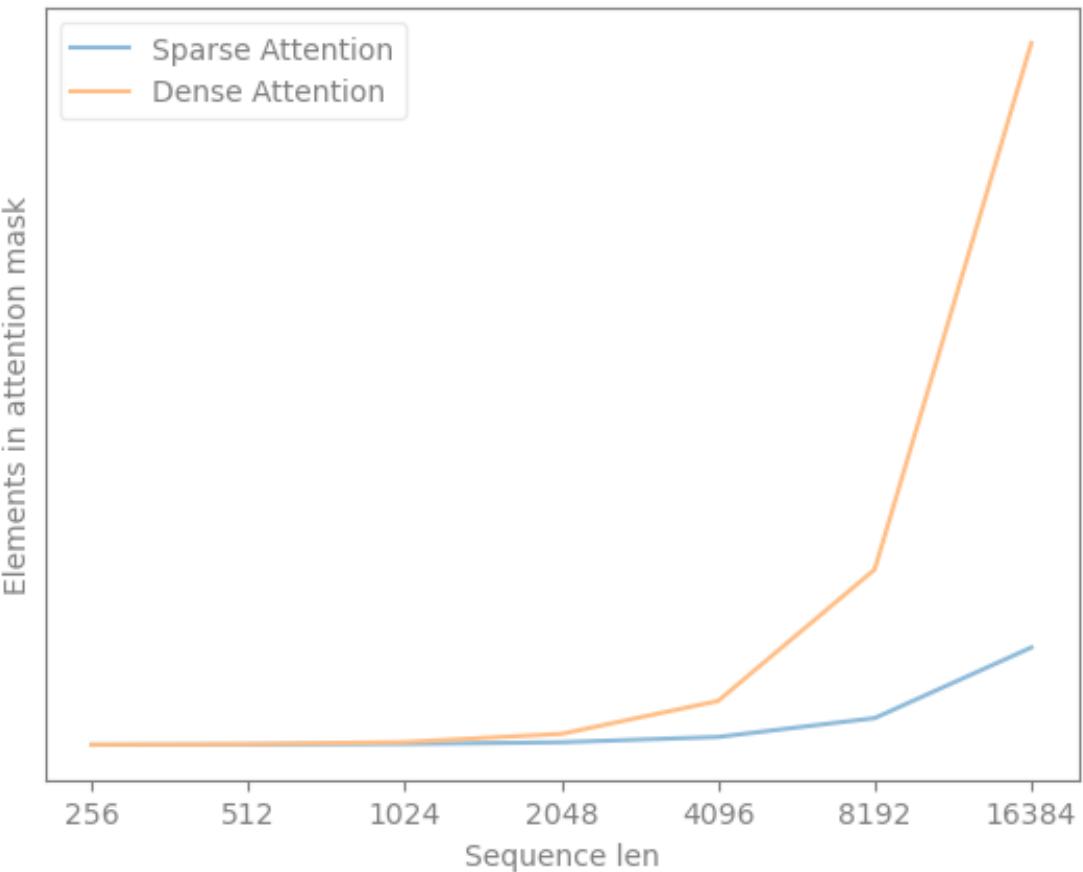
Хорошо ли работает sparse attention?

Model	Bits per byte	Time/Iter
Enwik8 (12,288 context)		
Dense Attention	1.00	1.31
Sparse Transformer (Fixed)	0.99	0.55



Хорошо ли работает sparse attention?

Model	Bits per byte	Time/Iter
Enwik8 (12,288 context)		
Dense Attention	1.00	1.31
Sparse Transformer (Fixed)	0.99	0.55
<hr/>		
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	



Другие способы оптимизаций



(GPT-3) Generating Long Sequences with Sparse Transformers: arxiv.org/abs/1904.10509



Reformer: The Efficient Transformer: arxiv.org/abs/2001.04451



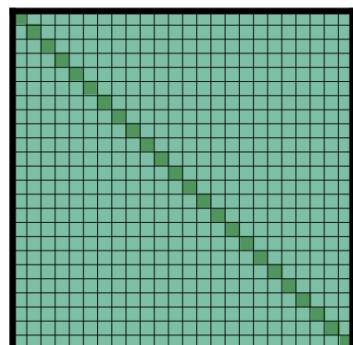
Longformer: The Long-Document Transformer: arxiv.org/abs/2004.05150



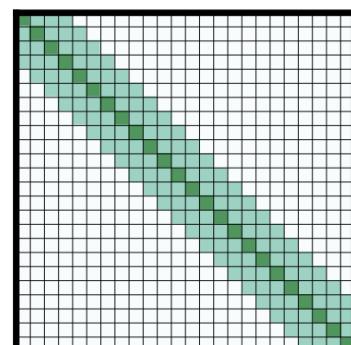
Linformer: Self-Attention with Linear Complexity: arxiv.org/abs/2006.04768



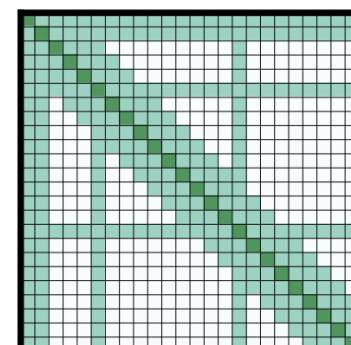
Rethinking Attention with Performers: arxiv.org/abs/2009.14794



(a) Full n^2 attention



(b) Sliding window attention



(d) Global+sliding window

Ho...

5.4. Saving memory by recomputing attention weights

Gradient checkpointing has been shown to be effective in reducing the memory requirements of training deep neural networks ([Chen et al., 2016](#)), ([Gruslys et al., 2016](#)). It is worth noting, however, that this technique is particularly effective for self-attention layers when long sequences are processed, as memory usage is high for these layers relative to the cost of computing them.

Using recomputation alone, we are able to train dense attention networks with hundreds of layers on sequence lengths of 16,384, which would be infeasible on modern hardware otherwise.

Ho...

5.4. Saving memory by recomputing attention weights

Gradient checkpointing has been shown to be effective in reducing the memory requirements of training deep neural networks (Chen et al., 2016), (Gruslys et al., 2016). It is worth noting, however, that this technique is particularly effective for self-attention layers when long sequences are processed, as memory usage is high for these layers relative to the cost of computing them.

Using recomputation alone, we are able to train dense attention networks with hundreds of layers on sequence lengths of 16,384, which would be infeasible on modern hardware otherwise.



Attention	Standard	FLASHATTENTION
GFLOPs	66.6	75.2
HBM R/W (GB)	40.3	4.4
Runtime (ms)	41.7	7.3

Ho...

5.4. Saving memory by recomputing attention weights

Gradient checkpointing has been shown to be effective in reducing the memory requirements of training deep neural networks (Chen et al., 2016), (Gruslys et al., 2016). It is worth noting, however, that this technique is particularly effective for self-attention layers when long sequences are processed, as memory usage is high for these layers relative to the cost of computing them.

Using recomputation alone, we are able to train dense attention networks with hundreds of layers on sequence lengths of 16,384, which would be infeasible on modern hardware otherwise.



Attention	Standard	FLASHATTENTION
GFLOPs	66.6	75.2
HBM R/W (GB)	40.3	4.4
Runtime (ms)	41.7	7.3

Model	Without FLASHATTENTION	FLASHATTENTION	FLASHATTENTION-2
GPT3-1.3B 2k context	142 TFLOPs/s	189 TFLOPs/s	196 TFLOPs/s
GPT3-1.3B 8k context	72 TFLOPs/s	170 TFLOPs/s	220 TFLOPs/s
GPT3-2.7B 2k context	149 TFLOPs/s	189 TFLOPs/s	205 TFLOPs/s
GPT3-2.7B 8k context	80 TFLOPs/s	175 TFLOPs/s	225 TFLOPs/s

Итоги

- GPT-3 в 115 раз больше GPT-2
- Тренировалась на 300B токенов

ИТОГИ

- GPT-3 в 115 раз больше GPT-2
- Тренировалась на 300B токенов
 - Начали использовать Common Crawl, но с агрессивной фильтрацией
 - High Quality-датасеты использовались больше одного раза

Итоги

- GPT-3 в 115 раз больше GPT-2
- Тренировалась на 300B токенов
 - Начали использовать Common Crawl, но с агрессивной фильтрацией
 - High Quality-датасеты использовались больше одного раза
- В 10 раз больше вычислительных мощностей, чем у других

Итоги

- GPT-3 в 115 раз больше GPT-2
- Тренировалась на 300B токенов
 - Начали использовать Common Crawl, но с агрессивной фильтрацией
 - High Quality-датасеты использовались больше одного раза
- В 10 раз больше вычислительных мощностей, чем у других
- Основное архитектурное изменение: Sparse Attention
 - цель — уменьшить потребление памяти при обучении с большим Seq Len
 - сейчас применяется редко (см. Flash Attention и Grad Checkpointing)

Directed by
IGOR D. KOTENKOV

Полная история семейства GPT: -1, 1, 2, 3...



Котенков Игорь



@stm

Модуль 2, часть 4. План:

- Как замеряют метрики, или что такое in-context learning
- Результаты в классических бенчмарках
- А понимает ли модель?
- GPT-3 в контексте AI Misuse

Как будем оценивать?

Traditional fine-tuning (not used for GPT-3)

Fine-tuning

The model is trained via repeated gradient updates using a large corpus of example tasks.



Как будем оценивать?

Traditional fine-tuning (not used for GPT-3)

Fine-tuning

The model is trained via repeated gradient updates on large corpuses of sample tasks.

1 sea => la mer example #1

1 ... gradient update

1 peppermint => ... example #2

1 ... gradient update

1 plush giraffe => girafe peluche example #N

gradient update

1 cheese => ... prompt

Zero-shot

The model predicts the answer given only a natural language description of the task. No gradient updates are performed.



Как будем оценивать?

Traditional fine-tuning (not used for GPT-3)

Fine-tuning

The model is trained via repeated gradient updates on large corpuses of sample tasks.

1 sea otter => loutre de mer

gradient update

1 peppermint => menthe poivrée

gradient update

1 plush giraffe => girafe peluche

gradient update

1 cheese => ...

Zero-shot

The model predicts the answer given only a natural language description of the task. No gradient updates are performed.

- 1 Translate English to French: ← *task description*
- 2 cheese => ... ← *prompt*

Few-shot

In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.

- 1 Translate English to French: ← *task description*
- 2 sea otter => loutre de mer ← *examples*
- 3 peppermint => menthe poivrée
- 4 plush giraffe => girafe peluche
- 5 cheese => ... ← *prompt*

Как будем оценивать?

Traditional fine-tuning (not used for GPT-3)

Fine-tuning

The model is trained via repeated gradient updates
large corpus of sample tasks.

1 sea otter => loutre de mer example #1

gradient update

1 peppermint => menthe poivrée example #2

gradient update

1 plush giraffe => girafe peluche example #N

gradient update

1 cheese => ... prompt

Zero-shot

The model predicts the answer given only a natural language description of the task. No gradient updates are performed.

- 1 Translate English to French: ← task description
- 2 cheese => ← prompt

One-shot

In addition to the task description, the model sees a single example of the task. No gradient updates are performed.

- 1 Translate English to French: ← task description
- 2 sea otter => loutre de mer ← example
- 3 cheese => ← prompt

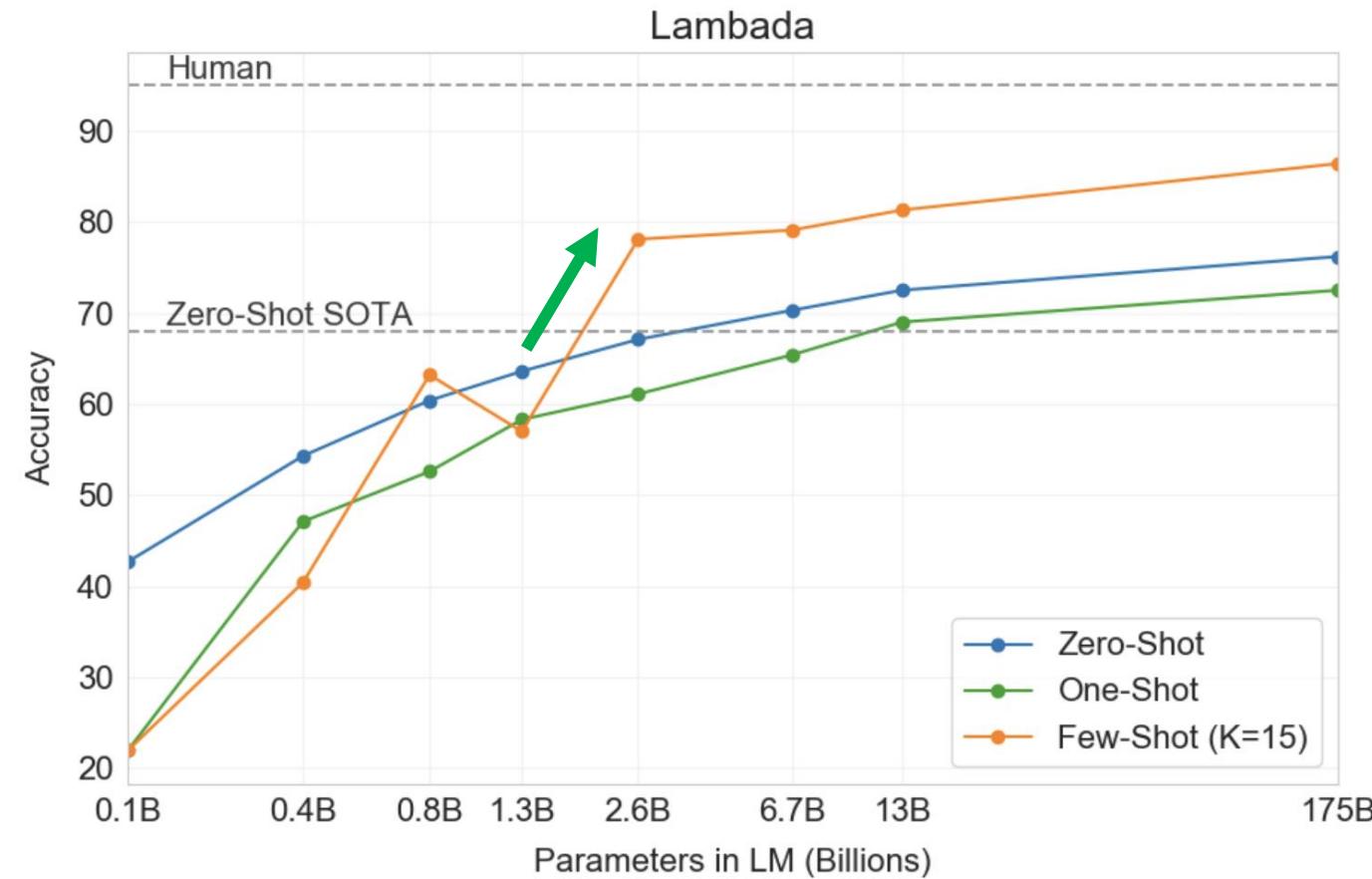
Few-shot

In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.

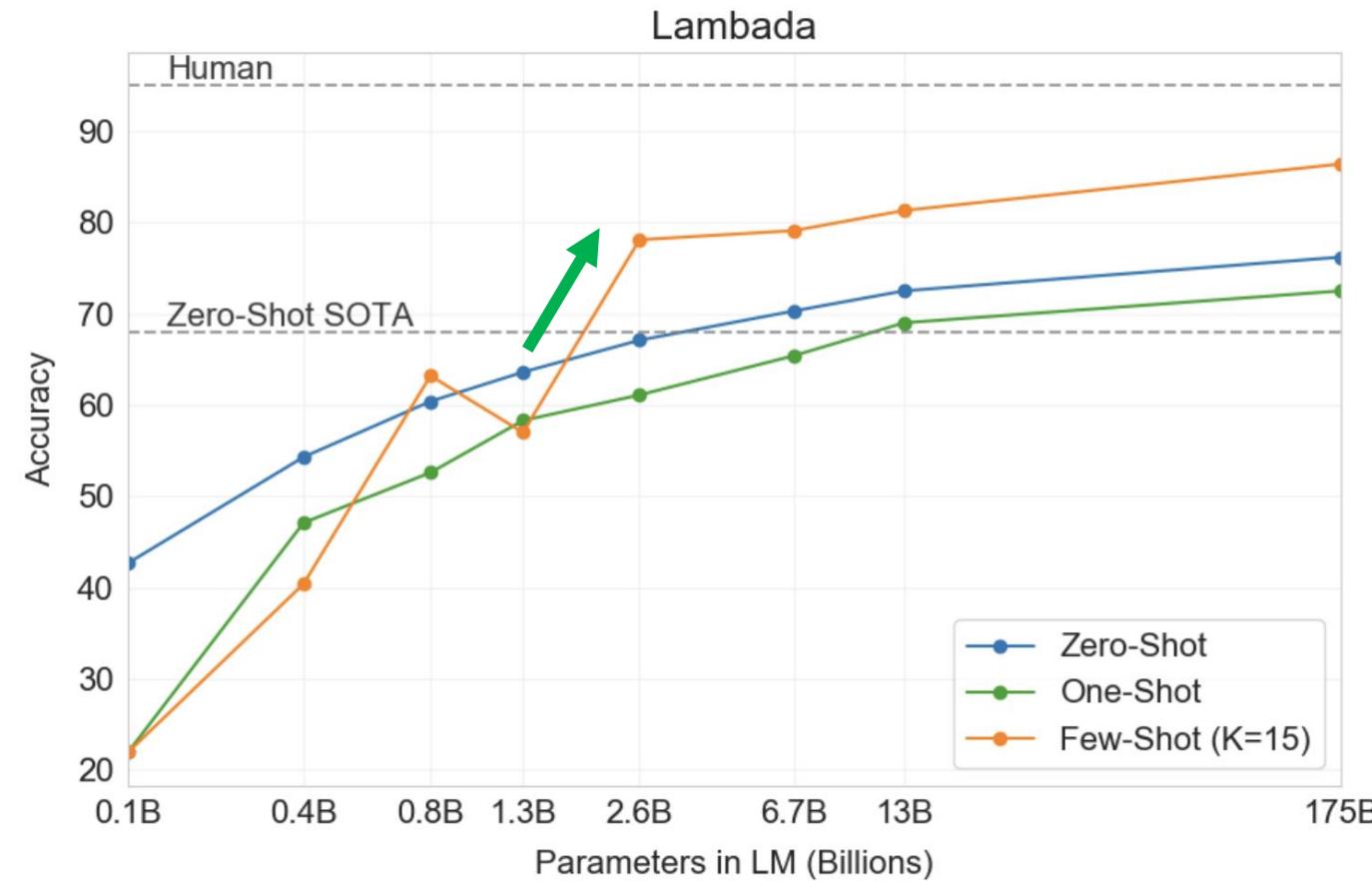
- 1 Translate English to French: ← task description
- 2 sea otter => loutre de mer ← examples
- 3 peppermint => menthe poivrée
- 4 plush girafe => girafe peluche
- 5 cheese => ← prompt

Также:
In-context learning

Языковое моделирование



Языковое моделирование



Alice was friends with Bob. Alice went to visit her friend _____. → Bob
George bought some baseball equipment, a ball, a glove, and a _____. →

Перевод

Setting	En→Fr	Fr→En	En→De	De→En	En→Ro	Ro→En
SOTA (Supervised)	45.6^a	35.0 ^b	41.2^c	40.2 ^d	38.5^e	39.9^e
XLM [LC19]	33.4	33.3	26.4	34.3	33.3	31.8
MASS [STQ ⁺ 19]	<u>37.5</u>	34.9	28.3	35.2	<u>35.2</u>	33.1
mBART [LGG ⁺ 20]	-	-	<u>29.8</u>	34.0	<u>35.0</u>	30.5
GPT-3 Zero-Shot	25.2	21.2	24.6	27.2	14.1	19.9
GPT-3 One-Shot	28.3	33.7	26.2	30.4	20.6	38.6
GPT-3 Few-Shot	32.6	<u>39.2</u>	29.7	<u>40.6</u>	21.0	<u>39.5</u>

BLEU scores

(вспоминаем токенайзер от GPT-2)

Перевод

Setting	En→Fr	Fr→En	En→De	De→En	En→Ro	Ro→En
SOTA (Supervised)	45.6^a	35.0 ^b	41.2^c	40.2 ^d	38.5^e	39.9^e
XLM [LC19]	33.4	33.3	26.4	34.3	33.3	31.8
MASS [STQ ⁺ 19]	<u>37.5</u>	34.9	28.3	35.2	<u>35.2</u>	33.1
mBART [LGG ⁺ 20]	-	-	<u>29.8</u>	34.0	<u>35.0</u>	30.5
GPT-3 Zero-Shot	25.2	21.2	24.6	27.2	14.1	19.9
GPT-3 One-Shot	28.3	33.7	26.2	30.4	20.6	38.6
GPT-3 Few-Shot	32.6	39.2	29.7	40.6	21.0	39.5

BLEU scores

(вспоминаем токенайзер от GPT-2)

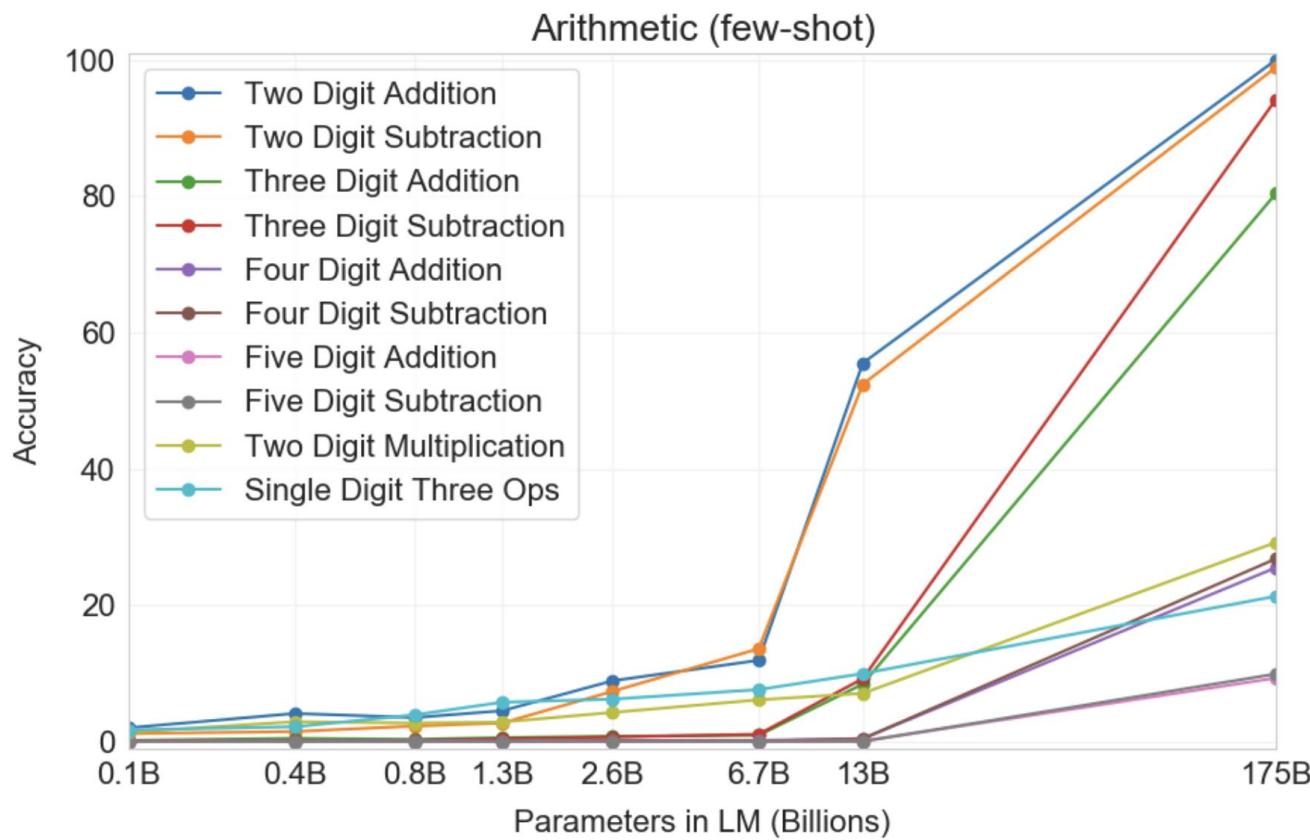
Перевод

Setting	En→Fr	Fr→En	En→De	De→En	En→Ro	Ro→En
SOTA (Supervised)	45.6^a	35.0 ^b	41.2^c	40.2 ^d	38.5^e	39.9^e
XLM [LC19]	33.4	33.3	26.4	34.3	33.3	31.8
MASS [STQ ⁺ 19]	<u>37.5</u>	34.9	28.3	35.2	<u>35.2</u>	33.1
mBART [LGG ⁺ 20]	-	-	<u>29.8</u>	34.0	<u>35.0</u>	30.5
GPT-3 Zero-Shot	25.2	21.2	24.6 →	27.2	14.1 →	19.9
GPT-3 One-Shot	28.3 →	33.7	26.2 →	30.4	20.6 →	38.6
GPT-3 Few-Shot	32.6 →	39.2	29.7 →	40.6	21.0 →	39.5

BLEU scores

(вспоминаем токенайзер от GPT-2)

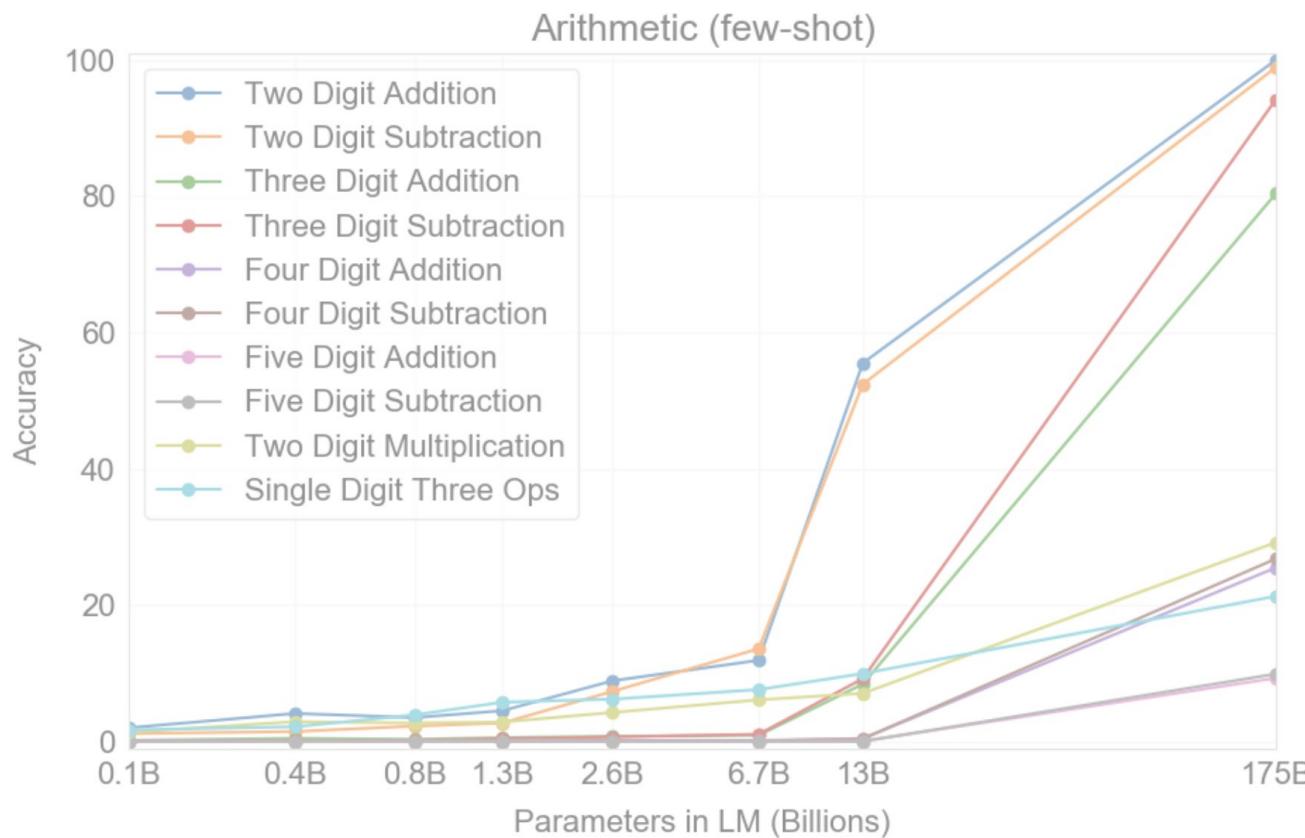
Синтетические задачи: математика



Setting	2D+	2D-	3D+	3D-	4D+	4D-
GPT-3 Zero-shot	76.9	58.0	34.2	48.3	4.0	7.5
GPT-3 One-shot	99.6	86.4	65.5	78.7	14.0	14.0
GPT-3 Few-shot	100.0	98.9	80.4	94.2	25.5	26.8

(для каждой группы 2000 случайных примеров)

Синтетические задачи: математика



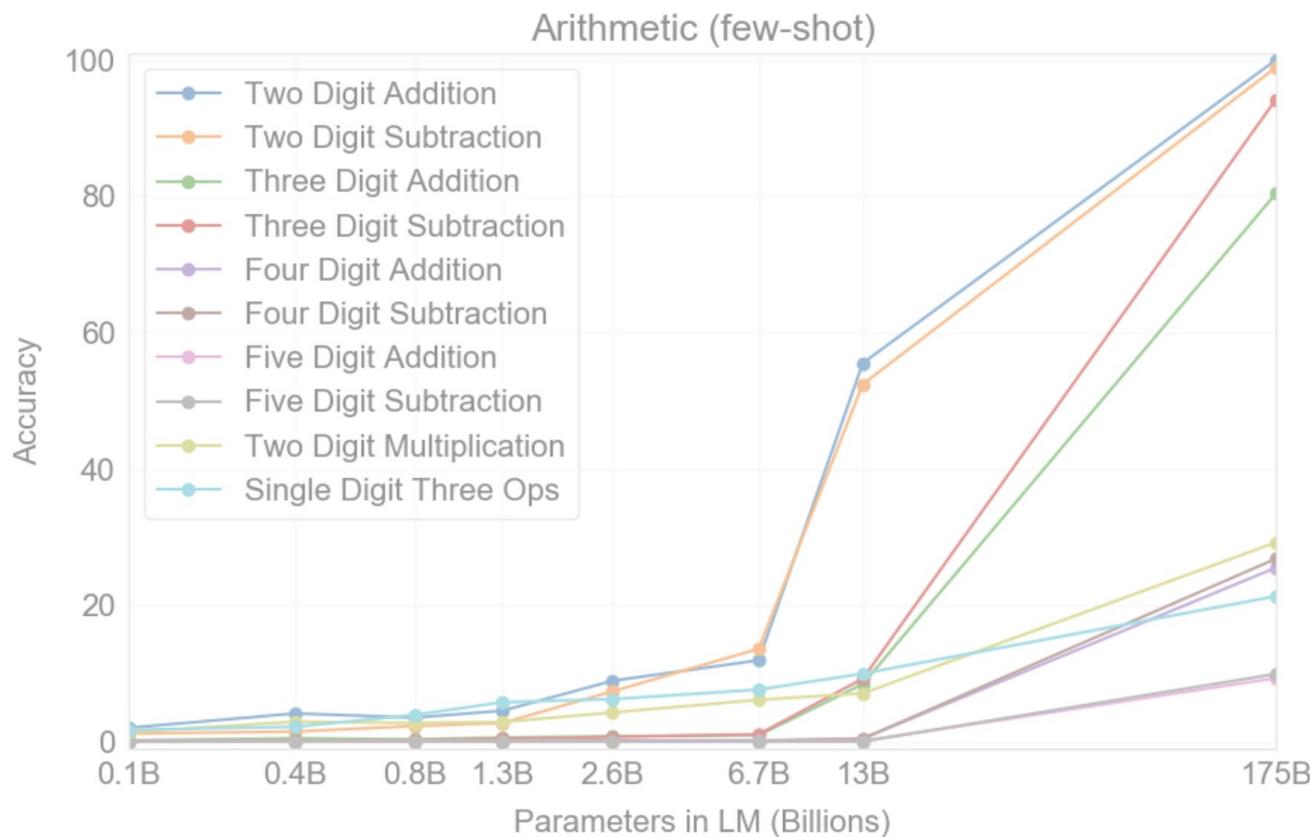
Setting	2D+	2D-	3D+	3D-	4D+	4D-
GPT-3 Zero-shot	76.9	58.0	34.2	48.3	4.0	7.5
GPT-3 One-shot	99.6	86.4	65.5	78.7	14.0	14.0
GPT-3 Few-shot	100.0	98.9	80.4	94.2	25.5	26.8

(для каждой группы 2000 случайных примеров)

Для трёхзначных операций:

- Сделали поиск по двум паттернам
 - "<NUM1> + <NUM2> ="
 - "<NUM1> plus <NUM2>"
- Доли пересечений с валидацией <1 % (~20 пар)

Синтетические задачи: математика



Setting	2D+	2D-	3D+	3D-	4D+	4D-
GPT-3 Zero-shot	76.9	58.0	34.2	48.3	4.0	7.5
GPT-3 One-shot	99.6	86.4	65.5	78.7	14.0	14.0
GPT-3 Few-shot	100.0	98.9	80.4	94.2	25.5	26.8

(для каждой группы 2000 случайных примеров)

Для трёхзначных операций:

- Сделали поиск по двум паттернам
 - "<NUM1> + <NUM2> ="
 - "<NUM1> plus <NUM2>"
- Доли пересечений с валидацией <1 % (~20 пар)
- Так что скорее всего не меморизация!
 - Да и она бы не объяснила прирост с in-context примерами

Question Answering

Setting	NaturalQS	WebQS	TriviaQA
<u>RAG</u> (Fine-tuned, Open-Domain) [LPP ⁺ 20]	44.5	45.5	68.0
T5-11B+SSM (Fine-tuned, Closed-Book) [RRS20]	36.6	44.7	60.5
T5-11B (Fine-tuned, Closed-Book)	34.5	37.4	50.1
GPT-3 Zero-Shot	14.6	14.4	64.3
GPT-3 One-Shot	23.0	25.3	68.0
GPT-3 Few-Shot	29.9	41.5	71.2

Question Answering

Setting	NaturalQS	WebQS	TriviaQA
RAG (Fine-tuned, Open-Domain) [LPP ⁺ 20]	44.5	45.5	68.0
T5-11B+SSM (Fine-tuned, Closed-Book) [RRS20]	36.6	44.7	60.5
T5-11B (Fine-tuned, Closed-Book)	34.5	37.4	50.1
GPT-3 Zero-Shot	14.6	14.4	64.3
GPT-3 One-Shot	23.0	25.3	68.0
GPT-3 Few-Shot	29.9	41.5	71.2

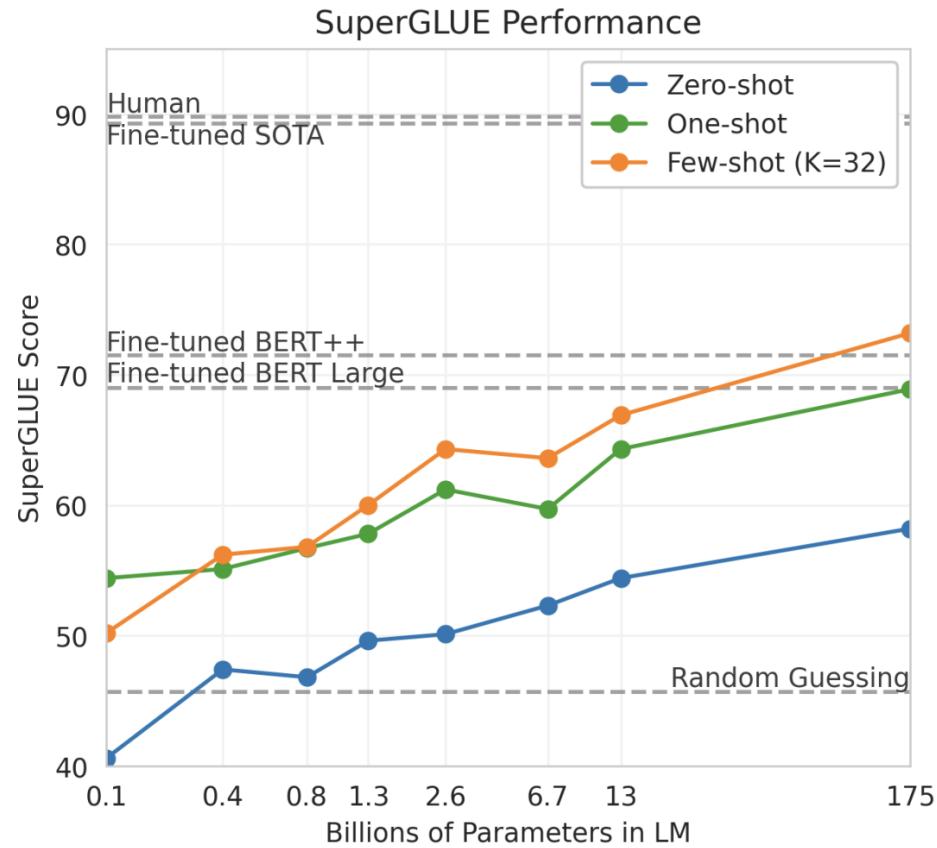
Context → Q: ‘Nude Descending A Staircase’ is perhaps the most famous painting by which 20th century artist?

A:

Target Completion → MARCEL DUCHAMP
Target Completion → r mutt
Target Completion → duchamp
Target Completion → marcel duchamp
Target Completion → R.Mutt
Target Completion → Marcel duChamp
Target Completion → Henri-Robert-Marcel Duchamp

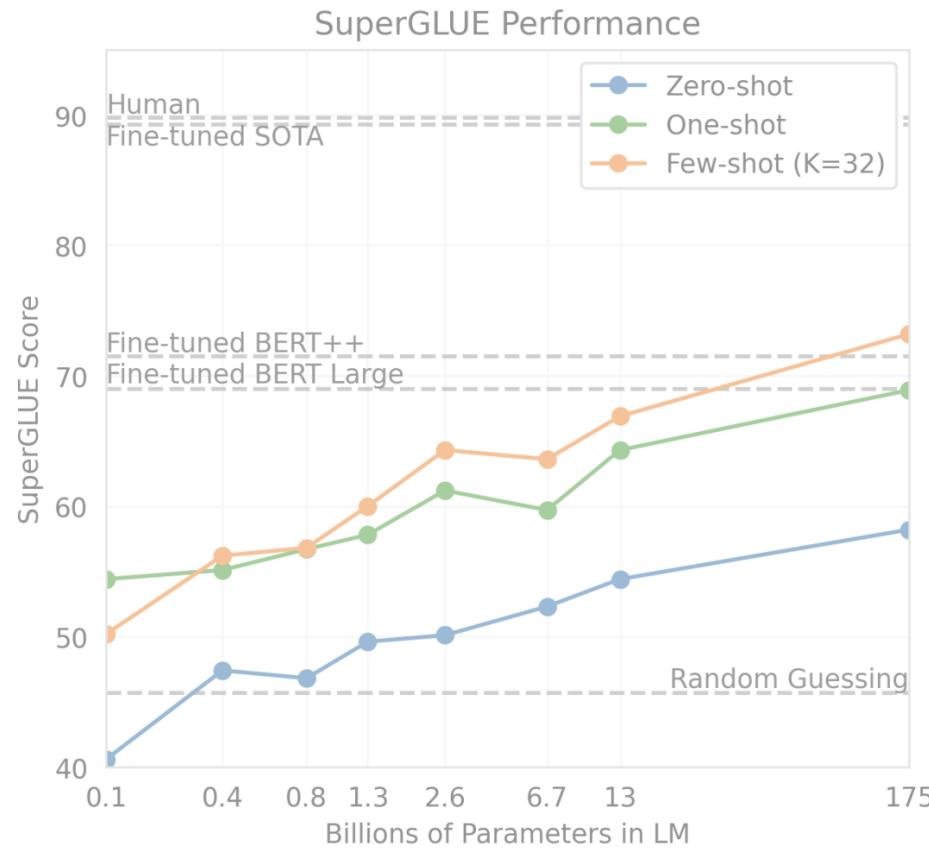
Пример из датасета TriviaQA

SuperGLUE (8 датасетов)

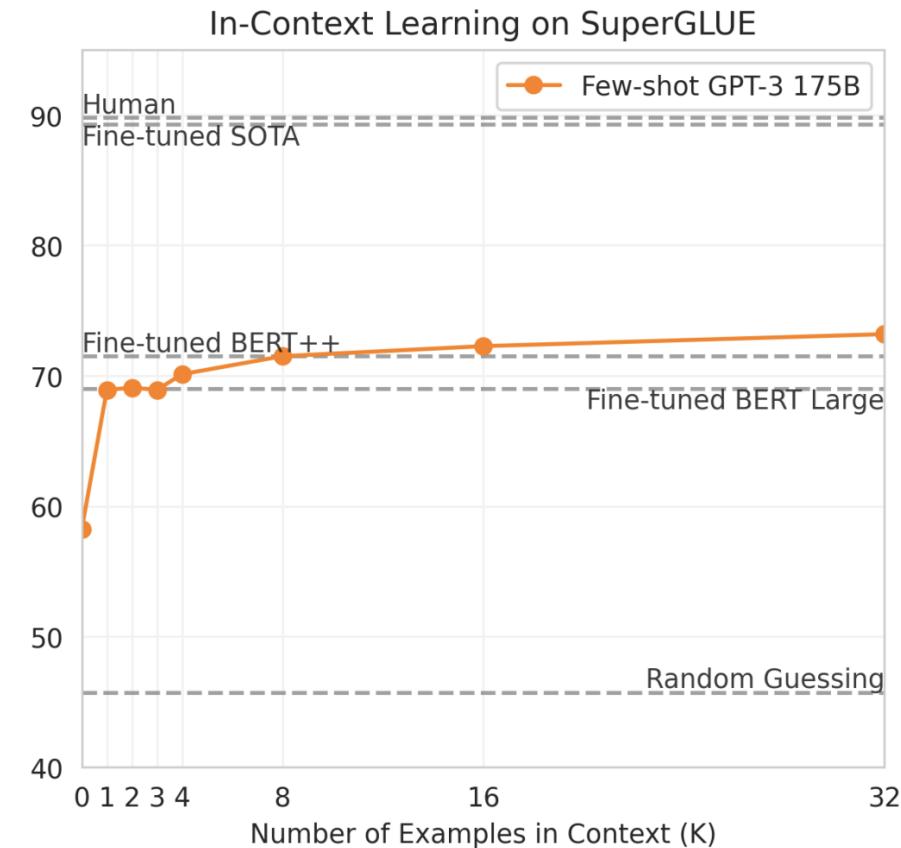


Обходим BERT++ всего с
256 (32 x 8) размеченными примерами!

SuperGLUE (8 датасетов)

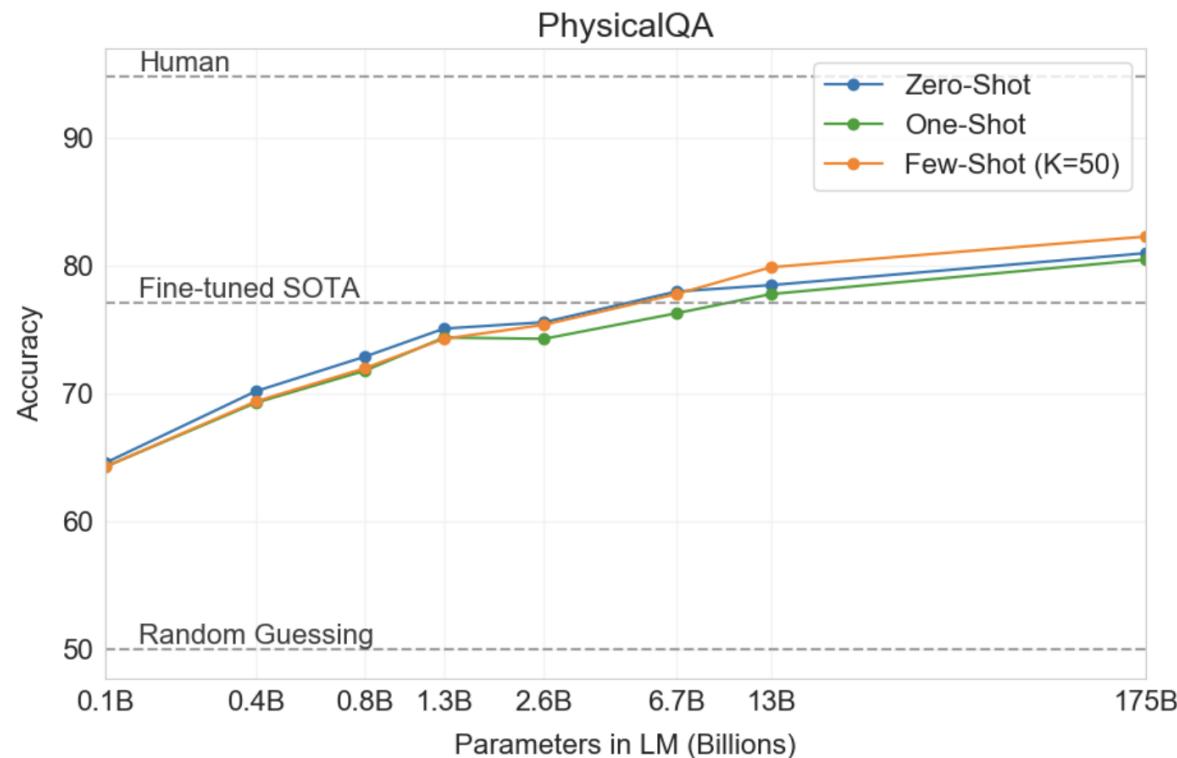


Обходим BERT++ всего с
256 (32 x 8) размеченными примерами!



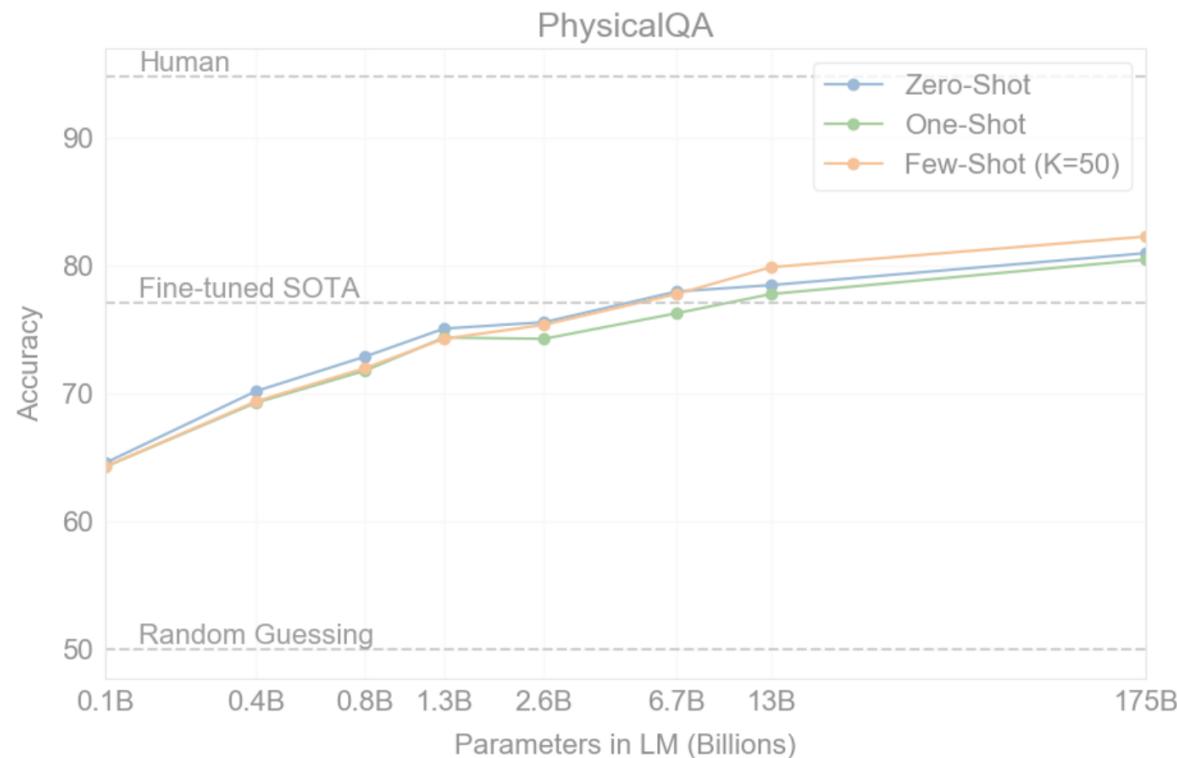
Чем больше контекста,
тем лучше качество

Common Sense Reasoning



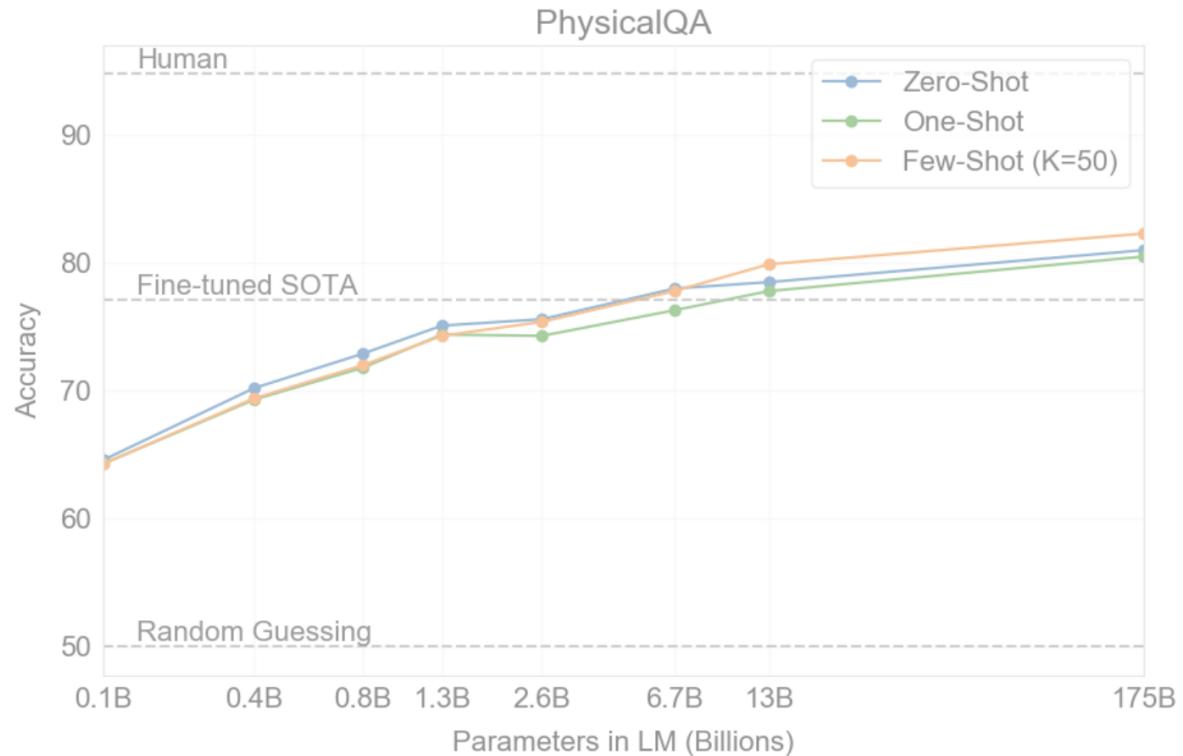
Setting	PIQA	ARC (Easy)	ARC (Challenge)	OpenBookQA
Fine-tuned SOTA	79.4	92.0 [KKS ⁺²⁰]	78.5 [KKS ⁺²⁰]	87.2 [KKS ⁺²⁰]
GPT-3 Zero-Shot	80.5*	68.8	51.4	57.6
GPT-3 One-Shot	80.5*	71.2	53.2	58.8
GPT-3 Few-Shot	82.8*	70.1	51.5	65.4

Common Sense Reasoning



Setting	PIQA	ARC (Easy)	ARC (Challenge)	OpenBookQA
Fine-tuned SOTA	79.4	92.0[KKS ⁺²⁰]	78.5[KKS ⁺²⁰]	87.2[KKS ⁺²⁰]
GPT-3 Zero-Shot	80.5*	68.8	51.4	57.6
GPT-3 One-Shot	80.5*	71.2	53.2	58.8
GPT-3 Few-Shot	82.8*	70.1	51.5	65.4

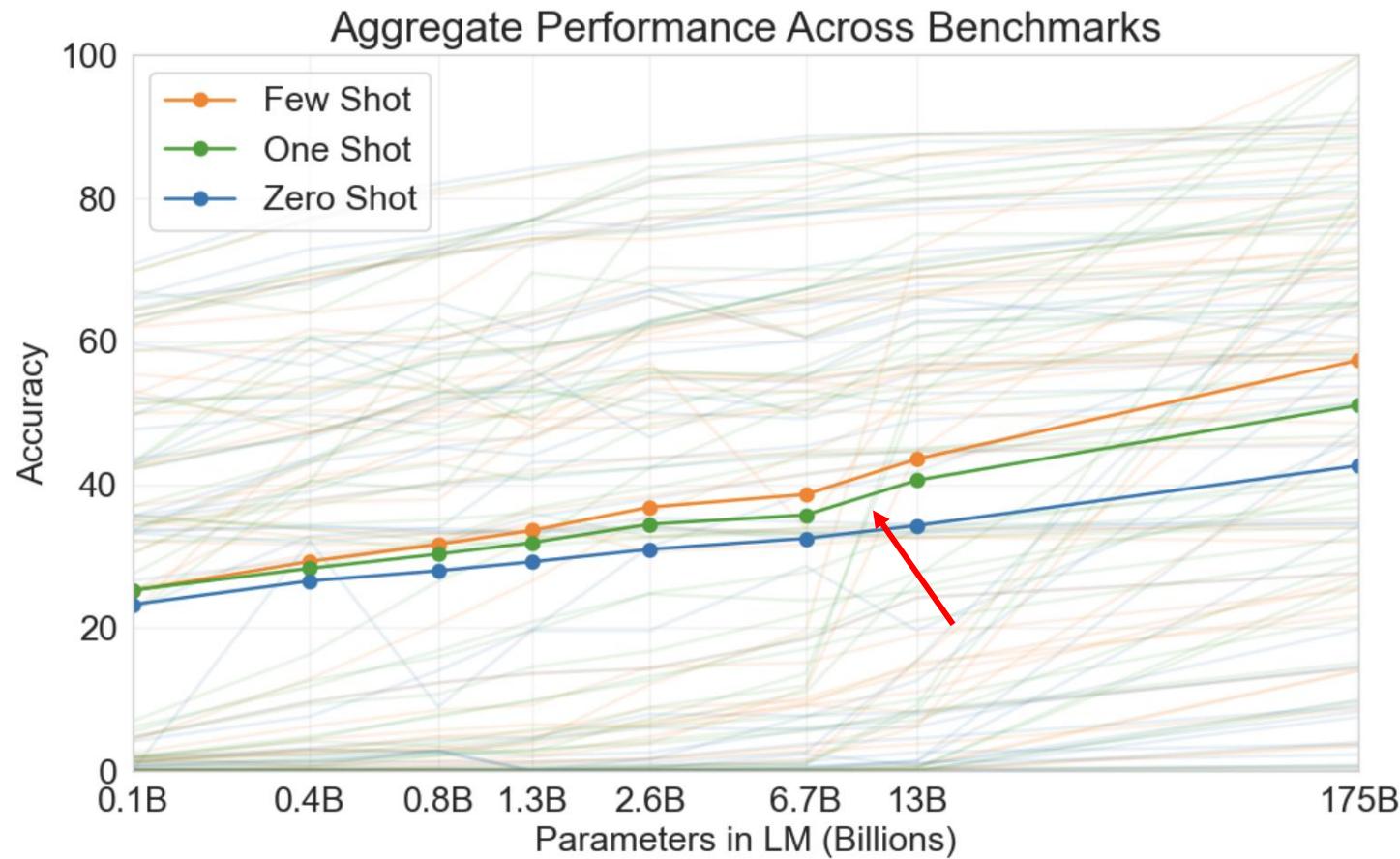
Common Sense Reasoning



“ Unfortunately, a bug in the filtering caused us to ignore some overlaps, and due to the cost of training it was **not feasible to retrain the model** ”

Setting	PIQA	ARC (Easy)	ARC (Challenge)	OpenBookQA
Fine-tuned SOTA	79.4	92.0[KKS ⁺²⁰]	78.5[KKS ⁺²⁰]	87.2[KKS ⁺²⁰]
GPT-3 Zero-Shot	80.5*	68.8	51.4	57.6
GPT-3 One-Shot	80.5*	71.2	53.2	58.8
GPT-3 Few-Shot	82.8*	70.1	51.5	65.4

Агрегированные метрики



Усреднённые результаты на 42 задачах
И всё — без обучения с SGD!

Запоминание или понимание?

p language is a language where every word is appended with p.

Q: you are a good boy.

A: youp arep ap goodp boyp.

Q: Where is the spoon?

A: Wherеп isp thep spoонp?

Q: The cat is a small animal.

A: Thep catp isp ap smallp animalp.

Q: Can he jump?

A: Canp hep jumpp?

Q: The forest was big. There were lots of animals and trees. But no humans.

A: Thep forestp wasp bigp. Therеп werеп lotsp ofp animalsp andp treesp. Butp nopp humansp.

Q: Can the lady call the guy? There is an emergency.

A: Canp thep ladyp callp thep guyp? Therеп isp anp emergenncyp.

Q: Thank the heavens. Peter pan went to his pop.

A: Thankp thep heavensp. Peterp panp wentp topp hisp popp.



Is GPT-3 really doing few shot learning?

Запоминание или понимание?

p language is a language where only animal names are appended with p. Cat, dog, tiger etc are some animal names.

Q: you are a good boy.

A: you are a good boy.

Q: Where is the spoon?

A: Where is the spoon?

Q: The cat is a small animal.

A: The catp is a small animal.

Q: Can he jump over the dog?

A: Can he jump over the dogp?

Q: What is the time?

A: What is the time?

Q: The tiger is dangerous?

A: The tigerp is dangerous?

Q: The boy is going to school.

A: The boyp is going to school.

Запоминание или понимание?

p language is a language where only animal names are appended with p. Cat, dog, tiger etc are some animal names.

Q: you are a good boy.

A: you are a good boy.

Q: Where is the spoon?

A: Where is the spoon?

Q: The cat is a small animal.

A: The catp is a small animal.

Q: Can he jump over the dog?

A: Can he jump over the dogp?

Q: What is the time?

A: What is the time?

Q: The tiger is dangerous?

A: The tigerp is dangerous?

Q: The boy is going to school.

A: The boyp is going to school.



Ну Интернет — большой!

Запоминание или понимание?

A "whatpu" is a small, furry animal native to Tanzania. An example of a sentence that uses the word whatpu is:

We were traveling in Africa and we saw these very cute whatpus.

To do a "farduddle" means to jump up and down really fast. An example of a sentence that uses the word farduddle is:

One day when I was playing tag with my little sister, she got really excited and she started doing these crazy farduckles.

A "yalubalu" is a type of vegetable that looks like a big pumpkin. An example of a sentence that uses the word yalubalu is:

I was on a trip to Africa and I tried this yalubalu vegetable that was grown in a garden there. It was delicious.

A "Burringo" is a car with very fast acceleration. An example of a sentence that uses the word Burringo is:

In our garage we have a Burringo that my father drives to work every day.

A "Gigamuru" is a type of Japanese musical instrument. An example of a sentence that uses the word Gigamuru is:

I have a Gigamuru that my uncle gave me as a gift. I love to play it at home.

To "screeg" something is to swing a sword at it. An example of a sentence that uses the word screeg is:

We screeghed at each other for several minutes and then we went outside and ate ice cream.

Чудеса in-context learning

```
Input: 94, 47, 84, 31, output = 2
```

```
Input: 89, 51, 73, 31, output = 1
```

```
[...]
```

```
Input: 96, 51, 80, 38, output = 2
```

```
Input: 90, 37, 76, 27, output =
```

Чудеса in-context learning

```
Input: 94, 47, 84, 31, output = 2  
Input: 89, 51, 73, 31, output = 1  
[...]  
Input: 96, 51, 80, 38, output = 2  
Input: 90, 37, 76, 27, output =
```

$$x_{\text{new}} = \text{round}(14x_{\text{old}} + 6)$$

Model	Accuracy
kNN	95.73%
Logistic regr.	96.26%
Ada	89.86%
Babbage	93.06%
Curie	95.20%
Davinci	95.73%

Чудеса in-context learning

Input: 94, 47, 84, 31, output = 2

Input: 89, 51, 73, 31, output = 1

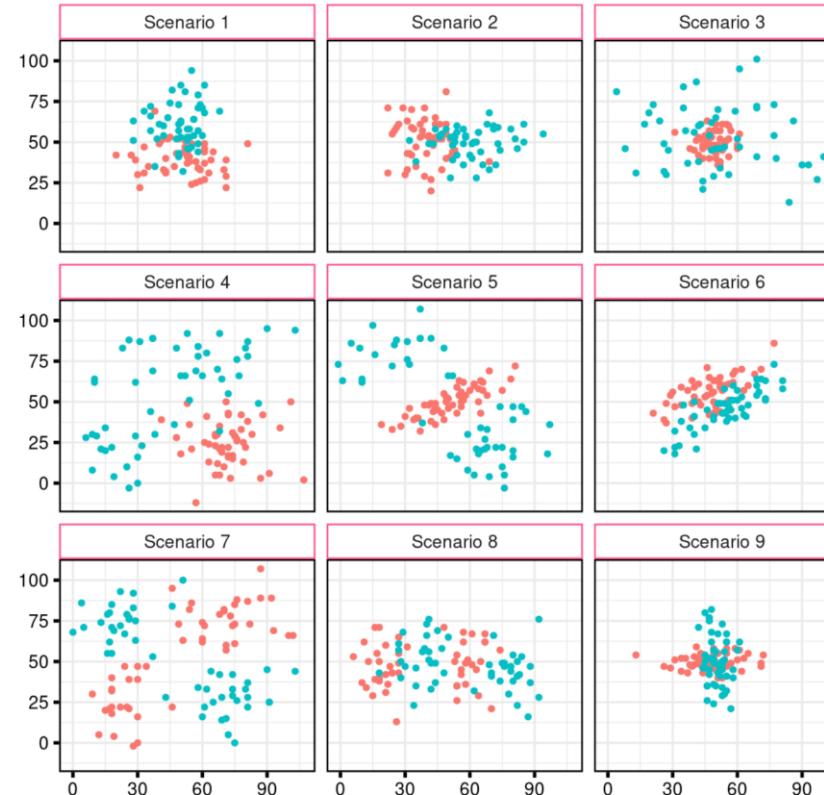
[...]

Input: 96, 51, 80, 38, output = 2

Input: 90, 37, 76, 27, output =

$$x_{\text{new}} = \text{round}(14x_{\text{old}} + 6)$$

Model	Accuracy
kNN	95.73%
Logistic regr.	96.26%
Ada	89.86%
Babbage	93.06%
Curie	95.20%
Davinci	95.73%

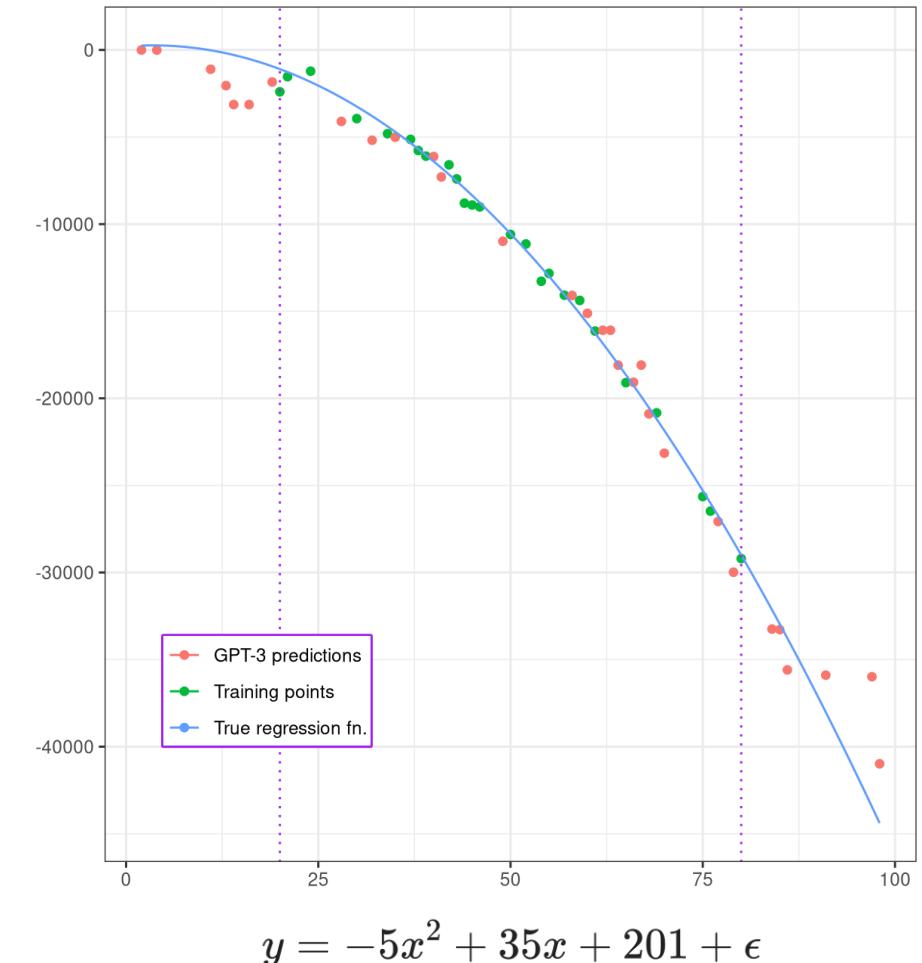
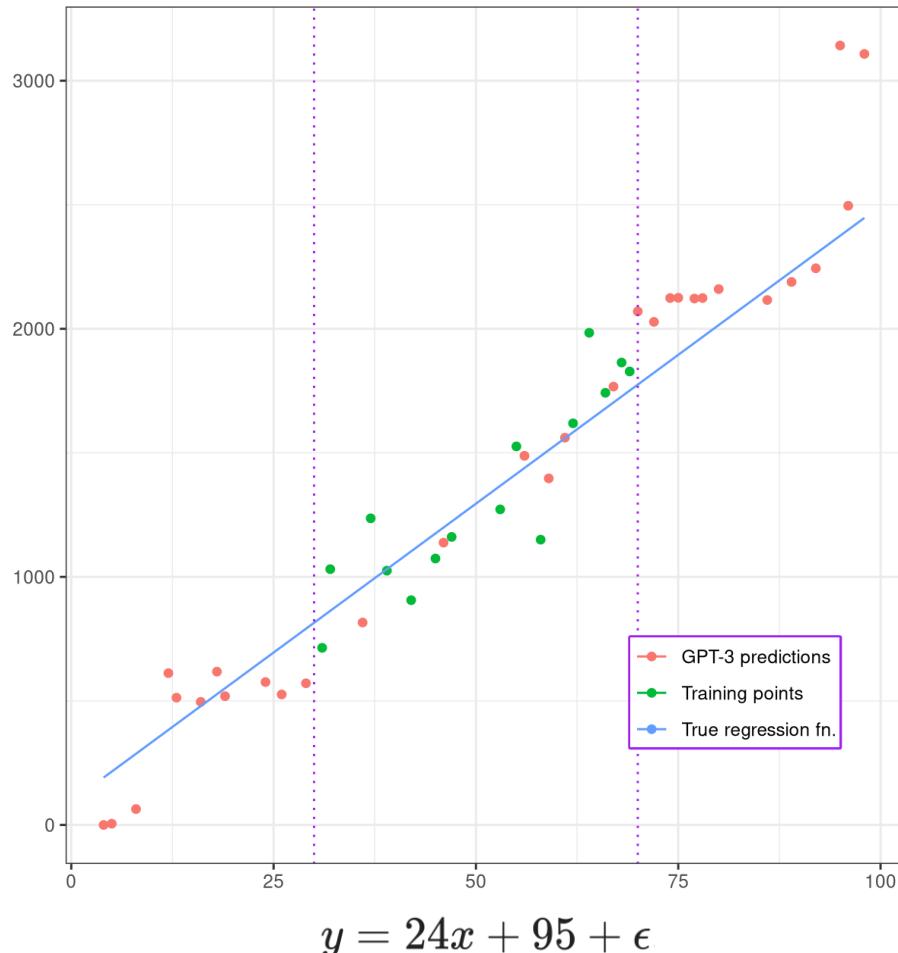


Model	Average acc.
kNN	81.78%
Logistic regr.	62.34%
Custom text	67.03%
Ada	73.70%
Babbage	72.10%
Curie	75.68%
Davinci	75.93%



Who models the models that model models?

Чудеса in-context learning



Who models the models that model models?

Генерация новостных статей

Генерации по заголовкам
newser.com (~200 слов)

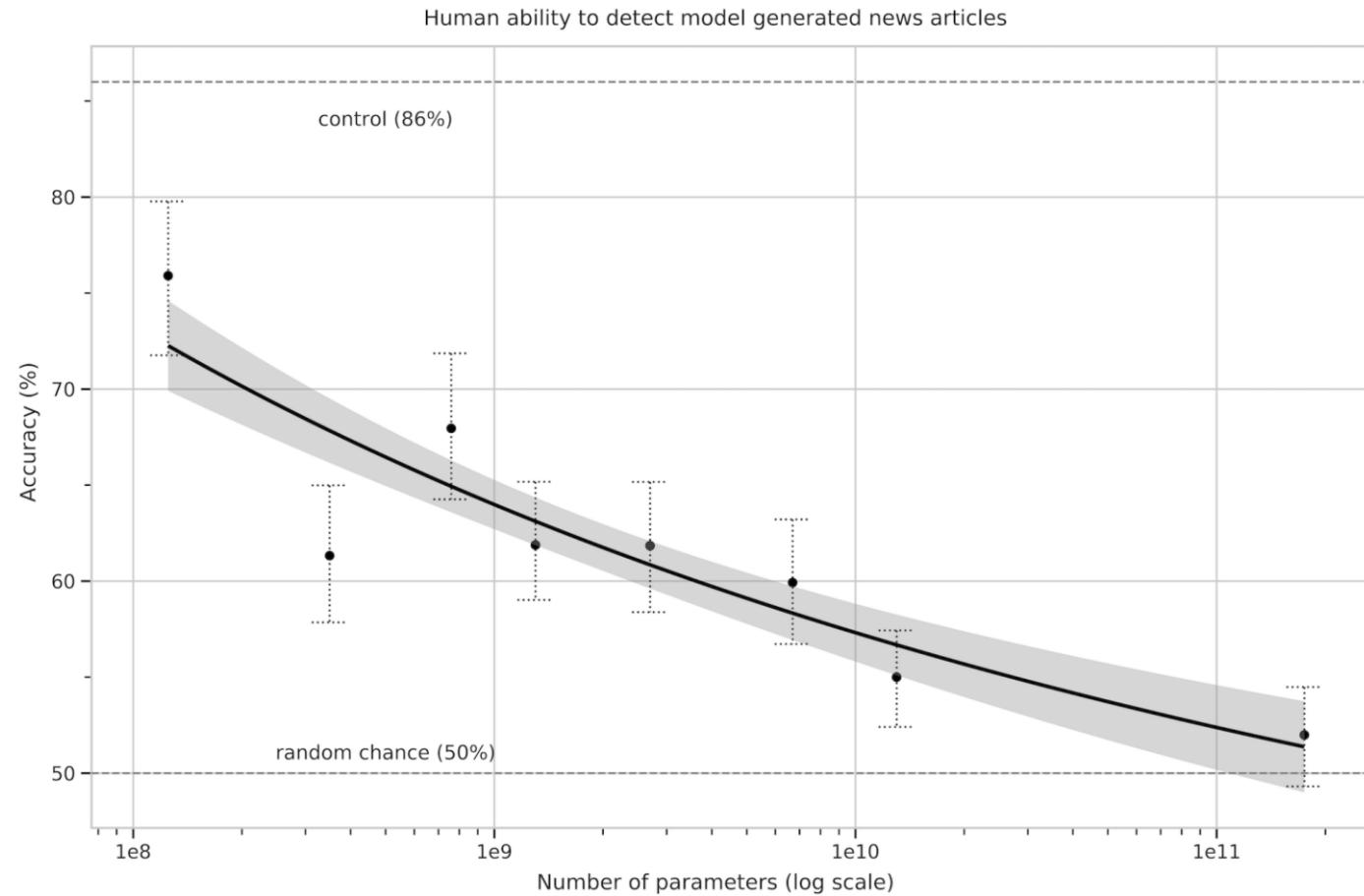


Выборки из ~80 граждан US



Генерации по заголовкам
Reuters (~500 слов)

Генерация новостных статей



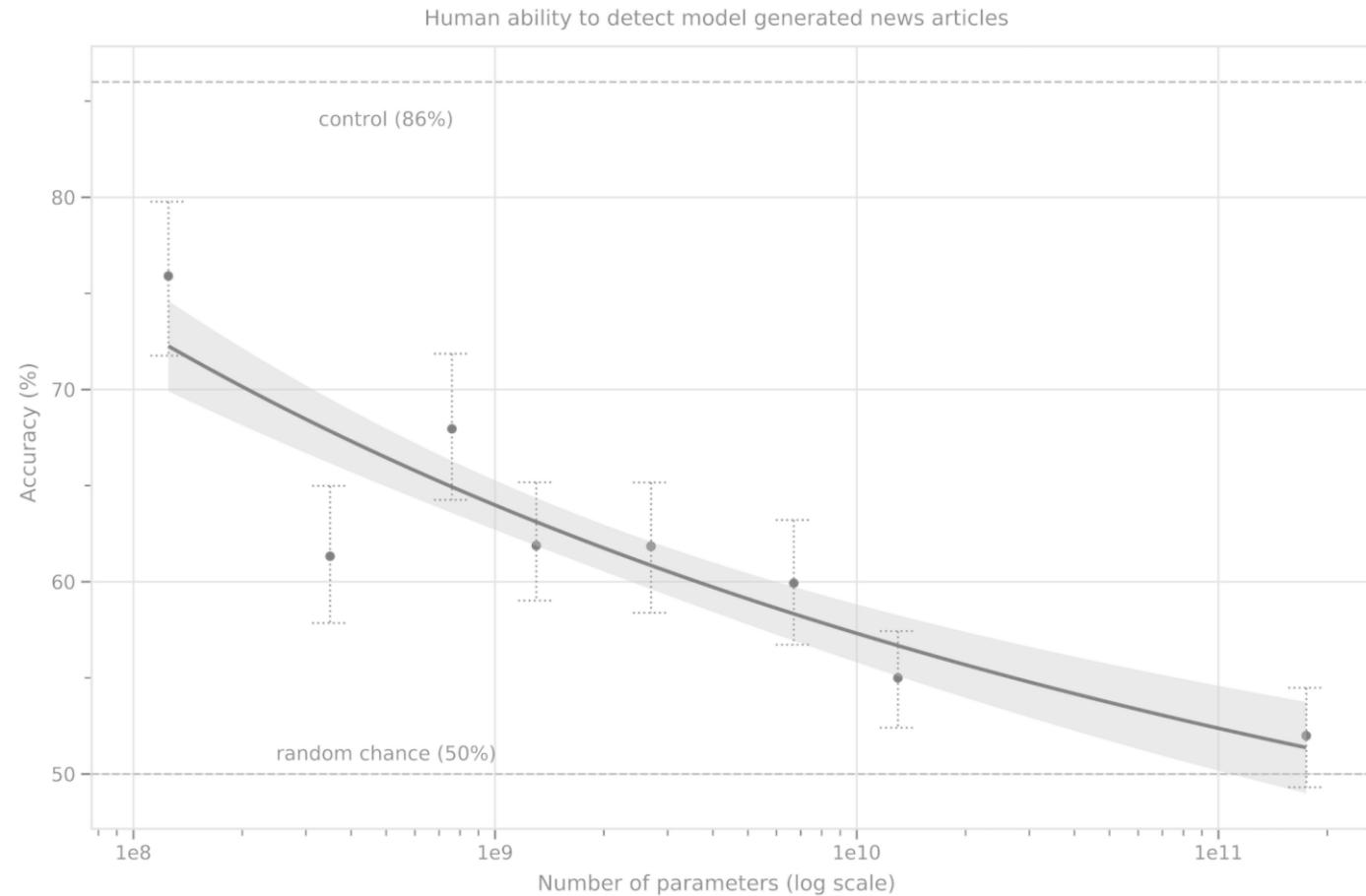
Генерации по заголовкам
newser.com (~200 слов)

Выборки из ~80 граждан US

Генерации по заголовкам
Reuters (~500 слов)

A large black curved arrow originates from the right side of the graph and points towards the text "Генерации по заголовкам newser.com (~200 слов)".

Генерация новостных статей



Генерации по заголовкам
newser.com (~200 слов)

Выборки из ~80 граждан US

Генерации по заголовкам
Reuters (~500 слов)

	Mean accuracy	95% Confidence Interval (low, hi)	t compared to control (p-value)	"I don't know" assignments
Control	88%	84%–91%	-	2.7%
GPT-3 175B	52%	48%–57%	12.7 (3.2e-23)	10.6%

ИТОГИ

- Вместо тренировки модели используем **in-context learning** (vs GPT-1)
 - Zero-shot, One-shot и Few-shot

ИТОГИ

- Вместо тренировки модели используем **in-context learning** (vs GPT-1)
 - Zero-shot, One-shot и Few-shot
- Вместо измерения вероятностей токенов генерируем ответы (vs GPT-2)

Итоги

- Вместо тренировки модели используем **in-context learning** (vs GPT-1)
 - Zero-shot, One-shot и Few-shot
- Вместо измерения вероятностей токенов генерируем ответы (vs GPT-2)
- SOTA на языковом моделировании и переводе (на английский)
- Чем более полон контекст, тем лучше

ИТОГИ

- Вместо тренировки модели используем **in-context learning** (vs GPT-1)
 - Zero-shot, One-shot и Few-shot
- Вместо измерения вероятностей токенов генерируем ответы (vs GPT-2)
- SOTA на языковом моделировании и переводе (на английский)
- Чем более полон контекст, тем лучше
- Очень сложно искать примеры «невыученных» задач
 - ...но что тестируем — то в целом работает хоть как-то

Итоги

- Вместо тренировки модели используем **in-context learning** (vs GPT-1)
 - Zero-shot, One-shot и Few-shot
- Вместо измерения вероятностей токенов генерируем ответы (vs GPT-2)
- SOTA на языковом моделировании и переводе (на английский)
- Чем более полон контекст, тем лучше
- Очень сложно искать примеры «невыученных» задач
 - ...но что тестируем — то в целом работает хоть как-то
- Генерация fake news, неотличимых от реальных

Directed by
IGOR D. KOTENKOV

Полная история семейства GPT: -1, 1, 2, 3...



Котенков Игорь



@stm

Модуль 2, часть 5. План:

- Что такое scaling и что мы масштабируем
- Почему трансформеры скейлятся лучше?
- Как оценить оптимум, не разорившись на GPU?
- Правда ли, что меньше — лучше?
- Чего боятся в OpenAI?

Почему мы масштабируем модели?

“

Ilya Sutskever was an early advocate for scaling large generative likelihood models

”



Почему мы масштабируем модели?

“

Ilya Sutskever was an early advocate for scaling large generative likelihood models

”



“

It was just before OpenAI started that I met Ilya. One of the first things he said to me was:
— Look. The models, **they just want to learn**. You have to understand this. **You get the obstacles out of their way**.



Почему мы масштабируем модели?

“

Ilya Sutskever was an early advocate for scaling large generative likelihood models

”



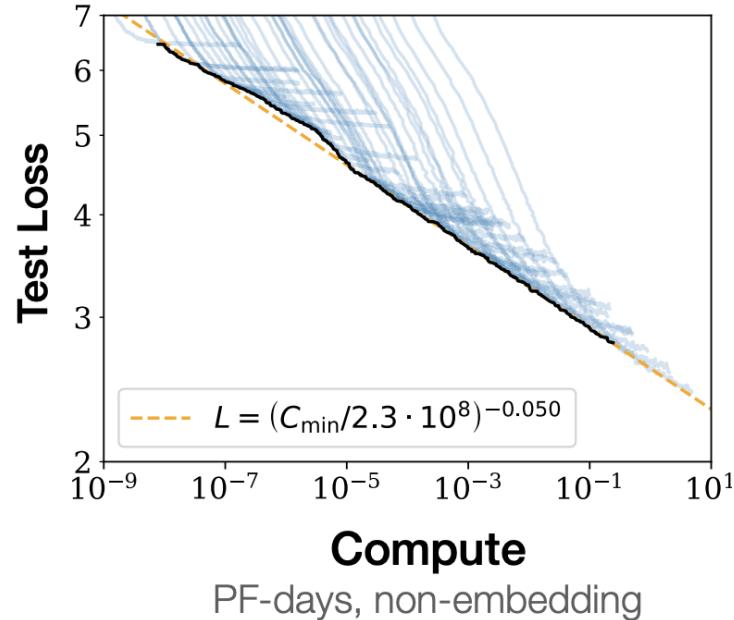
“

It was just before OpenAI started that I met Ilya. One of the first things he said to me was:
— Look. The models, **they just want to learn**. You have to understand this. You get the obstacles out of their way.

And it was a bit like a Zen Koan. I listened to this and **I became enlightened**.



Scaling Laws for Neural Language Models



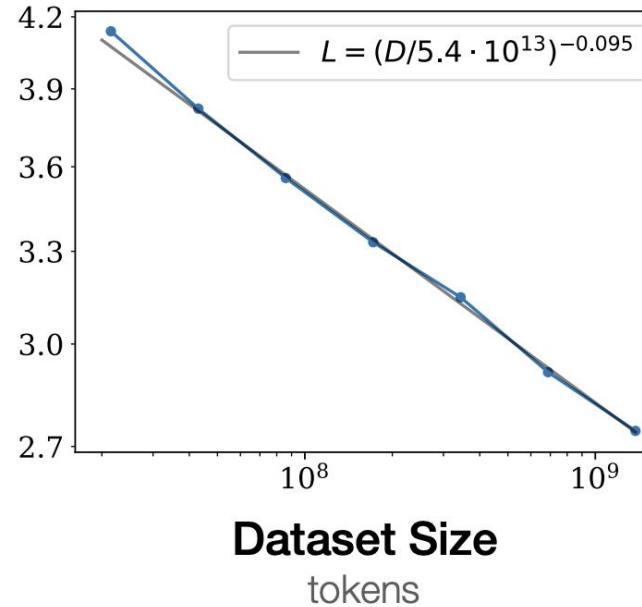
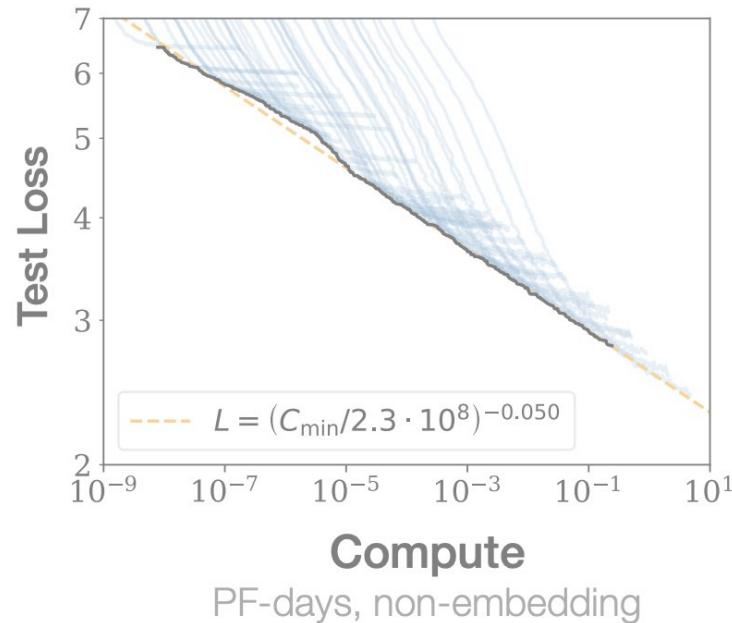
Фиксируем

Размер модели

Представляем достаточно

Данных

Scaling Laws for Neural Language Models



Фиксируем

Размер модели

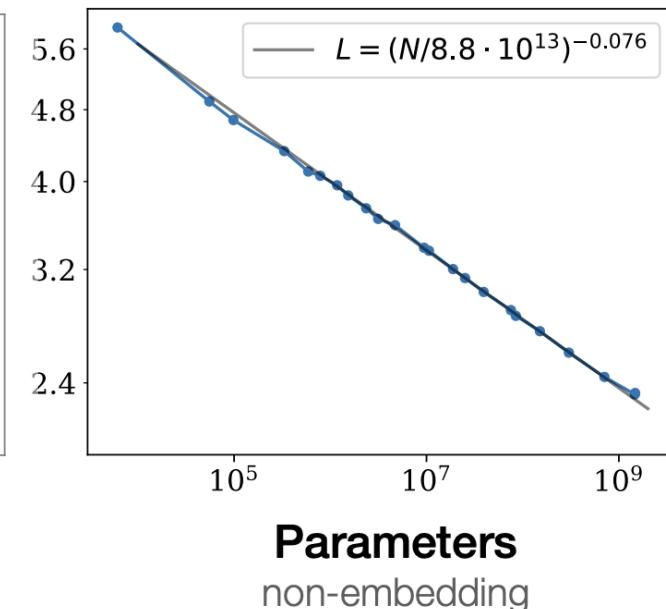
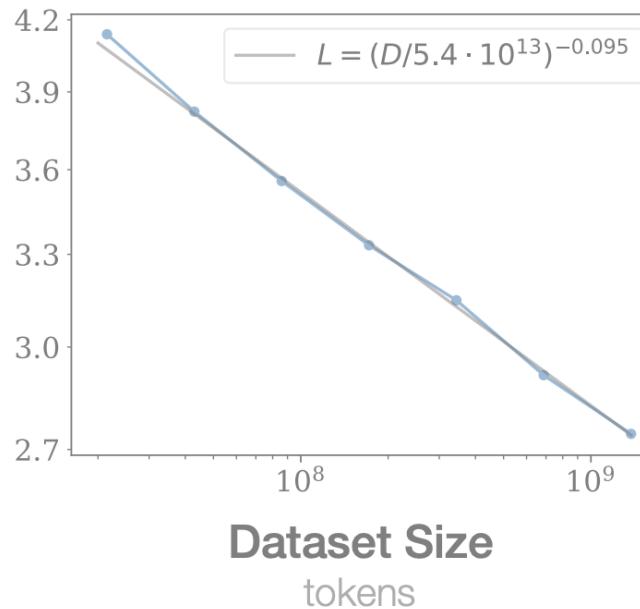
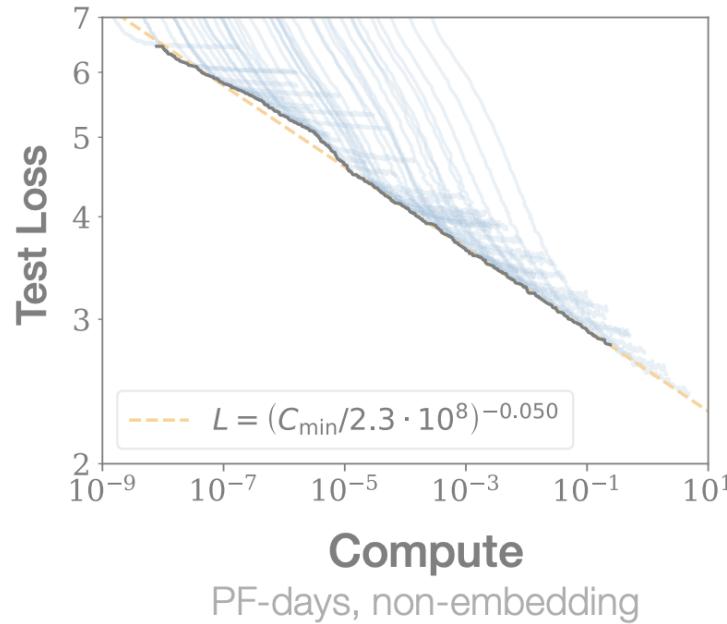
Размер модели

Предоставляем достаточно

Данных

Количество мощностей

Scaling Laws for Neural Language Models



Фиксируем

Размер модели

Размер модели

Количество мощностей

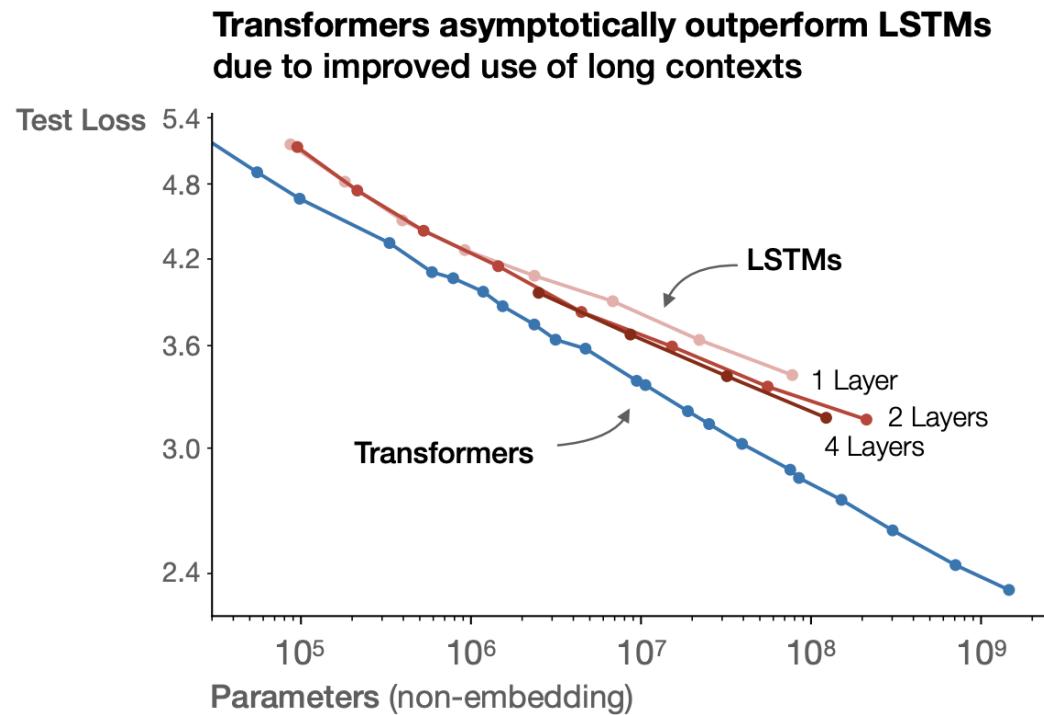
Предоставляем достаточно

Данных

Количество мощностей

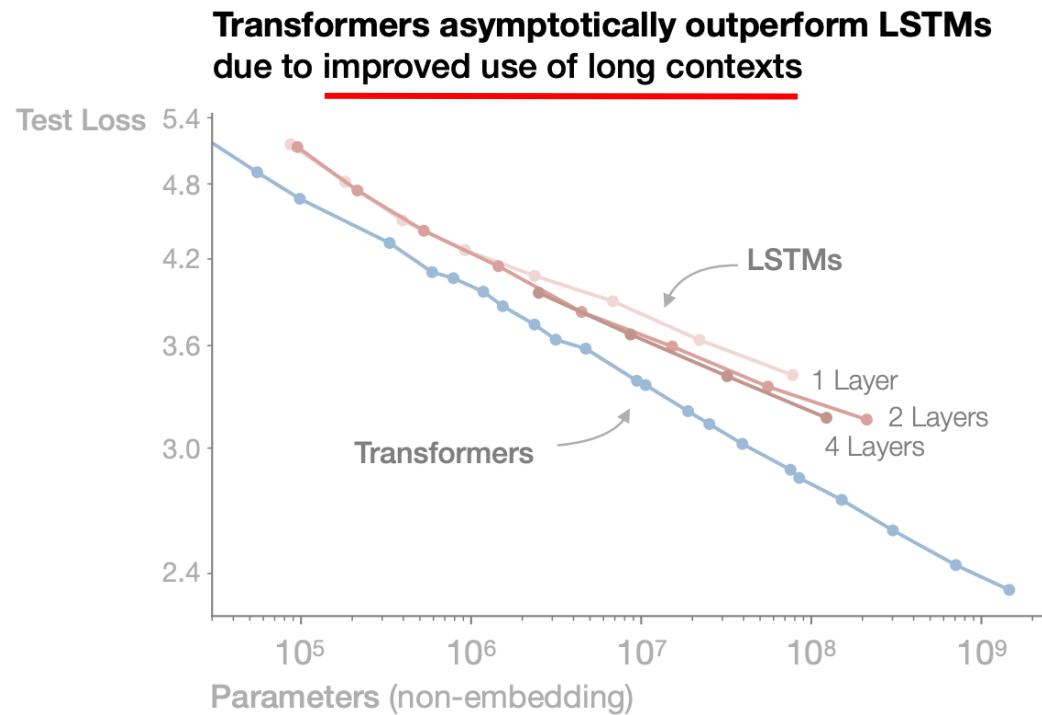
Данных

Хотим скейлить то, что скейлится лучше

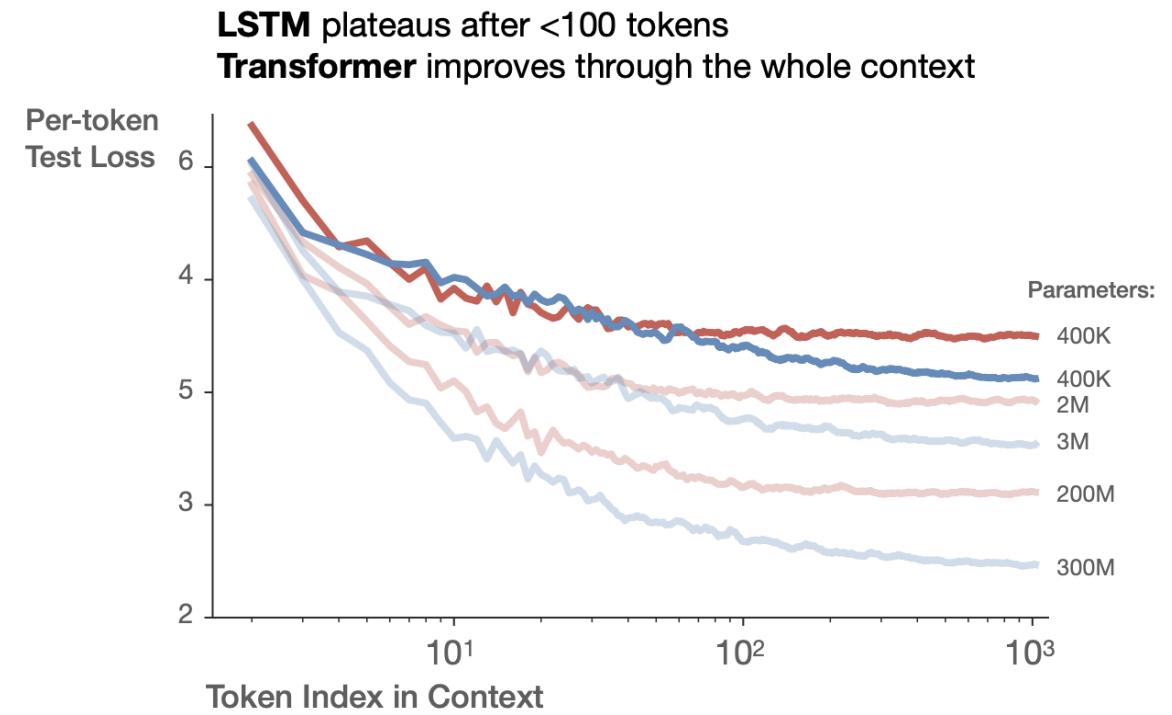


Обучение на том же датасете,
с одинаковой длинной контекста

Хотим скейлить то, что скейлится лучше

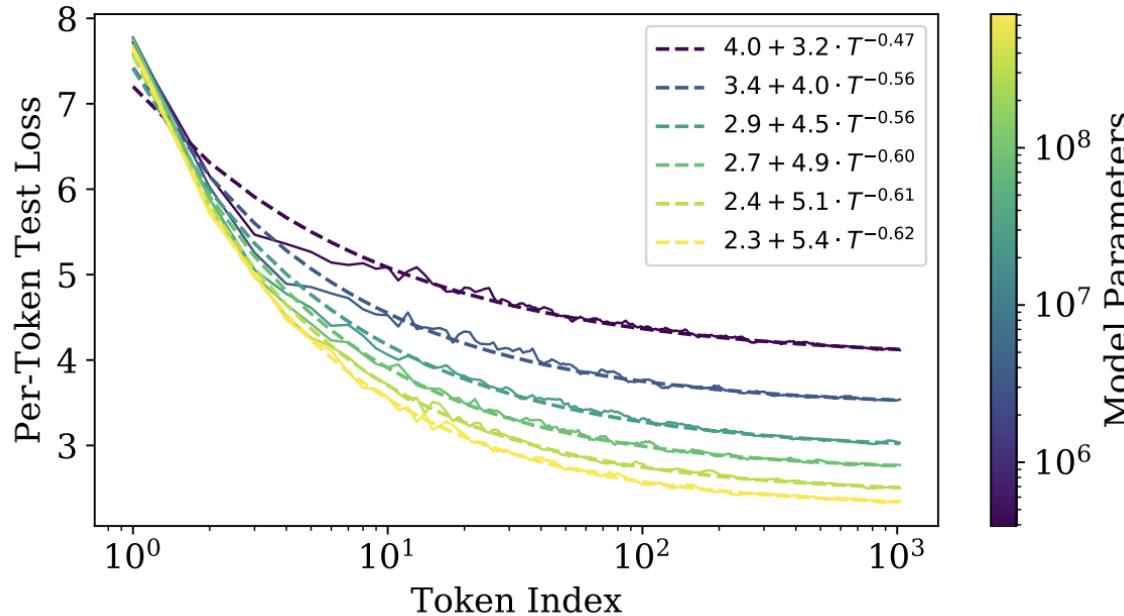


Обучение на том же датасете,
с одинаковой длиной контекста

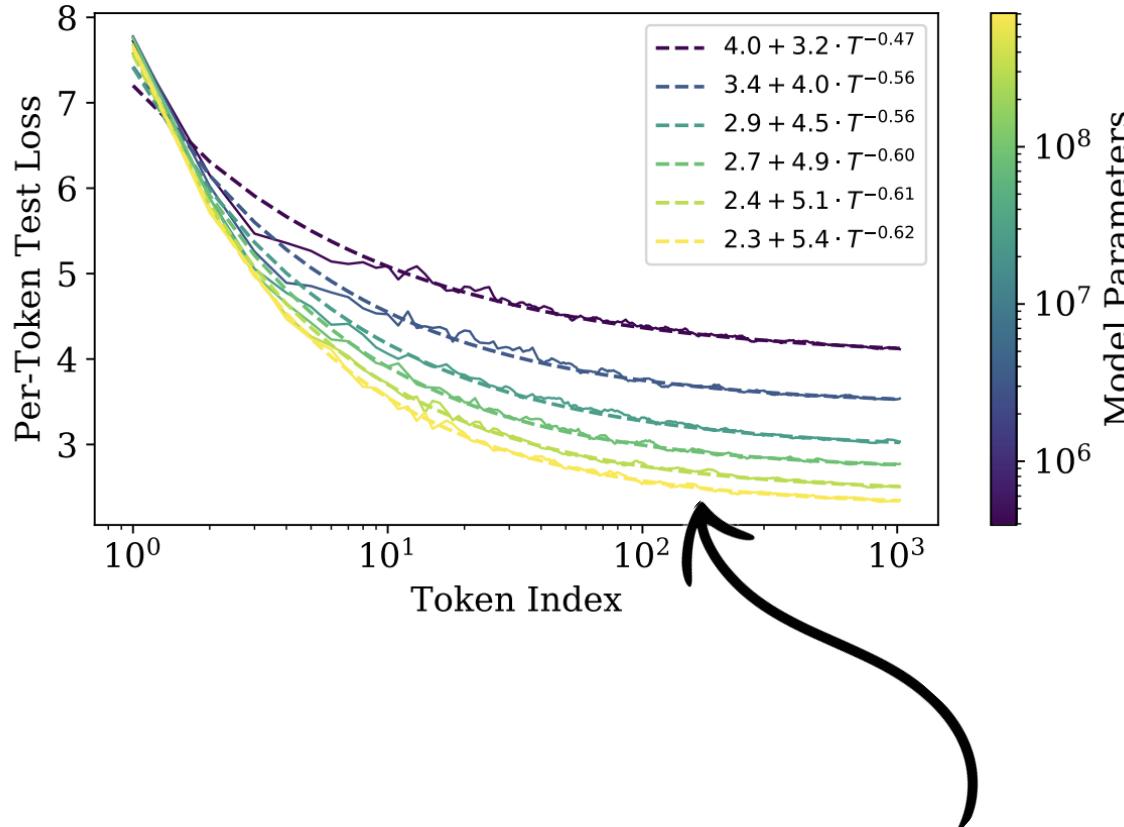


Один из самых важных графиков

Хотим скейлить то, что скейлится лучше

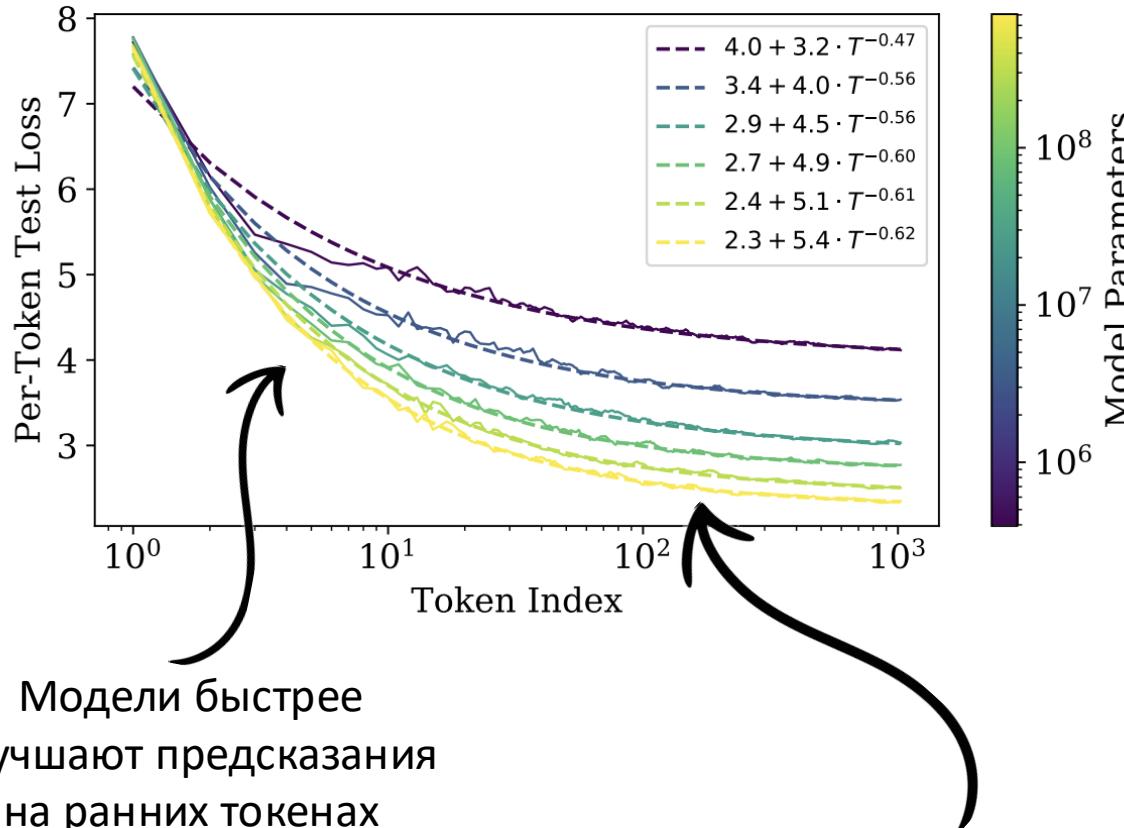


Хотим скейлить то, что скейлится лучше

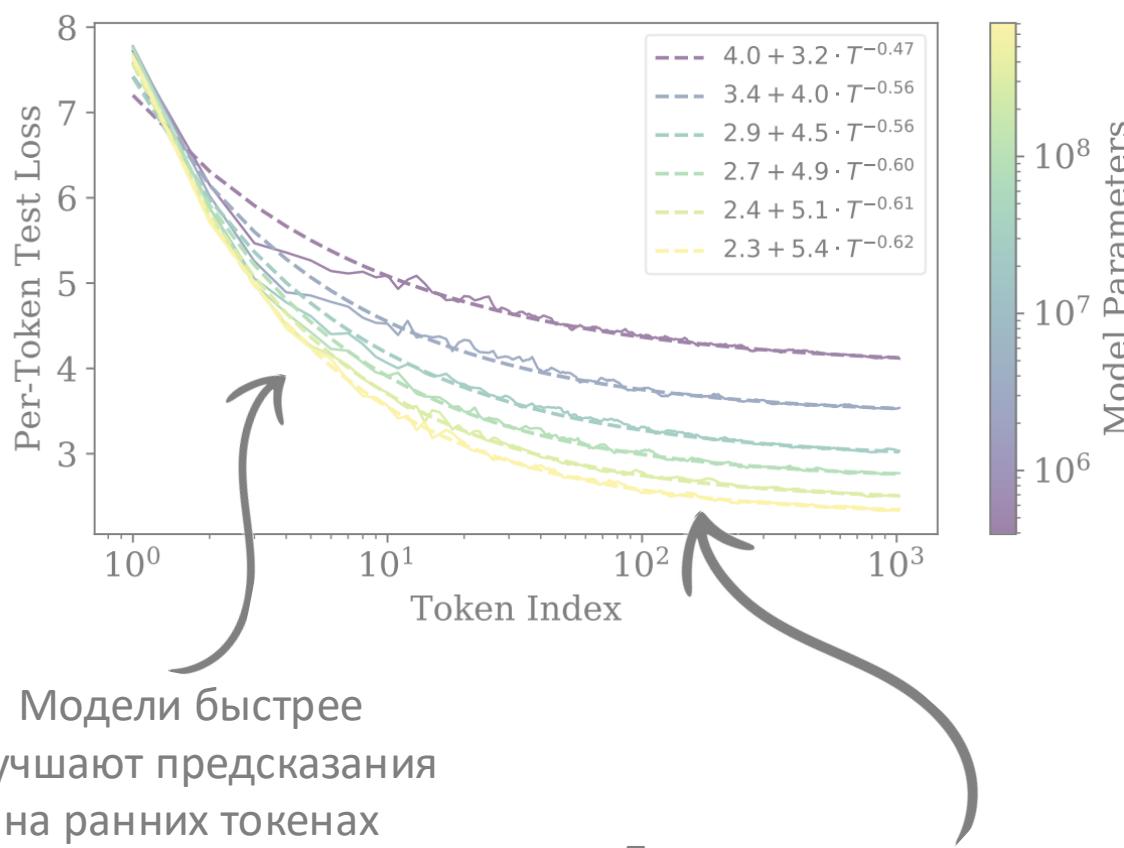


Большие модели ищут
паттерны эффективнее
(и требуют меньше контекста)

Хотим скейлить то, что скейлится лучше

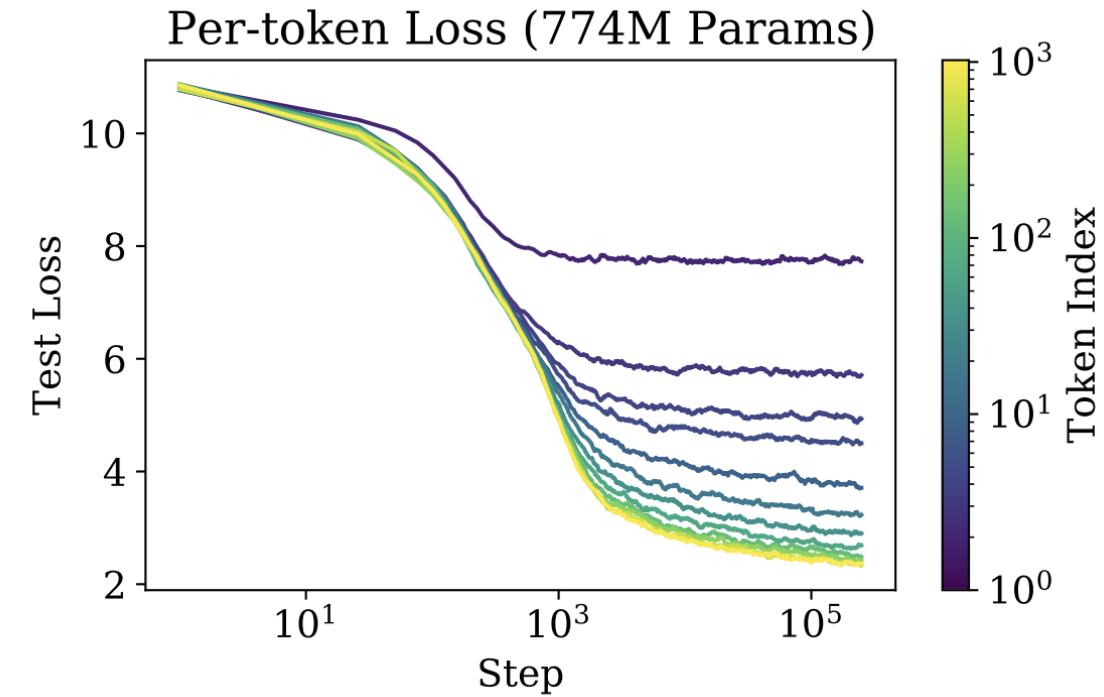


Хотим скейлить то, что скейлится лучше



Модели быстрее
улучшают предсказания
на ранних токенах

Большие модели ищут
паттерны эффективнее
(и требуют меньше контекста)



Модель сначала изучает
локальные, а затем —
глобальные зависимости в
контексте

Что мы ищем?

Как обычно случается:

- У вас есть R видеокарточек на T дней
 - Это позволяет оценить суммарную мощность C (в РF-днях)

Что мы ищем?

Как обычно случается:

- У вас есть R видеокарточек на T дней
 - Это позволяет оценить суммарную мощность C (в РF-днях)
- Как мне обучить лучшую модель за это время?
 - А что такое лучшая модель?

Что мы ищем?

Как обычно случается:

- У вас есть R видеокарточек на T дней
 - Это позволяет оценить суммарную мощность C (в РФ-днях)
- Как мне обучить лучшую модель за это время?
 - А что такое лучшая модель?
- Будем искать модель с наименьшим лоссом L
 - Лосс-функция у LLM: энтропия
 - Как мы выяснили, L зависит от размера датасета D и размера модели N

Что мы ищем?

Как обычно случается:

- У вас есть R видеокарточек на T дней
 - Это позволяет оценить суммарную мощность C (в PF-днях)
- Как мне обучить лучшую модель за это время?
 - А что такое лучшая модель?
- Будем искать модель с наименьшим лоссом L
 - Лосс-функция у LLM: энтропия
 - Как мы выяснили, L зависит от размера датасета D и размера модели N
- Поэтому мы решаем следующую задачу:

$$N_{opt}(C), D_{opt}(C) = \operatorname{argmin}_{N, D \text{ s.t. } \text{FLOPs}(N, D)=C} L(N, D)$$

Что мы ищем?

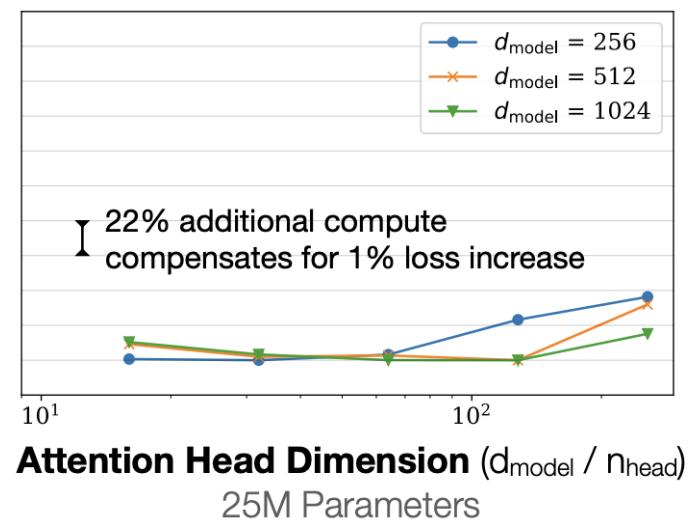
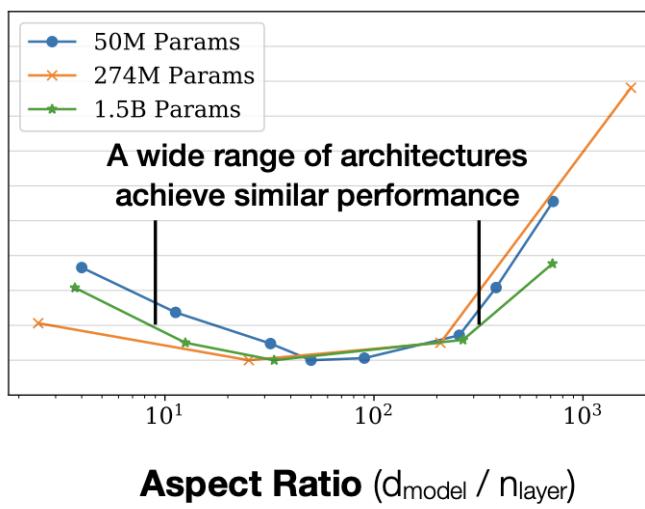
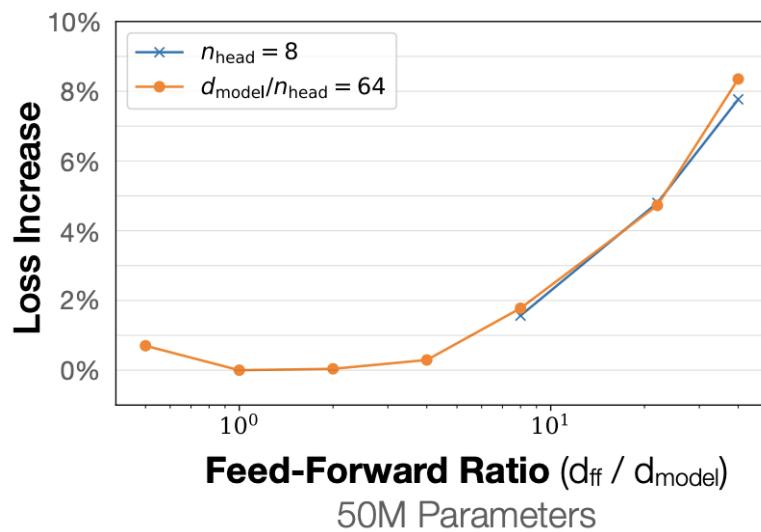
Как обычно случается:

- У вас есть R видеокарточек на T дней
 - Это позволяет оценить суммарную мощность C (в PF-днях)
- Как мне обучить лучшую модель за это время?
 - А что такое лучшая модель?
- Будем искать модель с наименьшим лоссом L
 - Лосс-функция у LLM: энтропия
 - Как мы выяснили, L зависит от размера датасета D и размера модели N
- Поэтому мы решаем следующую задачу:

$$N_{opt}(C), D_{opt}(C) = \operatorname{argmin}_{N, D \text{ s.t. } \text{FLOPs}(N, D)=C} L(N, D) \quad \begin{aligned} \text{FLOPs}(N, D) &\approx 6ND \\ &\approx 6NBS \end{aligned}$$

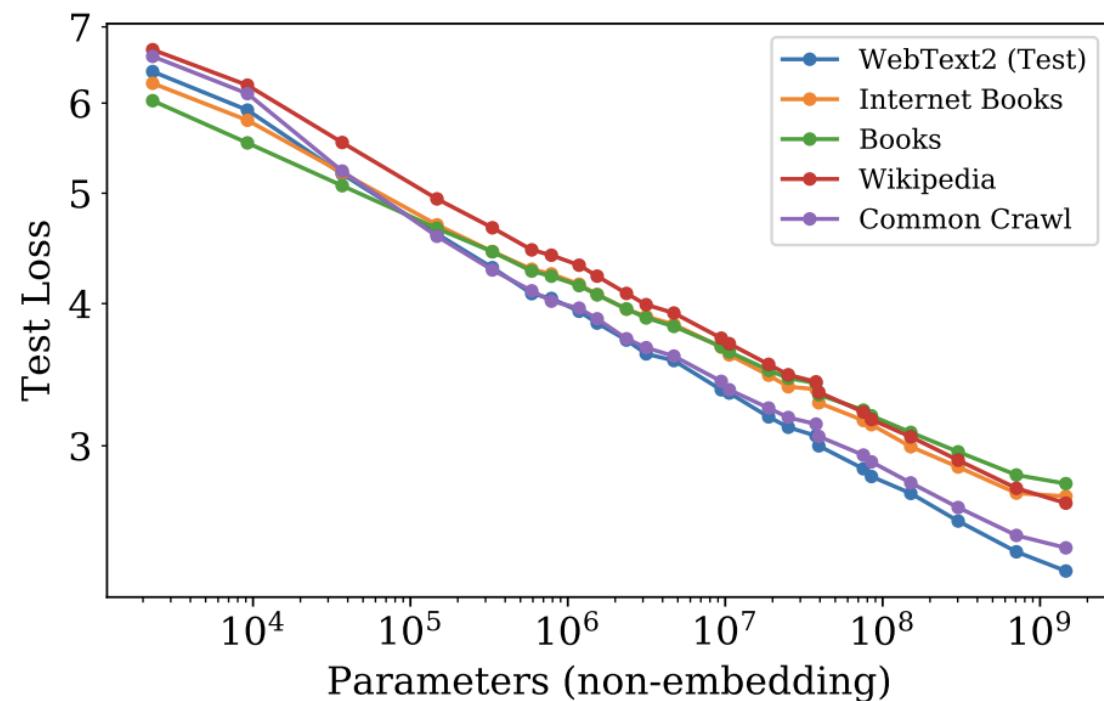
Выводы в работе OpenAI

- Качество модели (=лосс) больше всего **зависит от масштаба (N, C, D)**
- В разумных пределах качество **очень слабо** зависит от других гиперпараметров



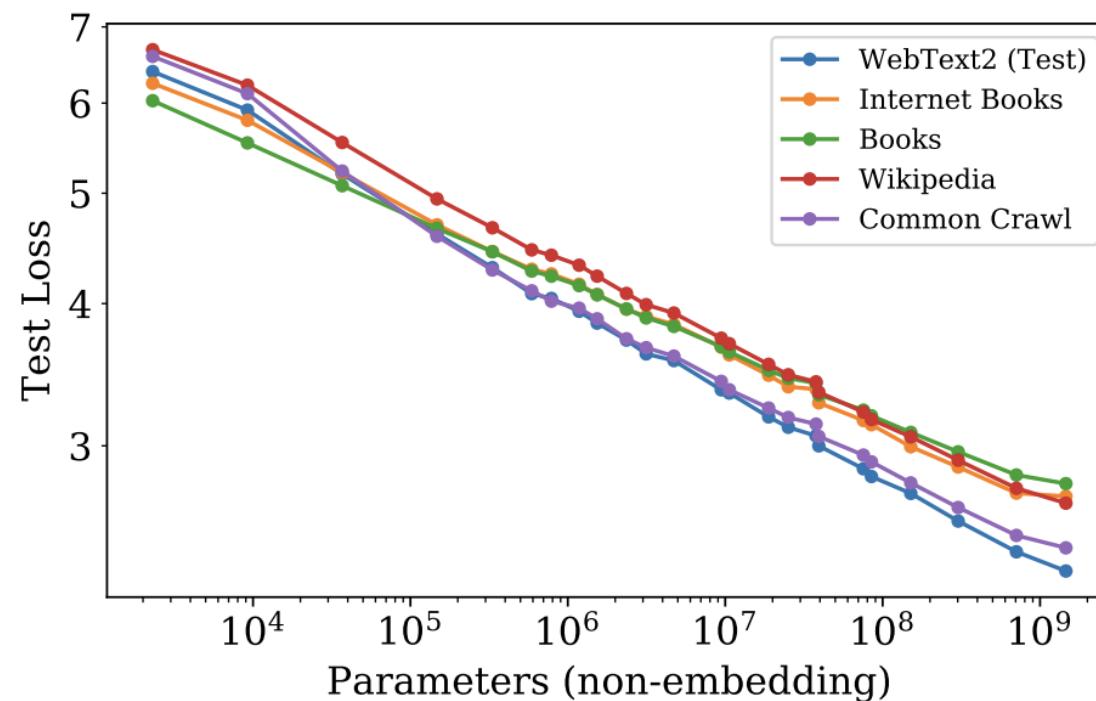
Выводы в работе OpenAI

- Значение лосса следуют предсказуемому поведению
- Можно экстраполировать по малым моделям



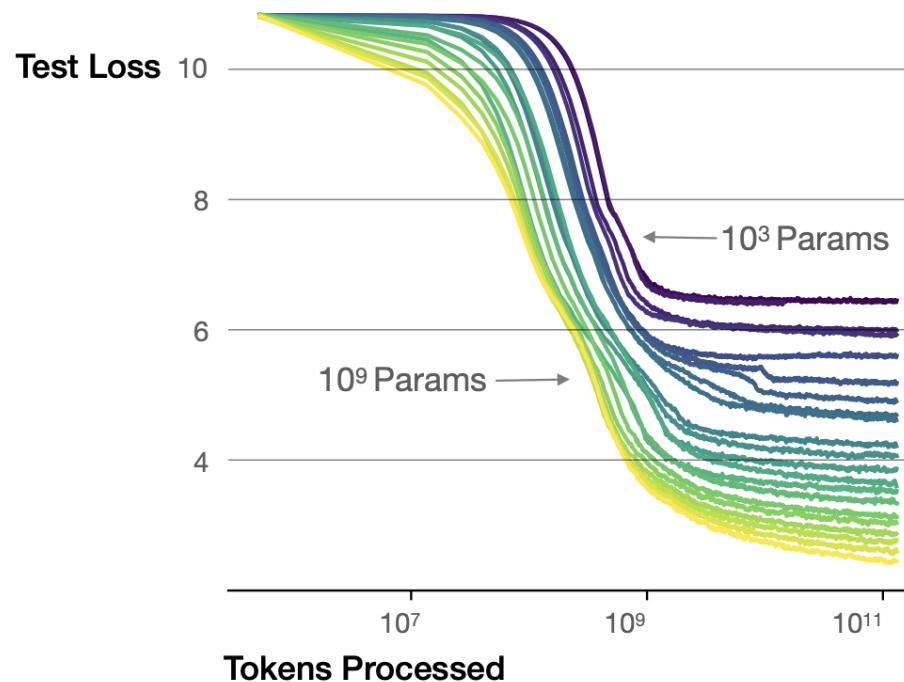
Выводы в работе OpenAI

- Значение лосса следуют предсказуемому поведению
- Можно экстраполировать по малым моделям
- Паттерн универсален для разных датасетов
- Transfer Learning тоже улучшается



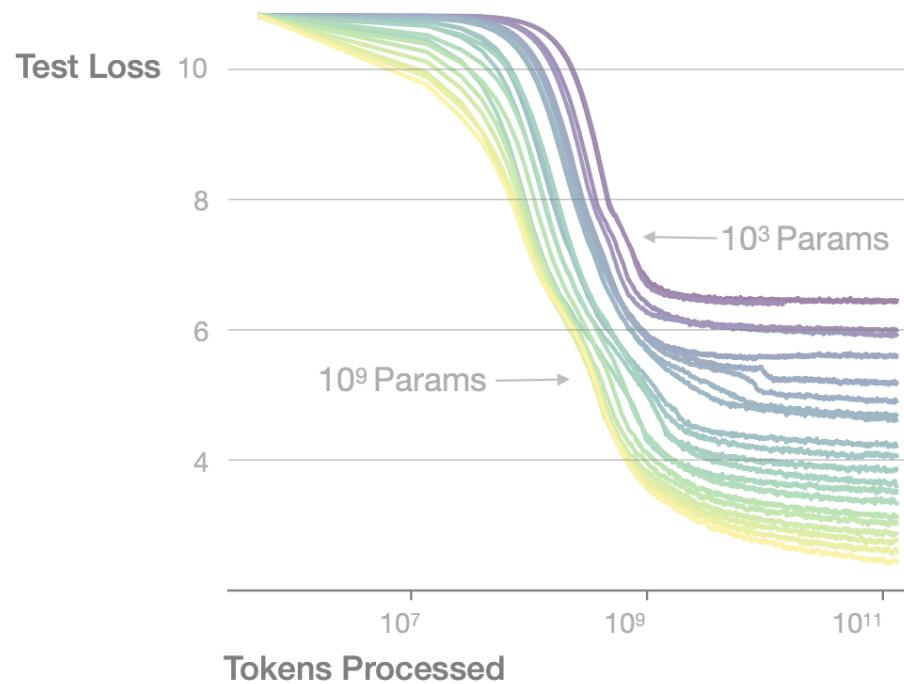
Выводы в работе OpenAI

Большие модели обладают
лучшим Sample Efficiency

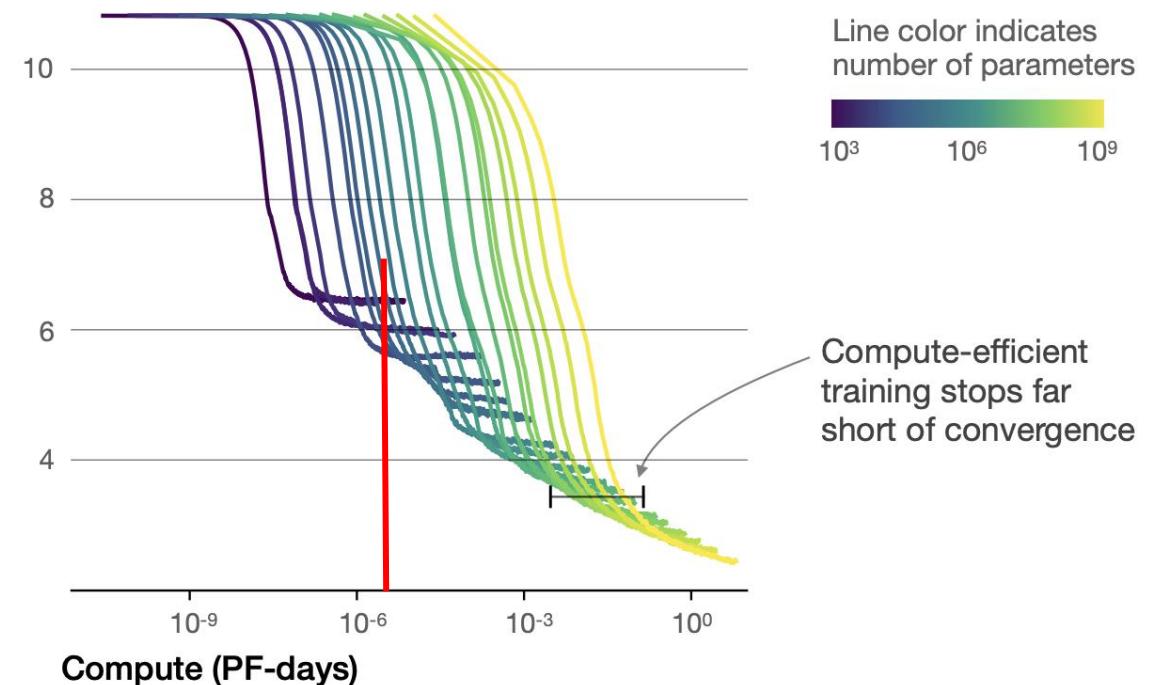


Выводы в работе OpenAI

Большие модели обладают
лучшим Sample Efficiency



Эффективнее **НЕ** учить
модели до сходимости



Выводы в работе OpenAI: самое главное

Каждый раз, когда мы увеличиваем размер модели **в 8 раз**,
нам нужно увеличить датасет примерно **в 5 раз**, чтобы
сохранять оптимальное соотношение.

Выводы в работе OpenAI: самое главное

Kaplan et al. (2020), OpenAI

Каждый раз, когда мы увеличиваем размер модели **в 8 раз**,
нам нужно увеличить датасет примерно **в 5 раз**, чтобы
сохранять оптимальное соотношение.



Jared Kaplan *

Johns Hopkins University, OpenAI

jaredk@jhu.edu

Sam McCandlish*

OpenAI

sam@openai.com

Tom Henighan

OpenAI

henighan@openai.com

Tom B. Brown

OpenAI

tom@openai.com

Benjamin Chess

OpenAI

bchess@openai.com

Rewon Child

OpenAI

rewon@openai.com

Scott Gray

OpenAI

scott@openai.com

Alec Radford

OpenAI

alec@openai.com

Jeffrey Wu

OpenAI

jeffwu@openai.com

Dario Amodei

OpenAI

damodei@openai.com

Выводы в работе OpenAI: самое главное

Kaplan et al. (2020), OpenAI

Каждый раз, когда мы увеличиваем размер модели **в 8 раз**,
нам нужно увеличить датасет примерно **в 5 раз**, чтобы
сохранять оптимальное соотношение.

Model	Size (# Parameters)	Training Tokens
LaMDA (Thoppilan et al., 2022)	137 Billion	168 Billion
GPT-3 (Brown et al., 2020)	175 Billion	300 Billion
Jurassic (Lieber et al., 2021)	178 Billion	300 Billion
<i>Gopher</i> (Rae et al., 2021)	280 Billion	300 Billion
MT-NLG 530B (Smith et al., 2022)	530 Billion	270 Billion

Выводы в работе OpenAI: самое главное

Kaplan et al. (2020), OpenAI

Каждый раз, когда мы увеличиваем размер модели в 8 раз, нам нужно увеличить бюджет времени на обучение в 5 раз, чтобы сохранять оптимальные результаты.



Jared Kaplan *

Johns Hopkins University, OpenAI

jaredk@jhu.edu

Sam McCandlish*

OpenAI

sam@openai.com

Tom B. Brown

OpenAI

tom@openai.com

Benjamin Chess

OpenAI

bchess@openai.com

Rewon Child

OpenAI

rewon@openai.com

Scott Gray

OpenAI

scott@openai.com

Alec Radford

OpenAI

alec@openai.com

Jeffrey Wu

OpenAI

jeffwu@openai.com

Dario Amodei

OpenAI

damodei@openai.com

Выводы в работе OpenAI: самое главное

Kaplan et al. (2020), OpenAI

Каждый раз, когда мы увеличиваем размер модели **в 8 раз**,
нам нужно увеличить датасет примерно **в 5 раз**, чтобы
сохранять оптимальное соотношение.

Hoffmann et al. (2022), DeepMind

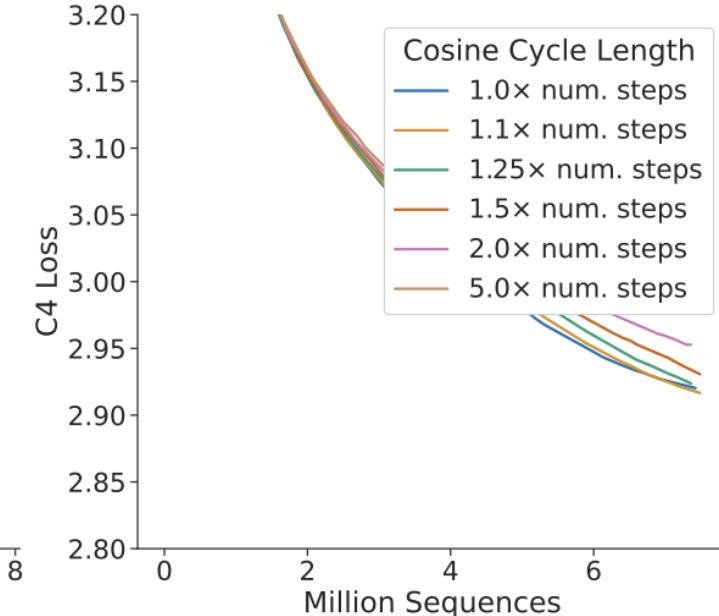
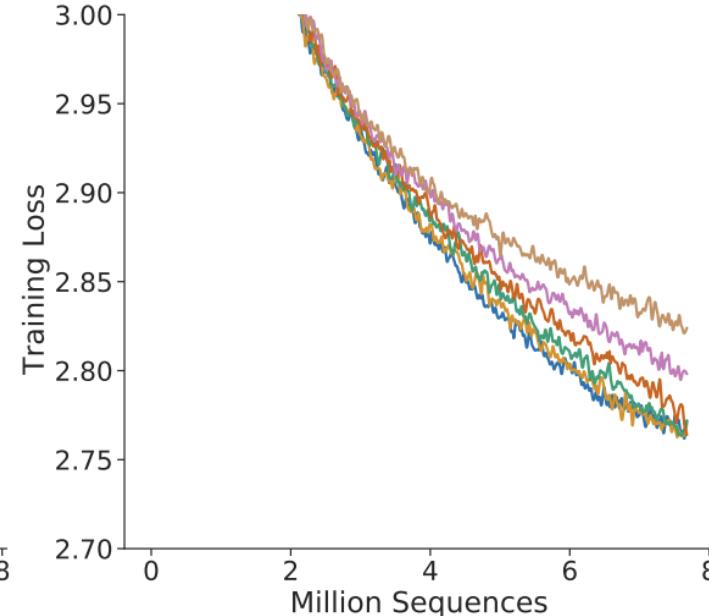
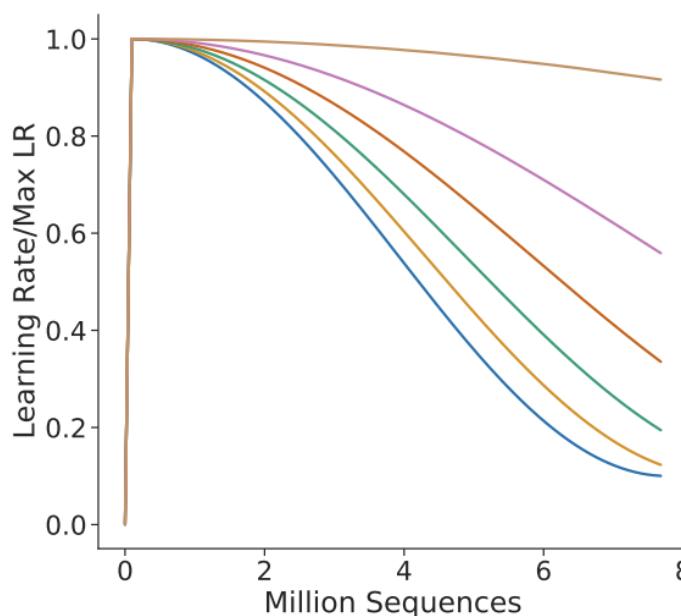
Каждый раз, когда мы увеличиваем размер модели **в 8 раз**,
нам нужно увеличить датасет примерно **в 8 раз**, чтобы
сохранять оптимальное соотношение.

В чём ошибка?

В экспериментах OpenAI для почти
всех моделей были **записаны**
количество токенов и LR Scheduler

В чём ошибка?

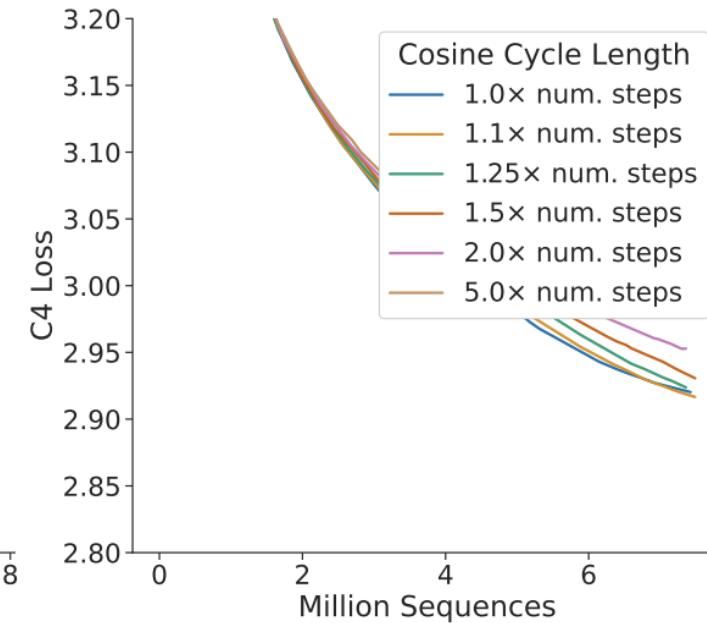
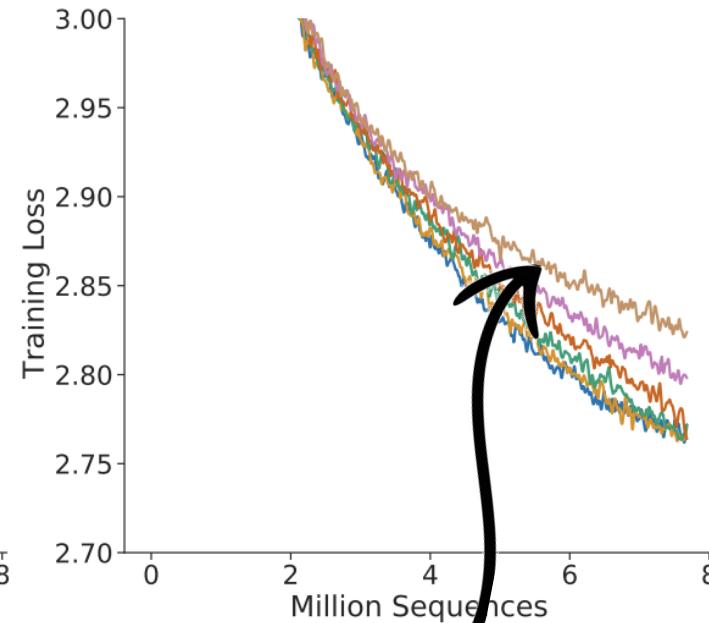
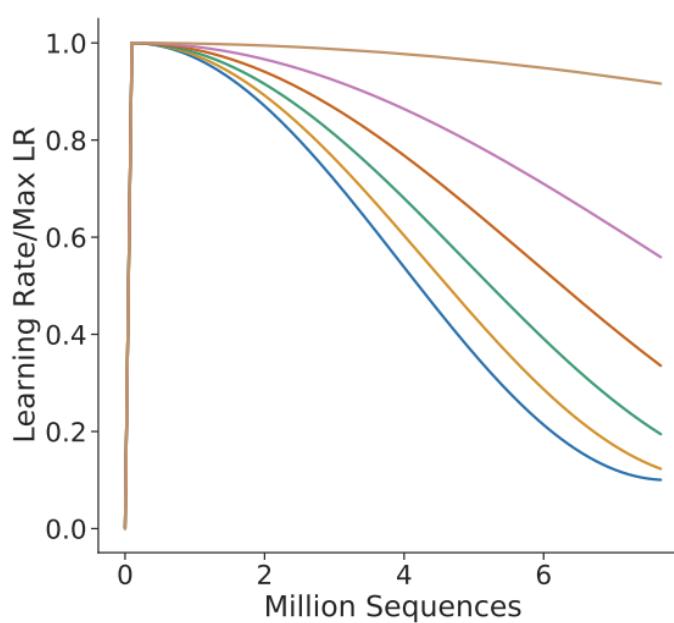
В экспериментах OpenAI для почти всех моделей были **записаны количество токенов и LR Scheduler**



Cosine Cycle Length
— 1.0x num. steps
— 1.1x num. steps
— 1.25x num. steps
— 1.5x num. steps
— 2.0x num. steps
— 5.0x num. steps

В чём ошибка?

В экспериментах OpenAI для почти всех моделей были **записаны количество токенов и LR Scheduler**

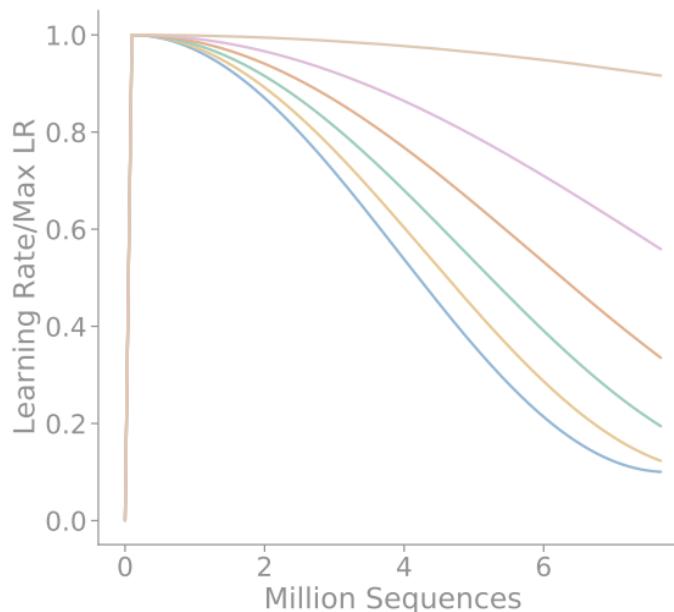


OpenAI экстраполировали из середины...

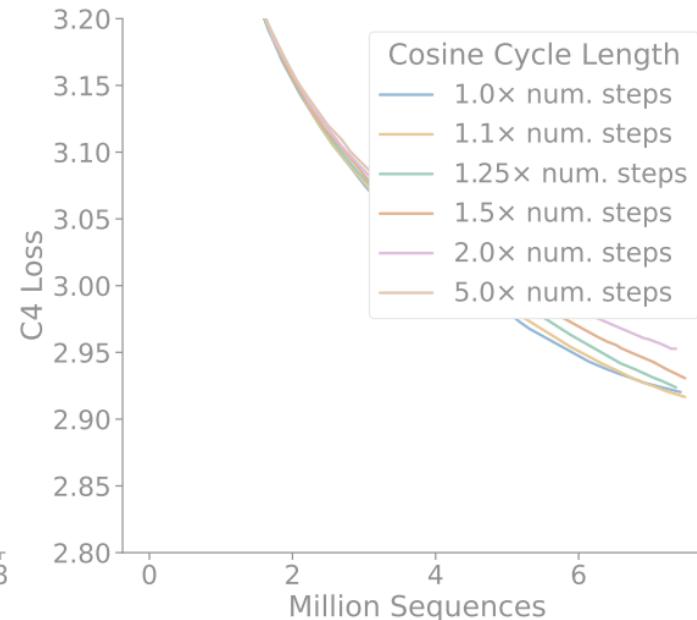
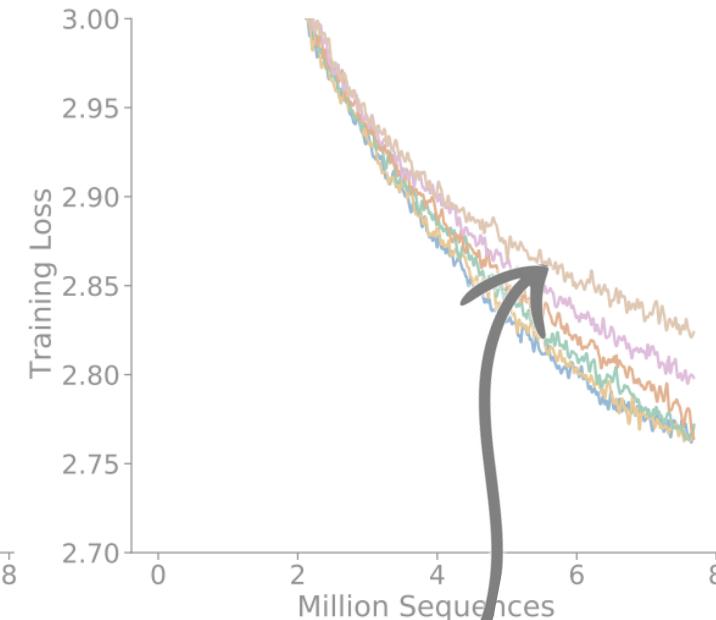
...поэтому получили переоценку

В чём ошибка?

В экспериментах OpenAI для почти всех моделей были **записаны количество токенов и LR Scheduler**



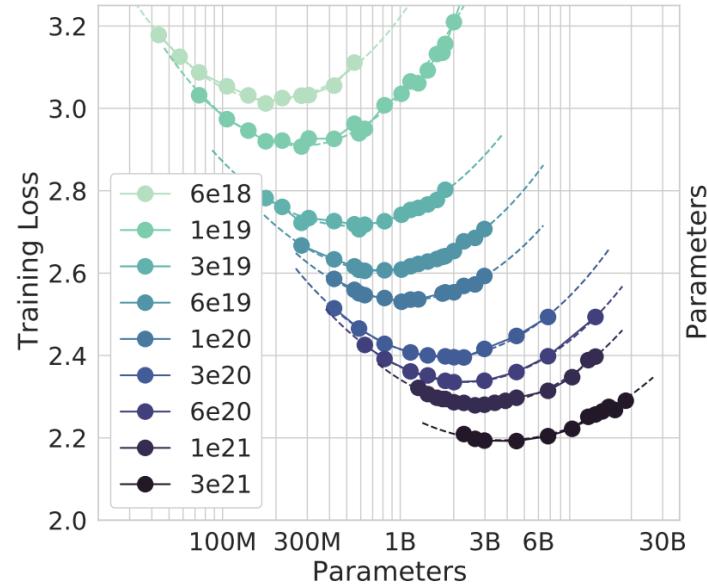
В DeepMind обучали больше моделей, которые сами по себе в среднем были больше (до 16B)



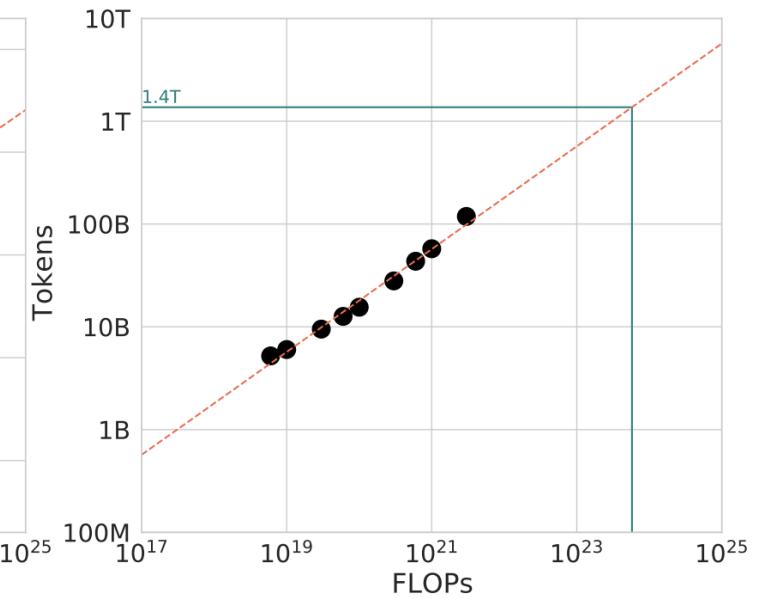
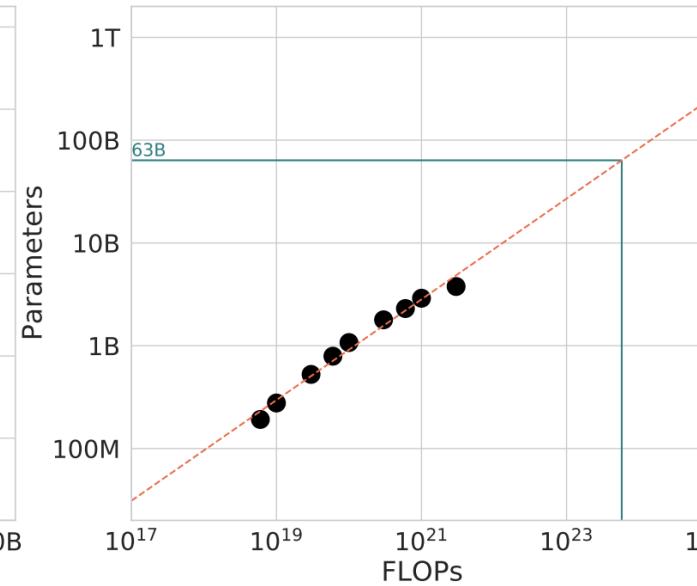
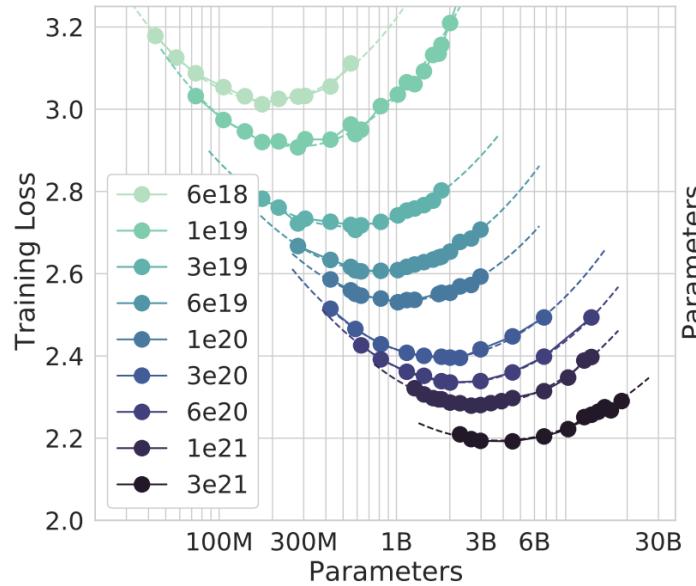
OpenAI экстраполировали из середины...

...поэтому получили переоценку

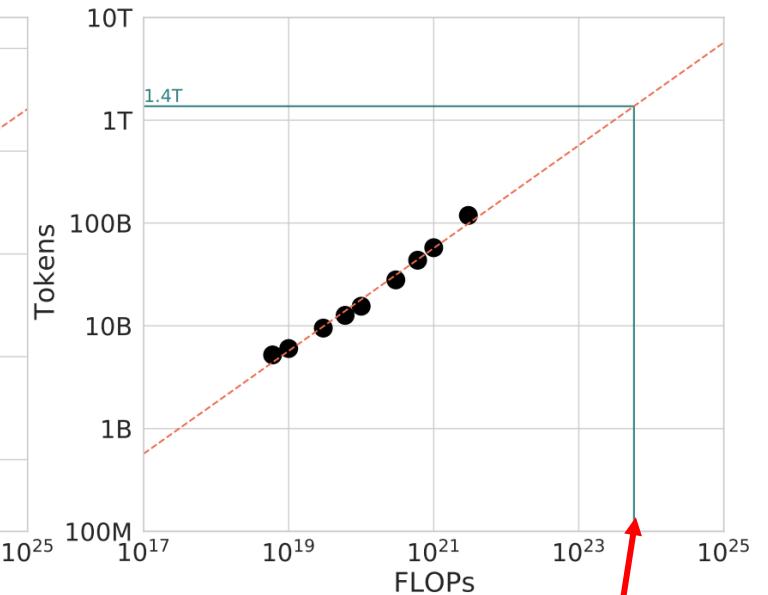
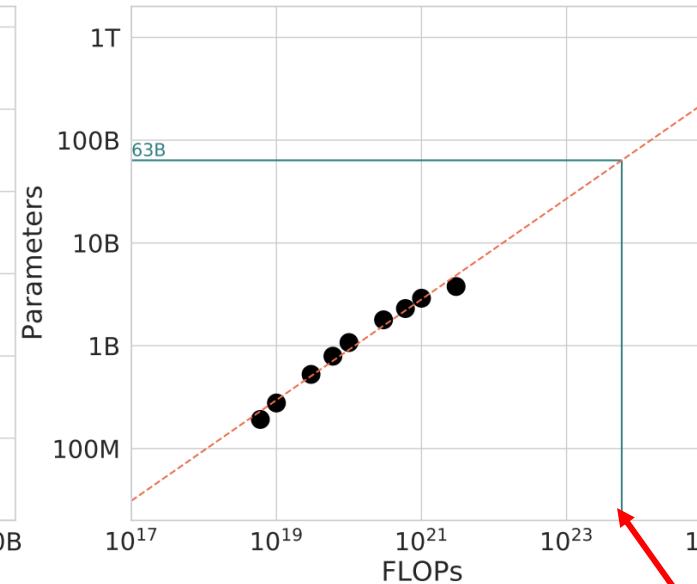
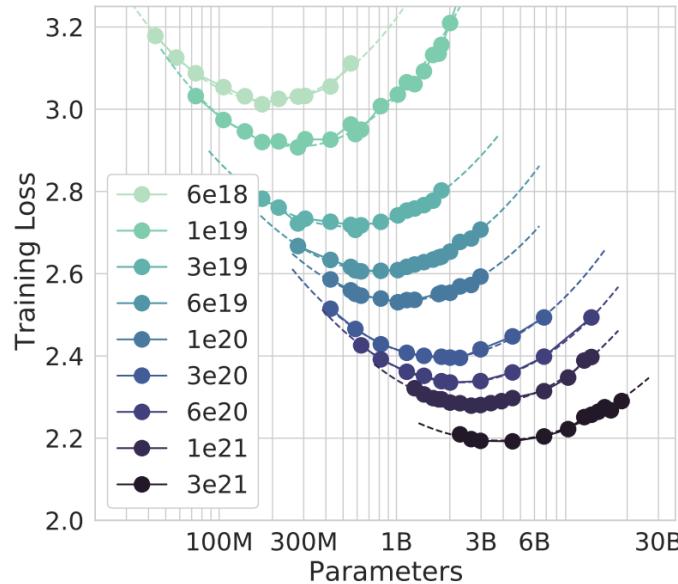
Как измеряли в Chinchilla?



Как измеряли в Chinchilla?



Как измеряли в Chinchilla?

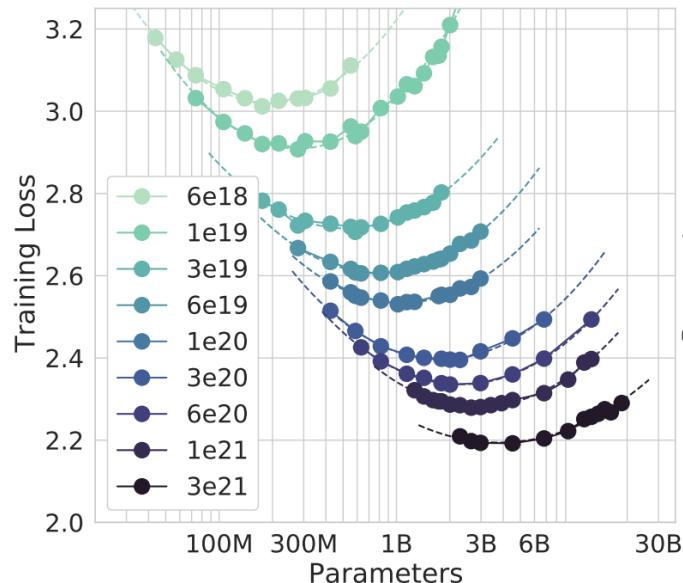


Gopher (Rae et al., 2021) 280 Billion

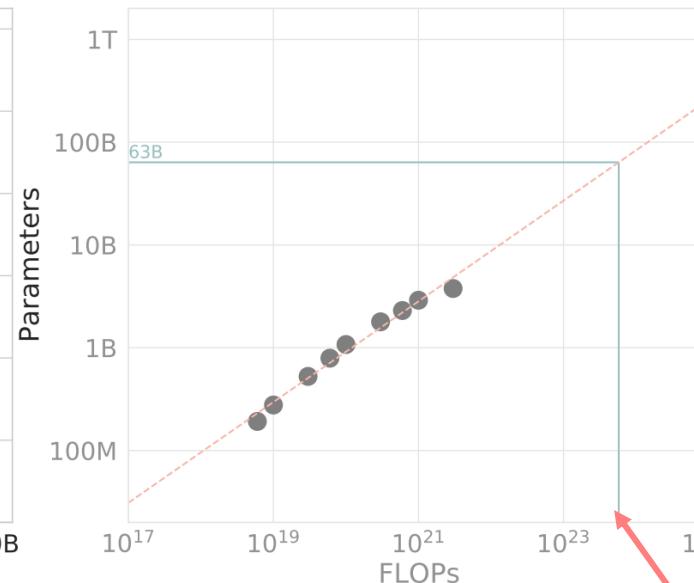
300 Billion

Бюджет как у Gopher

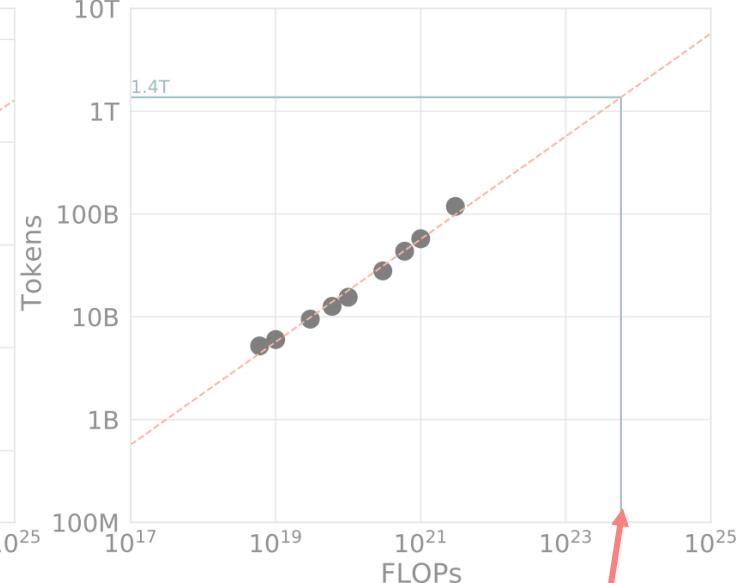
Как измеряли в Chinchilla?



Gopher (Rae et al., 2021) 280 Billion



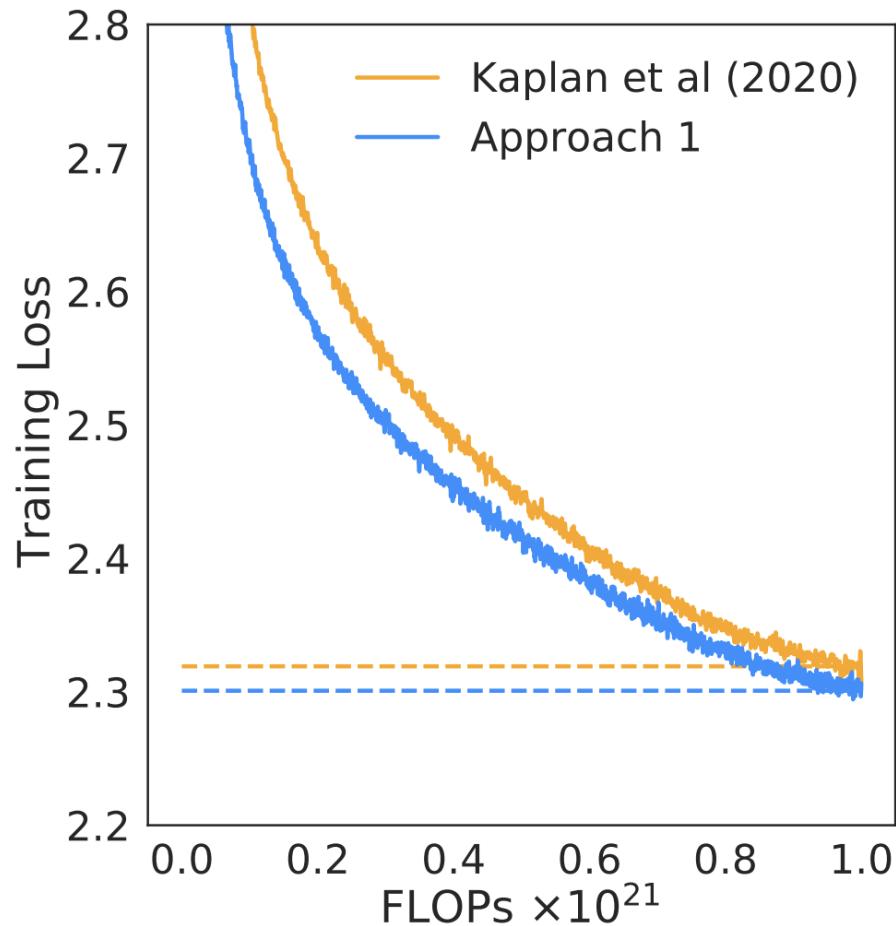
300 Billion



Бюджет как у Gopher

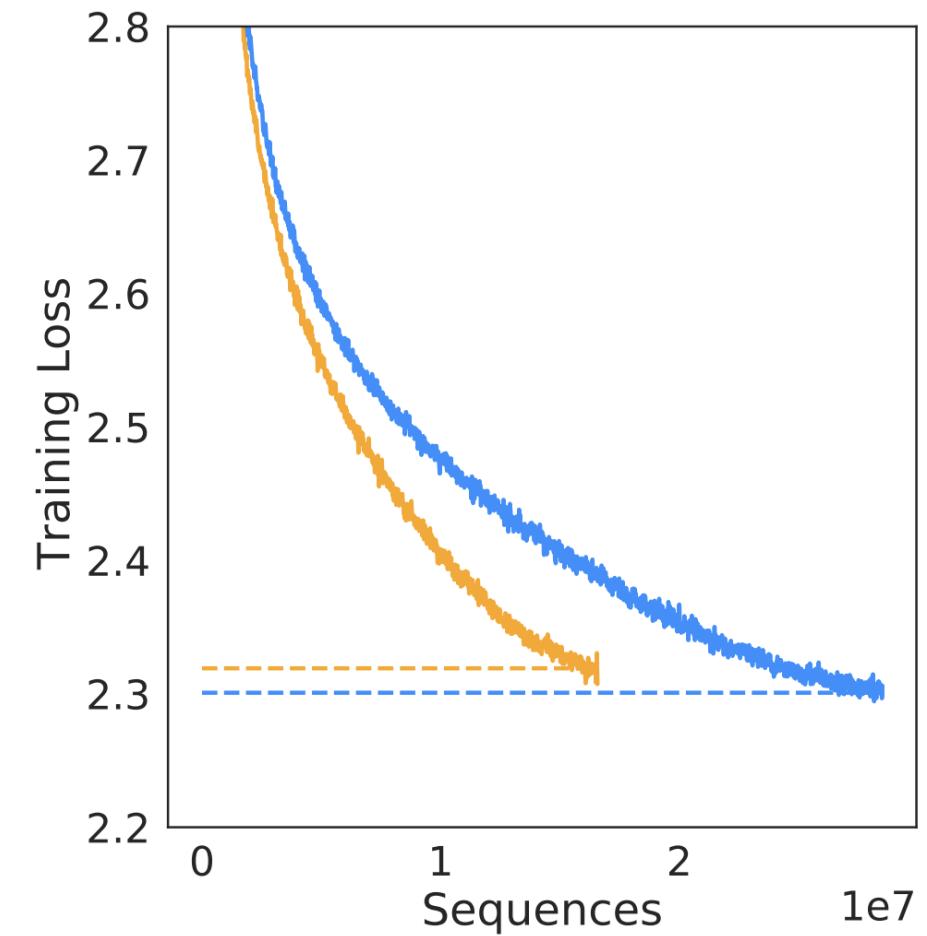
Approach	Coeff. a where $N_{opt} \propto C^a$	Coeff. b where $D_{opt} \propto C^b$
1. Minimum over training curves	0.50 (0.488, 0.502)	0.50 (0.501, 0.512)
2. IsoFLOP profiles	0.49 (0.462, 0.534)	0.51 (0.483, 0.529)
3. Parametric modelling of the loss	0.46 (0.454, 0.455)	0.54 (0.542, 0.543)
Kaplan et al. (2020)	0.73	0.27

Proof of concept (маленький масштаб)



Везде меньший лосс
на тех же мощностях

4.74В модель
VS
2.8В модель



...но проходим
больше токенов

Так и сколько нужно?

Parameters	FLOPs	FLOPs (in <i>Gopher</i> unit)	Tokens
400 Million	1.92e+19	1/29,968	8.0 Billion
1 Billion	1.21e+20	1/4,761	20.2 Billion
10 Billion	1.23e+22	1/46	205.1 Billion
<u>67 Billion</u>	5.76e+23	1	<u>1.5 Trillion</u>
<u>175 Billion</u>	3.85e+24	6.7	<u>3.7 Trillion</u>
280 Billion	9.90e+24	17.2	5.9 Trillion
520 Billion	3.43e+25	59.5	11.0 Trillion
<u>1 Trillion</u>	1.27e+26	221.3	<u>21.2 Trillion</u>
10 Trillion	1.30e+28	22515.9	216.2 Trillion

Так и сколько нужно?

Parameters	FLOPs	FLOPs (in Gopher unit)	Tokens	Total train compute (flops)
Model				
400 Million	1.92e+19	1/29,968	8.0 Billion	
1 Billion	1.21e+20	1/4,761	20.2 Billion	
10 Billion	1.23e+22	1/46	205.1 Billion	
<u>67 Billion</u>	<u>5.76e+23</u>	1	<u>1.5 Trillion</u>	
<u>175 Billion</u>	<u>3.85e+24</u>	6.7	<u>3.7 Trillion</u>	
280 Billion	9.90e+24	17.2	5.9 Trillion	GPT-3 13B 2.31E+22
520 Billion	3.43e+25	59.5	11.0 Trillion	GPT-3 175B 3.14E+23
<u>1 Trillion</u>	<u>1.27e+26</u>	221.3	<u>21.2 Trillion</u>	
10 Trillion	1.30e+28	22515.9	216.2 Trillion	

Proof of concept (большой масштаб)

Model	Layers	Number Heads	Key/Value Size	d_{model}	Max LR	Batch Size
<i>Gopher</i> 280B	80	128	128	16,384	4×10^{-5}	3M → 6M
<i>Chinchilla</i> 70B	80	64	128	8,192	1×10^{-4}	1.5M → 3M

Random	25.0%
Average human rater	34.5%
GPT-3 5-shot	43.9%
<i>Gopher</i> 5-shot	60.0%
<i>Chinchilla</i> 5-shot	67.6%
Average human expert performance	89.8%

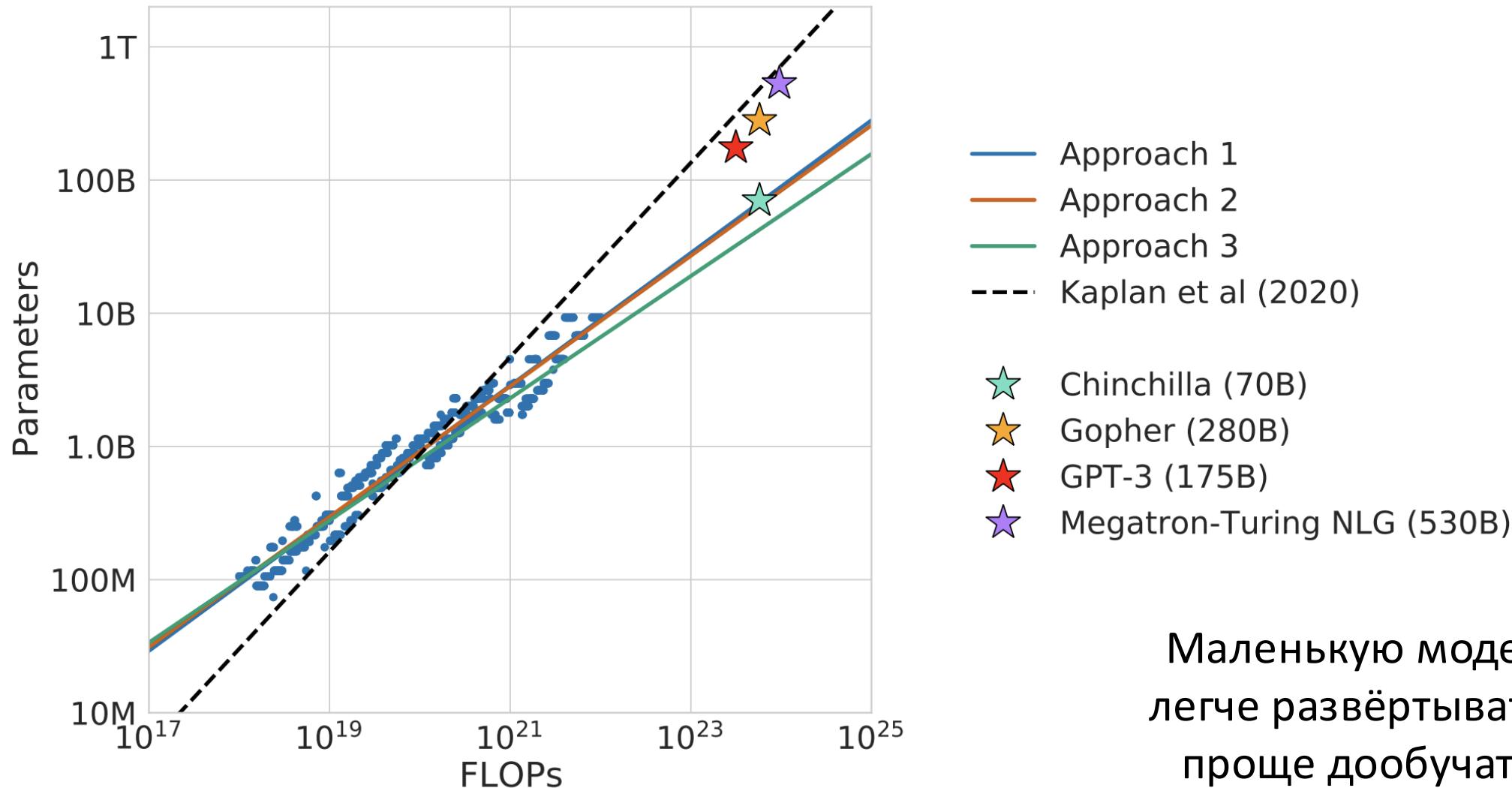
MMLU

	<i>Chinchilla</i>	<i>Gopher</i>	GPT-3	MT-NLG 530B
HellaSWAG	80.8%	79.2%	78.9%	80.2%
PIQA	81.8%	81.8%	81.0%	82.0%
Winogrande	74.9%	70.1%	70.2%	73.0%
SIQA	51.3%	50.6%	-	-
BoolQ	83.7%	79.3%	60.5%	78.2%



	Method	<i>Chinchilla</i>	<i>Gopher</i>	GPT-3
Natural Questions (dev)	0-shot	16.6%	10.1%	14.6%
	5-shot	31.5%	24.5%	-
	64-shot	35.5%	28.2%	29.9%
TriviaQA (unfiltered, test)	0-shot	67.0%	52.8%	64.3 %
	5-shot	73.2%	63.6%	-
	64-shot	72.3%	61.3%	71.2%
TriviaQA (filtered, dev)	0-shot	55.4%	43.5%	-
	5-shot	64.1%	57.0%	-
	64-shot	64.6%	57.2%	-

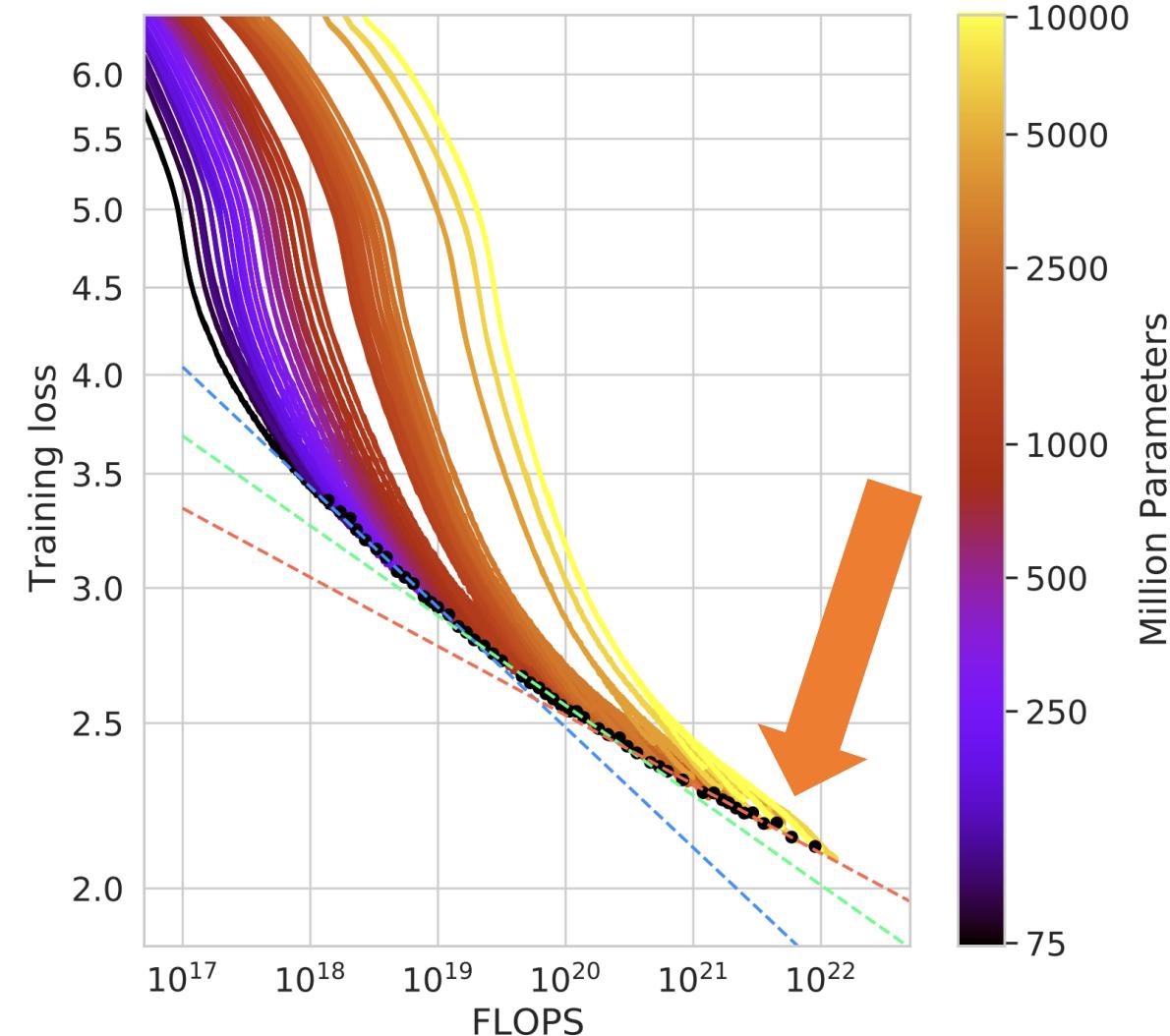
Сравнение скейла моделей



Чего боятся в OpenAI..?

“
there is a curvature in the
FLOP-minimal loss frontier
”

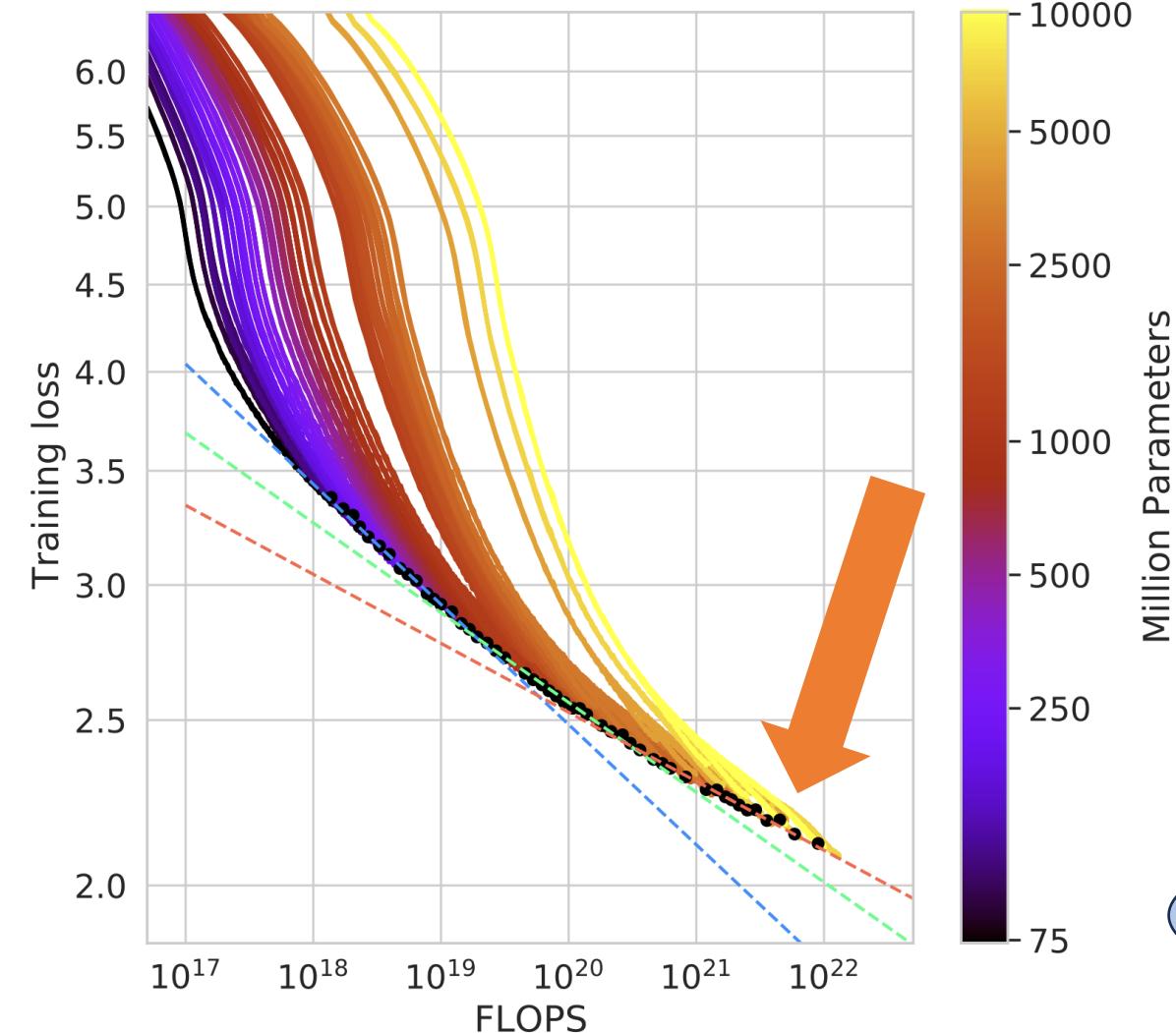
“
projections from very
small models lead to
different predictions than
those from larger model
”



Чего боятся в OpenAI..?

“
there is a curvature in the
FLOP-minimal loss frontier
”

“
projections from very
small models lead to
different predictions than
those from larger model
”



Я ещё скажу
своё слово...



ИТОГИ

- Масштабировать модели нужно правильно
 - В основном решает размер модели и датасета

Итоги

- Масштабировать модели нужно правильно
 - В основном решает размер модели и датасета
- Трансформеры более эффективно работают с контекстом

Итоги

- Масштабировать модели нужно правильно
 - В основном решает размер модели и датасета
- Трансформеры более эффективно работают с контекстом
- Пытаемся искать наилучший loss при заданном C

Итоги

- Масштабировать модели нужно правильно
 - В основном решает размер модели и датасета
- Трансформеры более эффективно работают с контекстом
- Пытаемся искать наилучший loss при заданном C
- В работе OpenAI ошиблись с LR Scheduler
 - Как следствие — сделали неправильную экстраполяцию

Итоги

- Масштабировать модели нужно правильно
 - В основном решает размер модели и датасета
- Трансформеры более эффективно работают с контекстом
- Пытаемся искать наилучший loss при заданном C
- В работе OpenAI ошиблись с LR Scheduler
 - Как следствие — сделали неправильную экстраполяцию
- Оптимальное* масштабирование 1:1

Итоги

- Масштабировать модели нужно правильно
 - В основном решает размер модели и датасета
- Трансформеры более эффективно работают с контекстом
- Пытаемся искать наилучший loss при заданном C
- В работе OpenAI ошиблись с LR Scheduler
 - Как следствие — сделали неправильную экстраполяцию
- Оптимальное* масштабирование 1:1
- На практике результат соответствует ожиданиям
 - меньшая модель реально лучше!

Итоги

- Масштабировать модели нужно правильно
 - В основном решает размер модели и датасета
- Трансформеры более эффективно работают с контекстом
- Пытаемся искать наилучший loss при заданном C
- В работе OpenAI ошиблись с LR Scheduler
 - Как следствие — сделали неправильную экстраполяцию
- Оптимальное* масштабирование 1:1
- На практике результат соответствует ожиданиям
 - меньшая модель реально лучше!
- Возможно, для очень больших моделей закон немного отличается

Directed by
IGOR D. KOTENKOV