курс «Глубокое обучение»

Языковые модели

Александр Дьяконов

30 марта 2020 года

План

Моделирование языка (Language Modeling)

Параметрическое оценивание

Немарковские модели

RNN-моделирование языка

Подходы к генерированию

Beam Search (метод луча)

ERNIE (Enhanced Representation through kNowledge IntEgration)

GPT

GPT2

Нейронная дегенерация текстов

Моделирование языка (Language Modeling)

Вероятность текста

$$p(x_1,\ldots,x_n)$$

Предсказание следующего слова

$$p(x_n \mid x_1,\ldots,x_{n-1})$$

свойство Маркова

$$p(x_n \mid x_{n-k},\ldots,x_{n-1})$$

```
в лесу родилась  ёлочка 0.4
                 белочка 0.2
                 лисичка 0.1
                 берёзка 0.05
                 баба 0.02
                 ...
```

Языковые модели в жизни (Language Models)

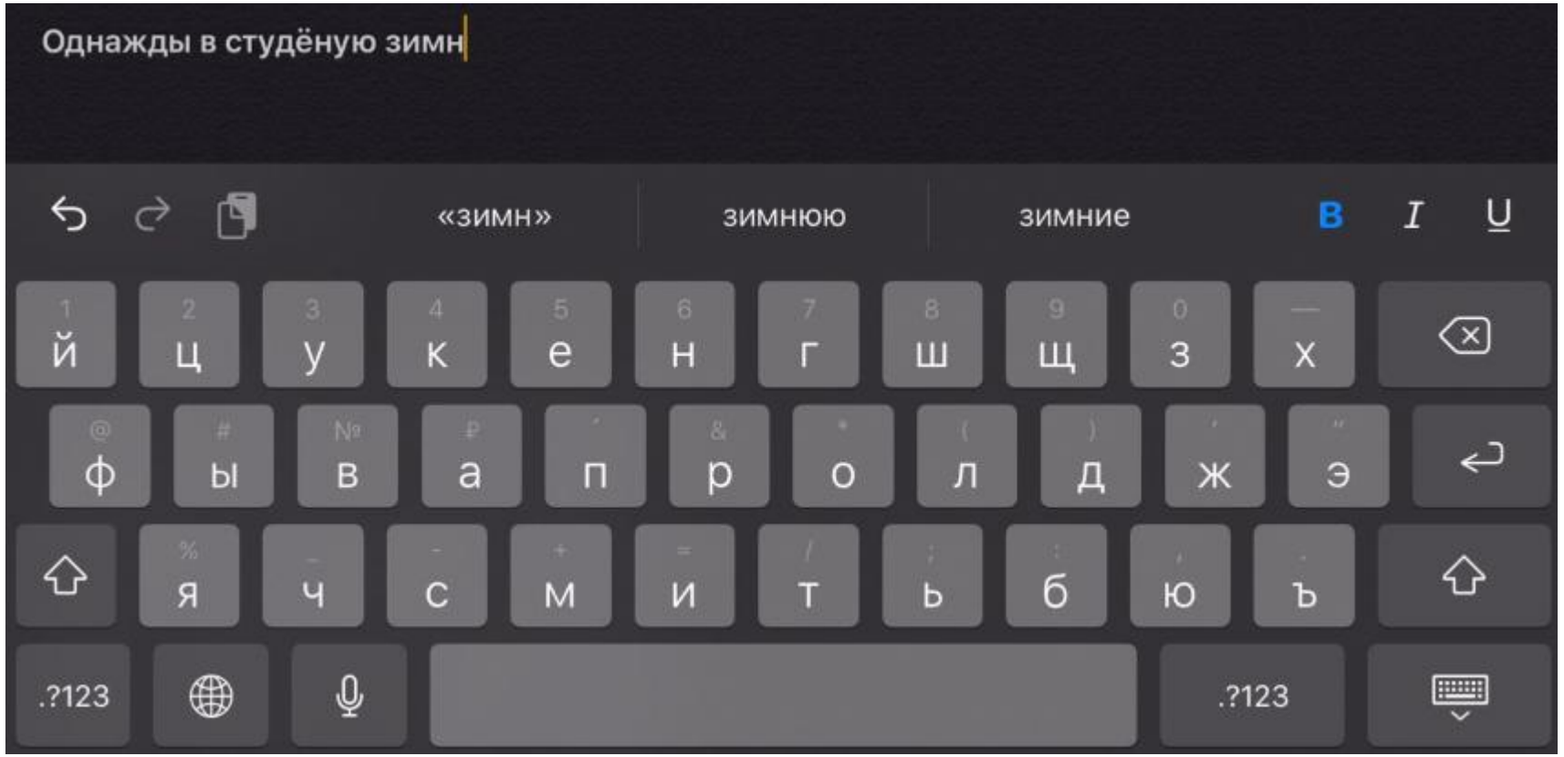
Google

анализ малых

- анализ малых **данных**
- анализ малых **литературных форм**
- анализ малых **данных гос auc**
- анализ малых **выборок**
- анализ малых **предприятий**
- анализ **бесконечно** малых
- **дьяконов** анализ малых **данных**
- **структурный** анализ малых **групп**
- **статистический** анализ малых **выборок**
- анализ **бесконечно** малых **на английском**

Поиск в Google | Мне повезёт!

Пожаловаться на неприемлемые подсказки



Моделирование языка: n-gram Language Models

учимся генерировать текст – как было до DL... n-gram Language Models

Насколько вероятно предложение
«кот поймал в мешок дровосека»

Unigram Modelling

$$p(\text{кот}) \cdot p(\text{поймал}) \cdot p(\text{в}) \cdot p(\text{мешок}) \cdot p(\text{дровосека})$$

Bigram Modelling

$$p(\text{кот}) \cdot p(\text{поймал} \mid \text{кот}) \cdot p(\text{в} \mid \text{поймал}) \cdot p(\text{мешок} \mid \text{в}) \cdot p(\text{дровосека} \mid \text{мешок})$$

Trigram Modelling

$$p(\text{кот}) \cdot p(\text{поймал} \mid \text{кот}) \cdot p(\text{в} \mid \text{кот}, \text{поймал}) \cdot p(\text{мешок} \mid \text{поймал}, \text{в}) \ldots$$

~~в лесу родилась~~ ёлочка, в лесу она MASK

Проблема

в корпусе может не быть некоторых сочетаний

Сглаживание

$$p(x_t \mid x_{t-n},\ldots,x_{t-1}) = \frac{\#(x_{t-n},\ldots,x_{t-1},x_t)+\alpha}{\#(x_{t-n},\ldots,x_{t-1})+\alpha\,|V|}$$

Backoff (примерно так...)

при $\#(x_{t-n},\ldots,x_{t-1}) = 0$

$$p(x_t \mid x_{t-n},\ldots,x_{t-1}) = \alpha(x_{t-n},\ldots,x_{t-1})\frac{\#(x_{t-n+1},\ldots,x_{t-1},x_t)}{\#(x_{t-n+1},\ldots,x_{t-1})}$$

умножаем на некоторый «понижающий множитель»

или через частоты меньших порядков (лк с ними)

Марковская парадигма

Проблема

Маленькое обобщение (Lack of Generalization)

если в выборке только

(идти, в, сад), (идти, в, огород)

тогда проблемы при

p(идти, в, парк) = ?

Выход: моделирование языка с помощью НС

Параметрическое оценивание: нейросетевой подход

$$p(x_t \mid x_{t-n},\ldots,x_{t-1})$$

пусть зависимость от n предыдущих

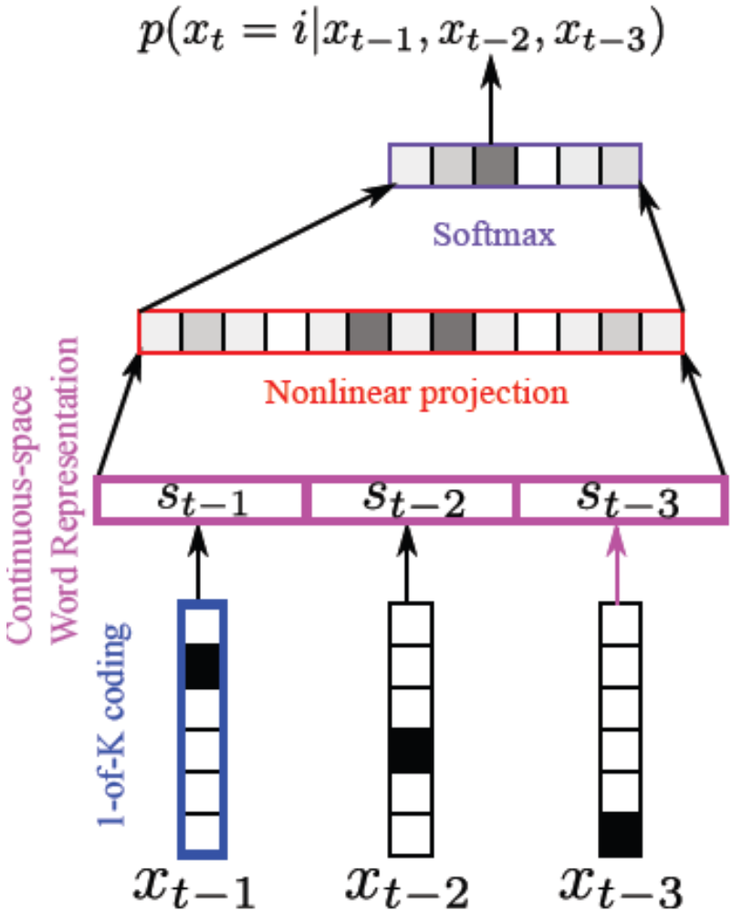
ONE для слов

$$s_j = W_{d\times|V|}x_j$$

$$h = \tanh(U_{d'\times nd}[s_{t-1},\ldots,s_{t-n}]+b)$$

$$y = V_{|V|\times d'}h + c$$

$$p(x_t = i \mid x_{t-n},\ldots,x_{t-1}) = \text{softmax}(y)$$



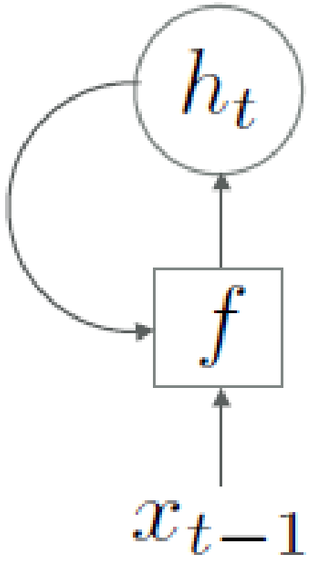
Немарковские модели: RNN-подход

$$p(x_t,\ldots,x_T)=\prod_{t=1}^{T} p(x_t \mid x_1,\ldots,x_{t-1})$$

т.е. зависимость от всех слов предложения!

Как подавать на вход НС информацию разной длины?

Рекурсия



$$h_0 = 0$$
$$h_t = f(x_{t-1}, h_{t-1})$$ (внутренне состояние = память)
$$p(x_t \mid x_1,\ldots,x_{t-1}) = g(h_t)$$

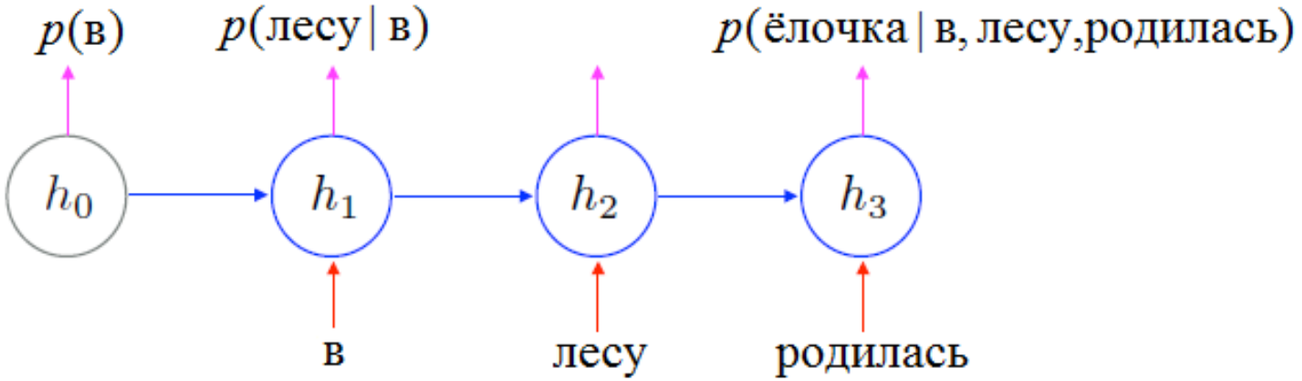
f **– transition function**

g **– output (readout) function**

RNN-моделирование языка

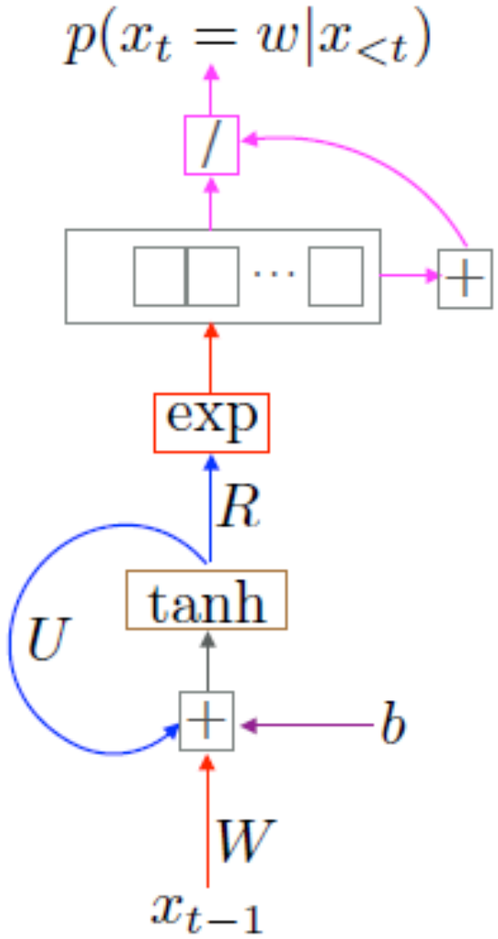
p(в, лесу, родилась, ёлочка)

$$h_0 = 0 \quad \Rightarrow \quad p(\text{в}) = g(h_0)$$
$$h_1 = f(h_0, \text{в}) \quad \Rightarrow \quad p(\text{лесу} \mid \text{в}) = g(h_1)$$
$$h_2 = f(h_1, \text{лесу}) \quad \Rightarrow \quad p(\text{родилась} \mid \text{в}, \text{лесу}) = g(h_2)$$
$$h_3 = f(h_2, \text{родилась}) \quad \Rightarrow \quad p(\text{ёлочка} \mid \text{в}, \text{лесу}, \text{родилась}) = g(h_3)$$
$$p(\text{в}, \text{лесу}, \text{родилась}, \text{ёлочка}) = g(h_0)g(h_1)g(h_2)g(h_3)$$



рекуррентная сеть – можно обрабатывать последовательности любой длины!

RNN-моделирование языка



Transition

$$h_t = \tanh(W_{d\times|V|}x_{t-1} + U_{d\times d}h_{t-1} + b)$$

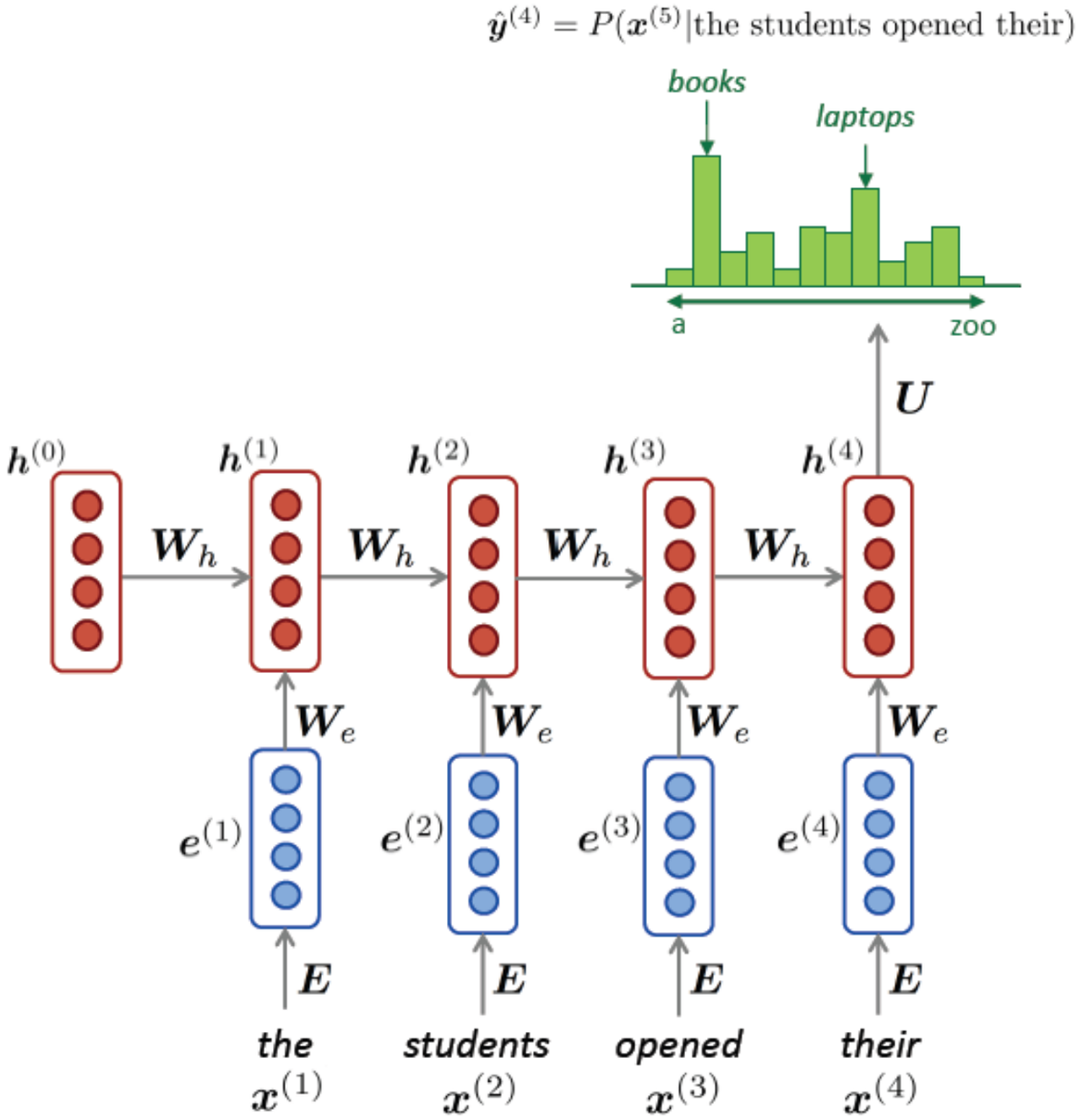
Readout

$$(p(x_t = w \mid x_{<t}))_{w=1}^{|V|} = g(h_t) = \mathrm{softmax}(R_{|V|\times d}h_{t-1} + c)$$

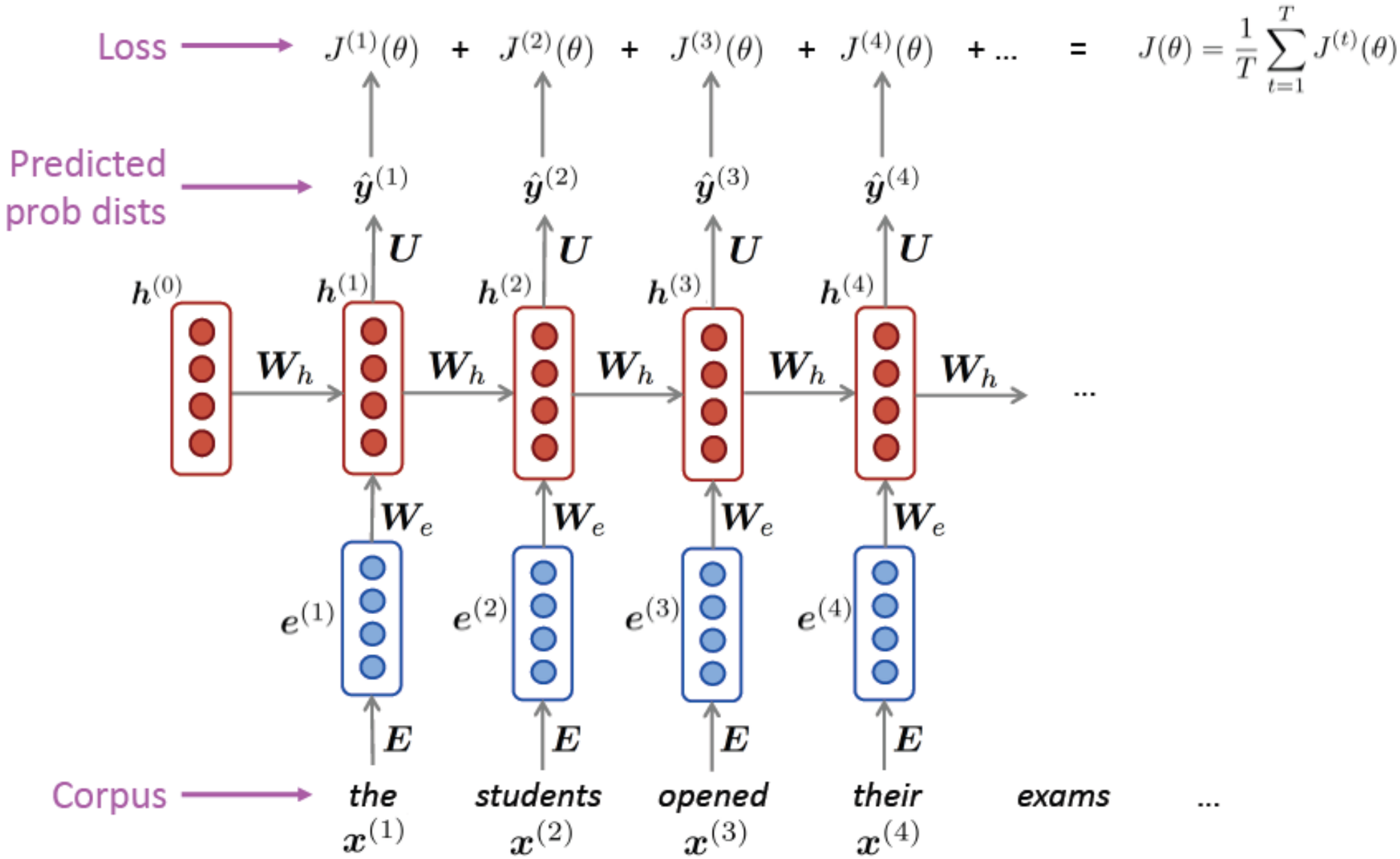
Обучение на выборке

$$-\frac{1}{m}\sum_{i=1}^{m}\sum_{t=1}^{\mathrm{len}(i)}\log p(x_t^{(i)} \mid x_1^{(i)},\ldots,x_{t-1}^{(i)}) \to \min$$

RNN-моделирование языка



RNN-моделирование языка: обучение



http://web.stanford.edu/class/cs224n/

Генерирование текста с помощью RNN

итераций	вывод
100	tyntd-iafhatawiaoihrdemot lytdws e ,tfti, astai f ogoh eoase rrranbyne 'nhthnee e plia tklrgd t o idoe ns,smtt h ne etie h,hregtrs nigtike,aoaenns lng
300	"Tmont thithey" fomesscerliund Keushey. Thom here sheulke, anmerenith ol sivh I lalterthend Bleipile shuwy fil on aseterlome coaniogennc Phe lism thond hon at. MeiDimorotion in ther thize."
700	Aftair fall unsuch that the hall for Prince Velzonski's that me of her hearly, and behs to so arwage fiving were to it beloge, pavu say falling misfort how, and Gogition is so overelical and ofter.
2000	"Why do what that day," replied Natasha, and wishing to himself the fact the princess, Princess Mary was easier, fed in had oftened him. Pierre aking his soul came to the packs and drove up his father-in-law women.

http://karpathy.github.io/2015/05/21/rnn-effectiveness/

Подходы к генерированию

$$p(x_1,\ldots,x_T)=\prod_{t=1}^{T} p(x_t \mid x_1,\ldots,x_{t-1}) \to \max$$

$$\frac{1}{T}\sum_{t=1}^{T} \log p(x_t \mid \ldots) \to \max$$

лучше среднее арифметическое, чтобы не было коротких предложений

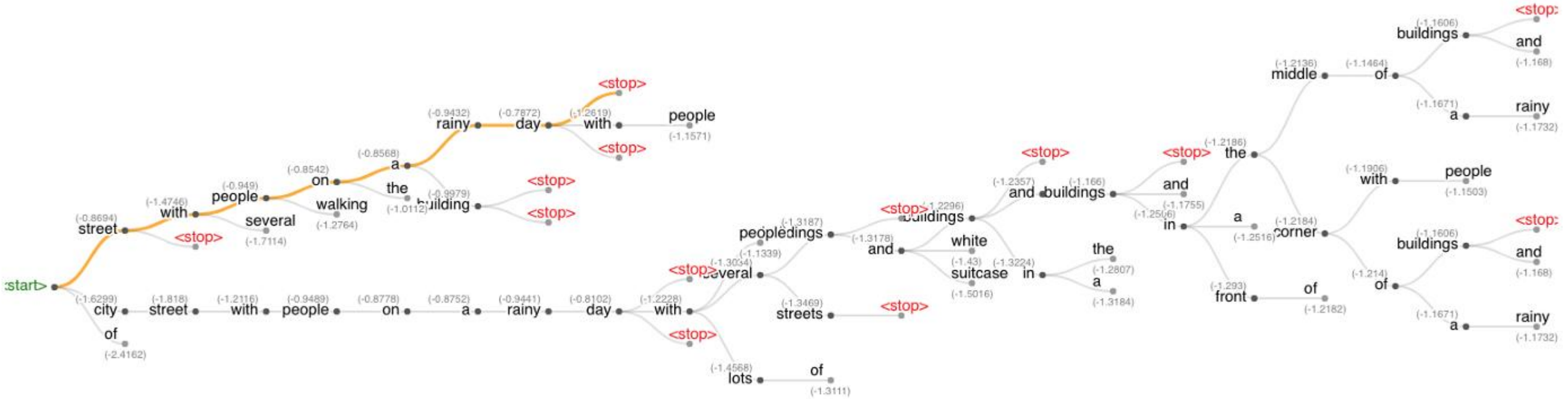
Генерация текста по картинке

Greedy decoder **Large building in the snow in the**
Beam search **Large building in a barn**
Pure sampling decoder **Photo of green boxes in the snow**
Top-k sampling decoder **Large building in the snow away from below**

+ более умные методы (см. дальше)

https://www.katnoria.com/nlg-decoders/

Beam Search (метод луча)



Beam search decoder with k=3 and max steps as 51

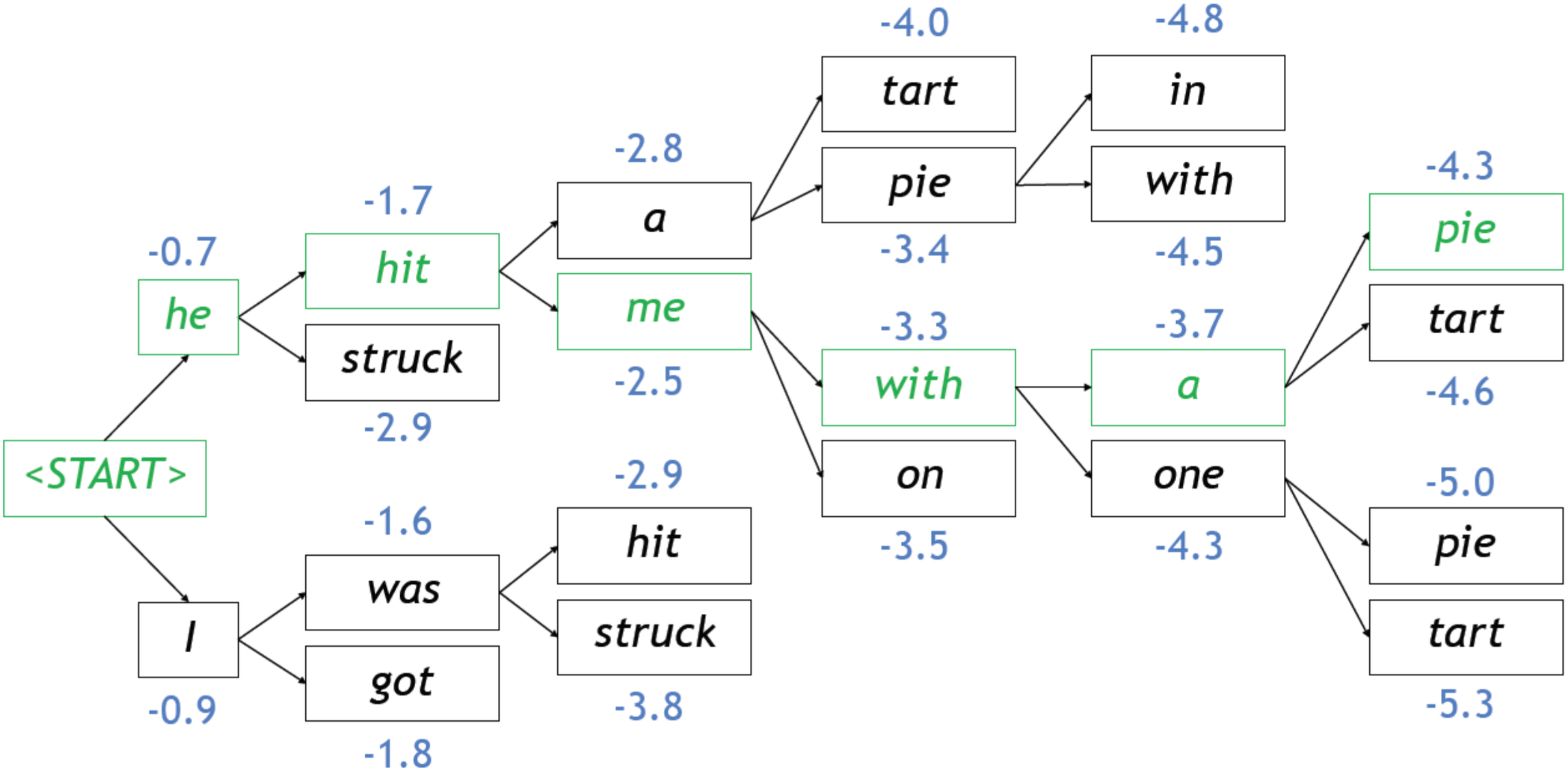
часто продолжают до какой-то максимальной длины

или пока не будет *k* законченных вариантов

https://www.katnoria.com/nlg-decoders/

Sam Wiseman, Alexander M. Rush «Sequence-to-Sequence Learning as Beam-Search Optimization» https://arxiv.org/abs/1606.02960

Beam Search (метод луча)



http://web.stanford.edu/class/cs224n/

Выбор параметра k в методе луча

Beam size	Model response
1	*I love to eat healthy and eat healthy*
2	*That is a good thing to have*
3	*I am a nurse so I do not eat raw food*
4	*I am a nurse so I am a nurse*
5	*Do you have any hobbies?*
6	*What do you do for a living?*
7	*What do you do for a living?*
8	*What do you do for a living?*

Маленькие значения – релевантно, но часто неязыковая фраза,
большие – грамматически верная фраза, но слишком общая

дальше будут стратегии сэмплирования

https://cs224n.stanford.edu/

Оценка языковых моделей

Перплексия (perplexity)

должна быть как можно меньше

$$p(x_1,\ldots,x_T)^{-1/T} = \prod_{t=1}^{T}\left(\frac{1}{p(x_t \mid x_1,\ldots,x_{t-1})}\right)^{1/T}$$

степень для нормировки

в методе луча используют такую же нормировку

Применение LM

Кроме генерации текстов…

Машинный перевод: выбор подходящего варианта

Распознавание речи: выбор подходящего варианта

Проверка текста: нахождение ошибок

Набор текста: подсказка вариантов

История LM

unsupervised pre-trained language models

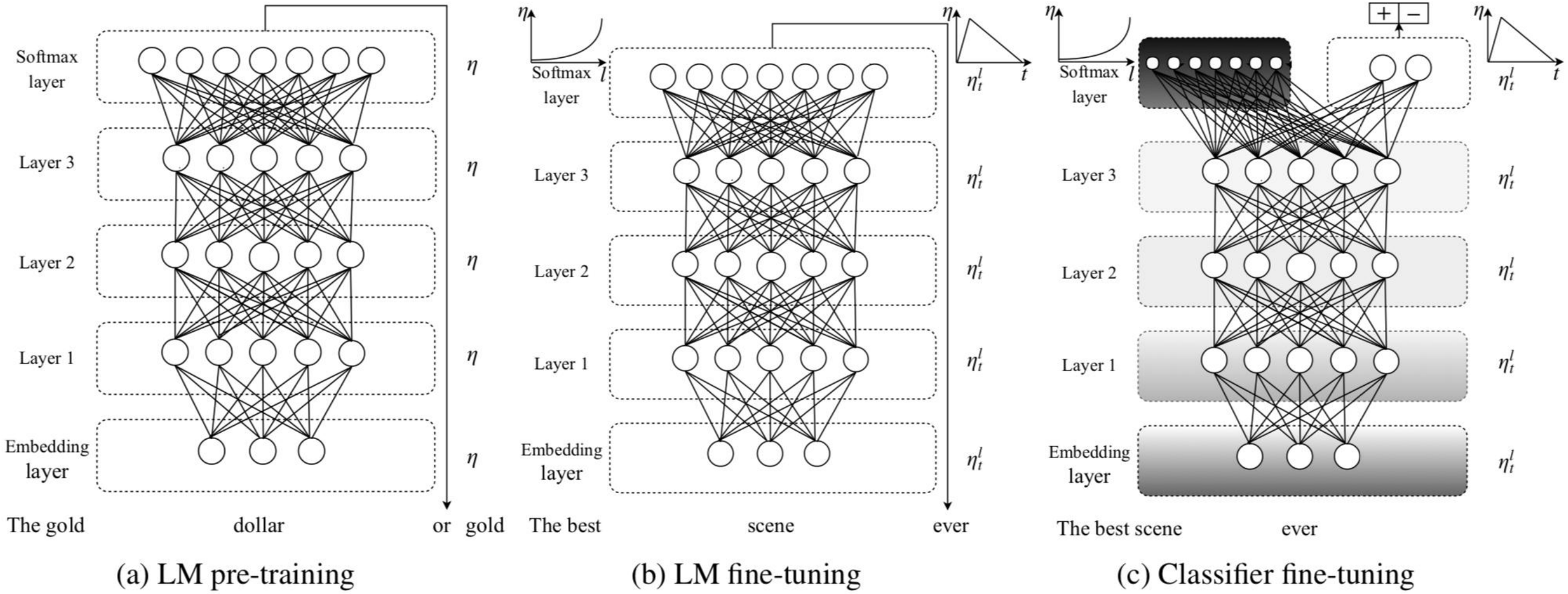
context-independent word embedding

Cove, ELMo, GPT (sentence-level semantic representation)

BERT (predicting shielded words and utilizing Transformer's multi-level self-attention bidirectional modeling capability)

ULMfit

ключевая идея – предобучение и транфер на любую задачу NLP!



Howard and Ruder (2018) Universal Language Model Fine-tuning for Text Classification. https://arxiv.org/pdf/1801.06146.pdf

ULMfit

просто идея

идея трансферного обучения…
Обучить LM на большом корпусе
Доучить на целевой задаче
Доучить классификатор

Разные по слоям темпы обучения
triangular learning rate (STLR) schedule
постепенная разморозка слоёв
при классификации конкатенация состояний, max и mean-пулингов

ERNIE (Enhanced Representation through kNowledge IntEgration)

multi-layer Transformer

WordPiece – посимвольная модель языка

проверена на разных задачах:

natural language inference,

semantic similarity,

named entity recognition,

sentiment analysis,

question-answer matching

результаты лучше Google's BERT!

Yu Sun «ERNIE: Enhanced Representation through Knowledge Integration» //

https://arxiv.org/abs/1904.09223

https://github.com/PaddlePaddle/LARK/tree/develop/ERNIE

ERNIE: «knowledge masking»

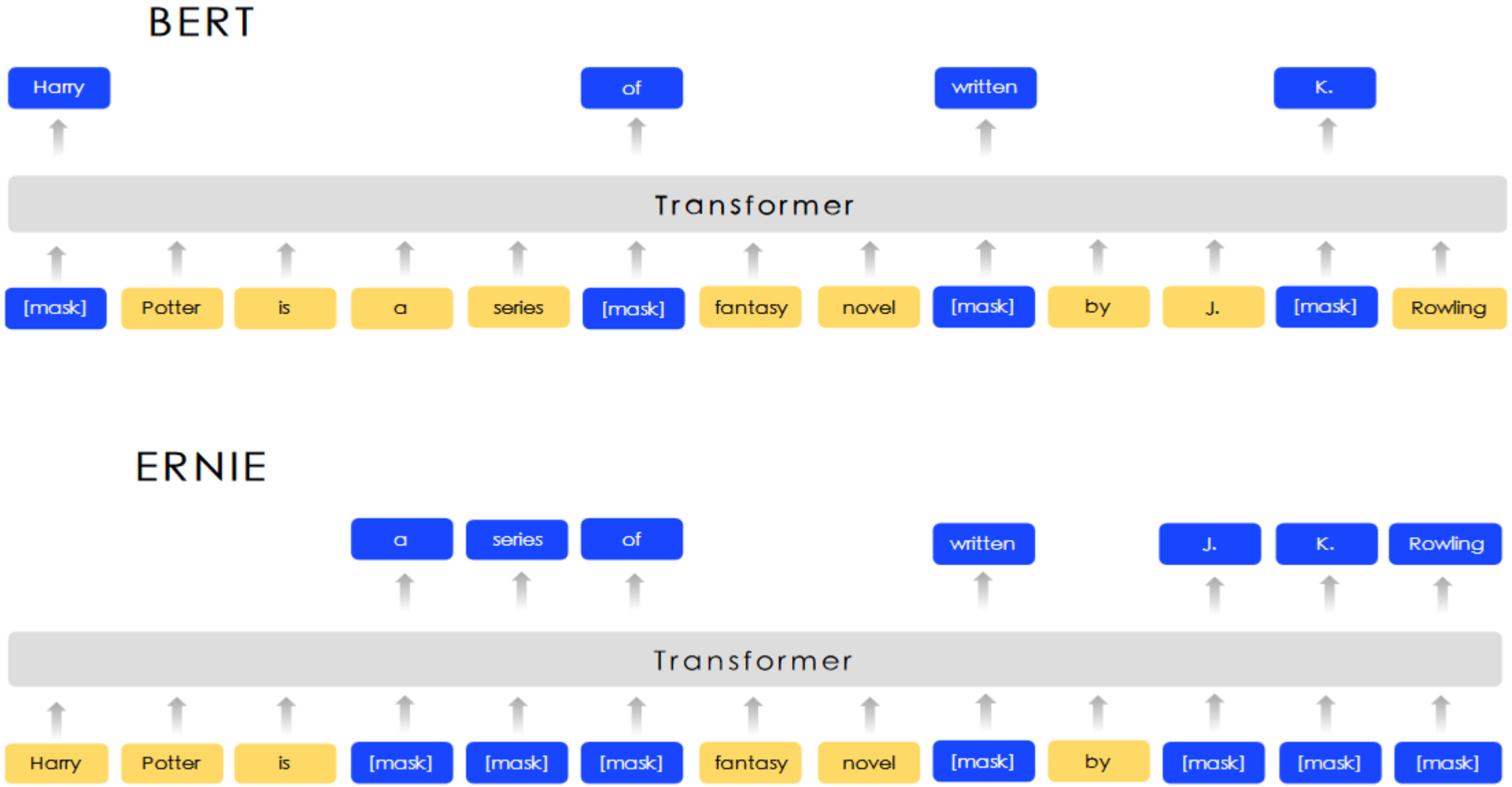


Figure 1: The different masking strategy between BERT and ERNIE

ERNIE: «knowledge masking»

Sentence	Harry	Potter	is	a	series	of	fantasy	novels	written	by	British	author	J.	K.	Rowling
Basic-level Masking	[mask]	Potter	is	a	series	[mask]	fantasy	novels	[mask]	by	British	author	J.	[mask]	Rowling
Entity-level Masking	Harry	Potter	is	a	series	[mask]	fantasy	novels	[mask]	by	British	author	[mask]	[mask]	[mask]
Phrase-level Masking	Harry	Potter	is	[mask]	[mask]	[mask]	fantasy	novels	[mask]	by	British	author	[mask]	[mask]	[mask]

Figure 2: Different masking level of a sentence

Table 2: XNLI performance with different masking strategy and dataset size

pre-train dataset size	mask strategy	dev Accuracy	test Accuracy
10% of all	word-level(chinese character)	77.7%	76.8%
10% of all	word-level&phrase-level	78.3%	77.3%
10% of all	word-level&phrase-leve&entity-level	78.7%	77.6%
all	word-level&phrase-level&entity-level	79.9 %	78.4%

Тестирование

Table 1: Results on 5 major Chinese NLP tasks

Task	Metrics	Bert		ERNIE	
		dev	test	dev	test
XNLI	accuracy	78.1	77.2	79.9 (+1.8)	78.4 (+1.2)
LCQMC	accuracy	88.8	87.0	89.7 (+0.9)	87.4 (+0.4)
MSRA-NER	F1	94.0	92.6	95.0 (+1.0)	93.8 (+1.2)
ChnSentiCorp	accuracy	94.6	94.3	95.2 (+0.6)	95.4 (+1.1)
nlpcc-dbqa	mrr	94.7	94.6	95.0 (+0.3)	95.1 (+0.5)
	F1	80.7	80.8	82.3 (+1.6)	82.7 (+1.9)

GPT – Generative Pre-Training (OpenAI)

- **Transformer**
- **Unidirectional**
- **предсказываем следующее слово**
- **BPE-кодировка**

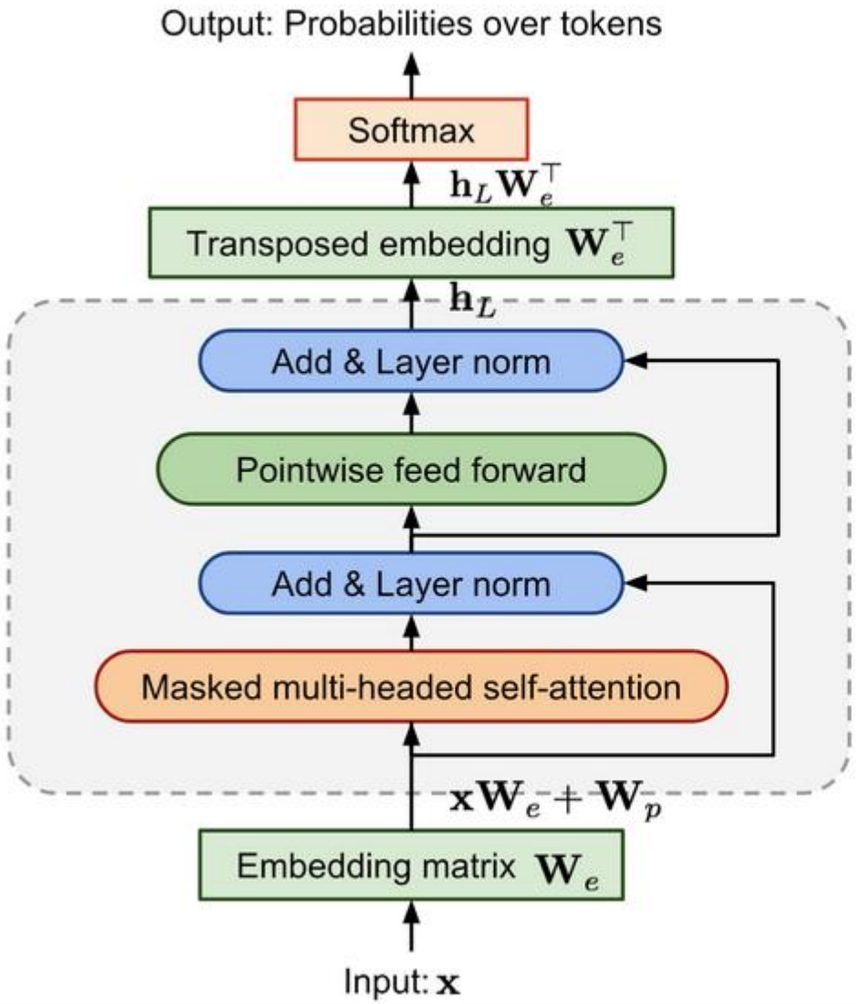
Идея из ELMo, но другая – трансформер – архитектура
у ELMo поднастройка на каждую задачу с помощью коэф. из разных слоёв
у GPT – такого нет

$$\mathcal{L}_{\text{LM}} = -\sum_i \log p(x_i \mid x_{i-k}, \ldots, x_{i-1})$$

предсказание «слева–направо»

Alec Radford Improving Language Understandingby Generative Pre-Training https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf

GPT (OpenAI)



Transformer Block
Repeat x L=12

$$\mathbf{h}_\ell = \text{transformer\_block}(\mathbf{h}_{\ell-1})$$
$$\ell = 1, \ldots, L$$

https://lilianweng.github.io/lil-log/2019/01/31/generalized-language-models.html

GPT (OpenAI)

как осуществляется настройка GPT на конкретную задачу
Пример – классификация

Пропускаем через трансформер
используем скрытое состояние только последнего токен

$$P(y \mid x_1, \ldots, x_n) = \text{softmax}(\mathbf{h}_L^{(n)} \mathbf{W}_y)$$

ошибка = сумма ошибки LM и классификации:

$$\mathcal{L}_{\text{cls}} = \sum_{(\mathbf{x},y)\in\mathcal{D}} \log P(y \mid x_1, \ldots, x_n) = \sum_{(\mathbf{x},y)\in\mathcal{D}} \log \text{softmax}(\mathbf{h}_L^{(n)}(\mathbf{x}) \mathbf{W}_y)$$

$$\mathcal{L}_{\text{LM}} = -\sum_i \log p(x_i \mid x_{i-k}, \ldots, x_{i-1})$$

$$\mathcal{L} = \mathcal{L}_{\text{cls}} + \lambda \mathcal{L}_{\text{LM}}$$

GPT (OpenAI)

другие задачи – тоже не требуют изменения архитектуры

если в задаче несколько входных предложений – они разделяются спецтокеном

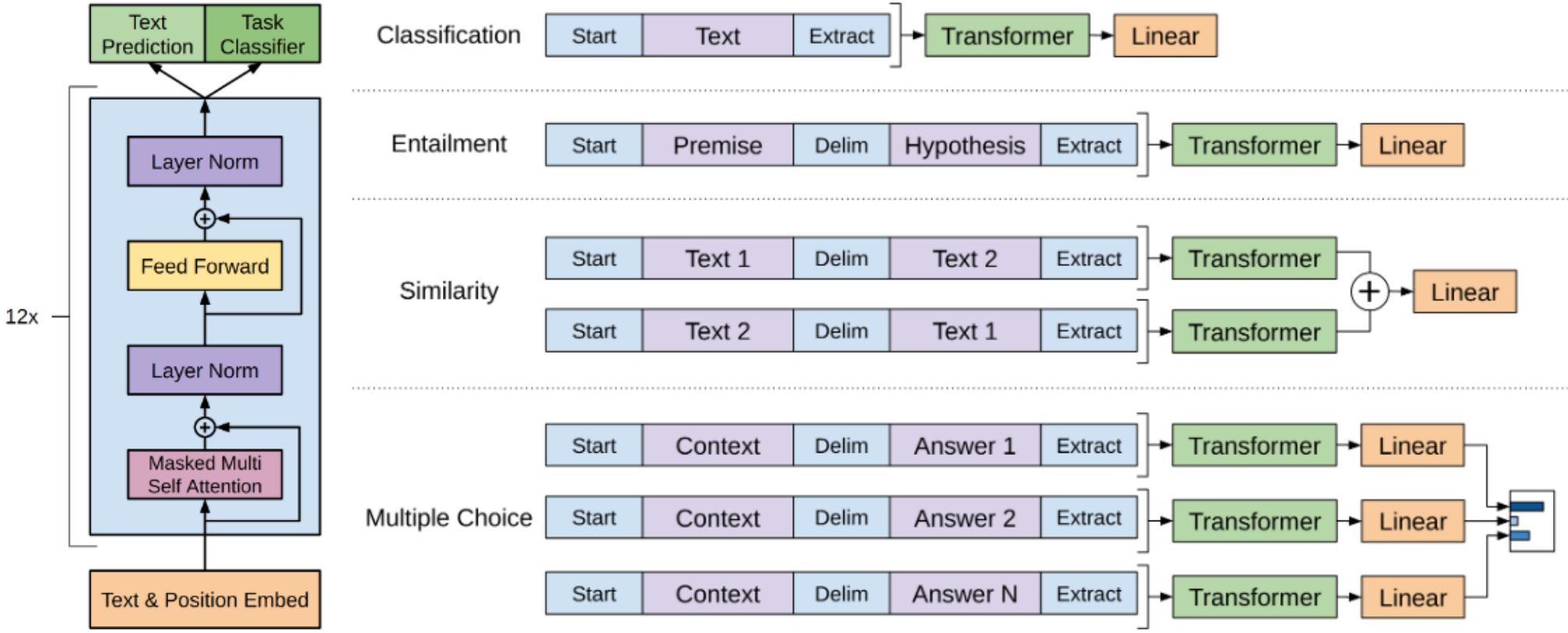
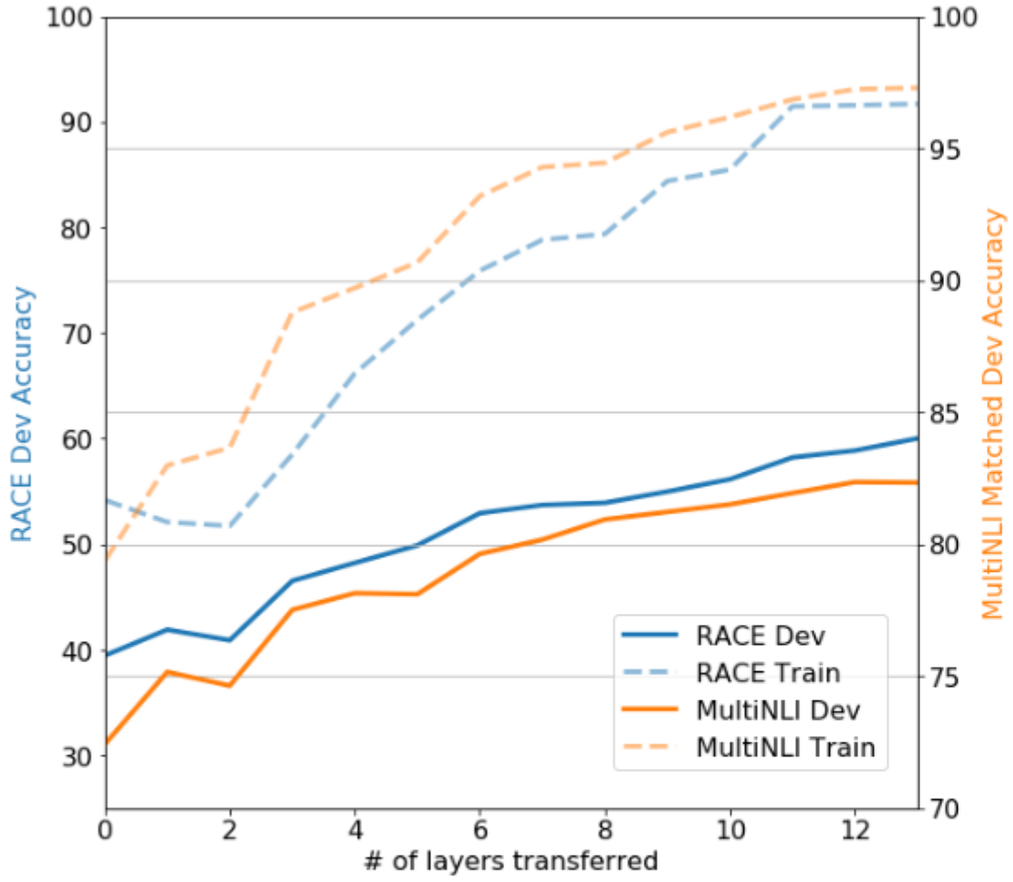


Figure 1: **(left)** Transformer architecture and training objectives used in this work. **(right)** Input transformations for fine-tuning on different tasks. We convert all structured inputs into token sequences to be processed by our pre-trained model, followed by a linear+softmax layer.

GPT (OpenAI)



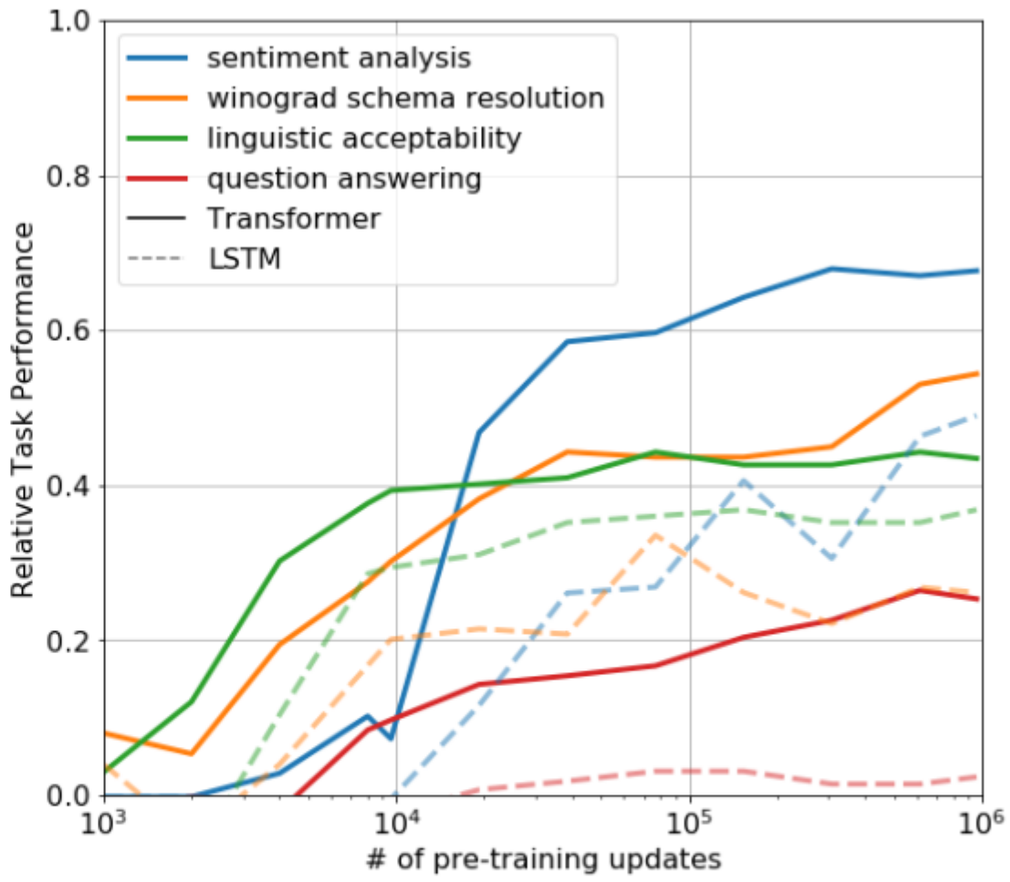


Figure 2: (**left**) Effect of transferring increasing number of layers from the pre-trained language model on RACE and MultiNLI. (**right**) Plot showing the evolution of zero-shot performance on different tasks as a function of LM pre-training updates. Performance per task is normalized between a random guess baseline and the current state-of-the-art with a single model.

GPT2 (2019, OpenAI)

https://blog.openai.com/better-language-models/

https://d4mucfpksywv.cloudfront.net/better-language-models/language_models_are_unsupervised_multitask_learners.pdf

1.5 млрд параметров (10×GPT)

Transformer

SOTA 7 из 8 задач (zero-shot setting – без подстраивания под задачи)

Parameters	Layers	d_{model}
117M	12	768
345M	24	1024
762M	36	1280
1542M	48	1600

Table 2. Architecture hyperparameters for the 4 model sizes.

d – размерность пространства представления токенов

GPT2 (2019, OpenAI)

обучение – новый датасет «WebText»

~ 1 млн web-страниц / 45 млн ссылок / 8 млн. документов 40Гб

ссылки с Reddit ≥3 кармы (т.е. отбором человека)

удалили Wiki ! (чтобы тестировать на других датасетах)

экстракторы текстов:

Dragnet (Peters & Lecocq, 2013) and Newspaper (https://github.com/codelucas/newspaper)

Есть гипотеза, что Wiki плоха для обучения…

GPT2 (2019, OpenAI)

Предобработка

lower-casing

tokenization

out-of-vocabulary tokens

Unicode $\rightarrow$ UTF-8

тут использована: Byte Pair Encoding (BPE)

потом кодируем частые слова и буквы (из которых состоят редкие слова)

GPT2 (2019, OpenAI)

Задачи

- **question answering**
- **machine translation**
- **reading comprehension**
- **summarization**

В основе – Language modeling

$$p(x) = \prod_{i=1}^{n} p(s_n | s_1, ..., s_{n-1})$$

p(output |input, task) – с помощью трансформера

предсказывает следующее слово в предложении
тут нет маскирования как в BERT

GPT2 (2019, OpenAI)

«Task»

~ специальная архитектура (encoders/decoders Kaiser et al., 2017)
~ специальные алгоритмы (inner/outer loop optimization framework of MAML Finn et al., 2017)
~ с помощью языка MQAN – McCann et al. (2018):

«переведи …»
«ответь на вопрос …»
«TL;DR:»

– надо задавать правильные вопросы;)

без дообучения с учителем на специализированных данных!
zero-shot task transfer

GPT2 (2019, OpenAI)

продолжение OpenAI GPT model (Radford et al., 2018)

что нового

Layer normalization → вход каждого под-блока

Layer normalization → после self-attention-блока

другая инициализация

vocabulary = 50,257

context size = 1024

batchsize = 512

Результат 2019 – GPT2

Language Models are Unsupervised Multitask Learners

	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

Table 3. Zero-shot results on many datasets. No training or fine-tuning was performed for any of these results. PTB and WikiText-2 results are from (Gong et al., 2018). CBT results are from (Bajgar et al., 2016). LAMBADA accuracy result is from (Hoang et al., 2018) and LAMBADA perplexity result is from (Grave et al., 2016). Other results are from (Dai et al., 2019).

Alec Radford et. al. «Language Models are Unsupervised Multitask Learners»
https://www.ceid.upatras.gr/webpages/faculty/zaro/teaching/alg-ds/PRESENTATIONS/PAPERS/2019-Radford-et-al_Language-Models-Are-Unsupervised-Multitask-%20Learners.pdf

Question	Generated Answer	Correct	Probability
Who wrote the book the origin of species?	Charles Darwin	✓	83.4%
Who is the founder of the ubuntu project?	Mark Shuttleworth	✓	82.0%
Who is the quarterback for the green bay packers?	Aaron Rodgers	✓	81.1%
Panda is a national animal of which country?	China	✓	76.8%
Who came up with the theory of relativity?	Albert Einstein	✓	76.4%
When was the first star wars film released?	1977	✓	71.4%
What is the most common blood type in sweden?	A	✗	70.6%
Who is regarded as the founder of psychoanalysis?	Sigmund Freud	✓	69.3%
Who took the first steps on the moon in 1969?	Neil Armstrong	✓	66.8%
Who is the largest supermarket chain in the uk?	Tesco	✓	65.3%
What is the meaning of shalom in english?	peace	✓	64.0%
Who was the author of the art of war?	Sun Tzu	✓	59.6%
Largest state in the us by land mass?	California	✗	59.2%
Green algae is an example of which type of reproduction?	parthenogenesis	✗	56.5%
Vikram samvat calender is official in which country?	India	✓	55.6%
Who is mostly responsible for writing the declaration of independence?	Thomas Jefferson	✓	53.3%
What us state forms the western boundary of montana?	Montana	✗	52.3%
Who plays ser davos in game of thrones?	Peter Dinklage	✗	52.1%
Who appoints the chair of the federal reserve system?	Janet Yellen	✗	51.5%
State the process that divides one nucleus into two genetically identical nuclei?	mitosis	✓	50.7%
Who won the most mvp awards in the nba?	Michael Jordan	✗	50.2%
What river is associated with the city of rome?	the Tiber	✓	48.6%
Who is the first president to be impeached?	Andrew Johnson	✓	48.3%
Who is the head of the department of homeland security 2017?	John Kelly	✓	47.0%
What is the name given to the common currency to the european union?	Euro	✓	46.8%
What was the emperor name in star wars?	Palpatine	✓	46.5%
Do you have to have a gun permit to shoot at a range?	No	✓	46.4%
Who proposed evolution in 1859 as the basis of biological development?	Charles Darwin	✓	45.7%
Nuclear power plant that blew up in russia?	Chernobyl	✓	45.7%
Who played john connor in the original terminator?	Arnold Schwarzenegger	✗	45.2%

Table 5. The 30 most confident answers generated by GPT-2 on the development set of Natural Questions sorted by their probability according to GPT-2. None of these questions appear in WebText according to the procedure described in Section 4.

Сравнение

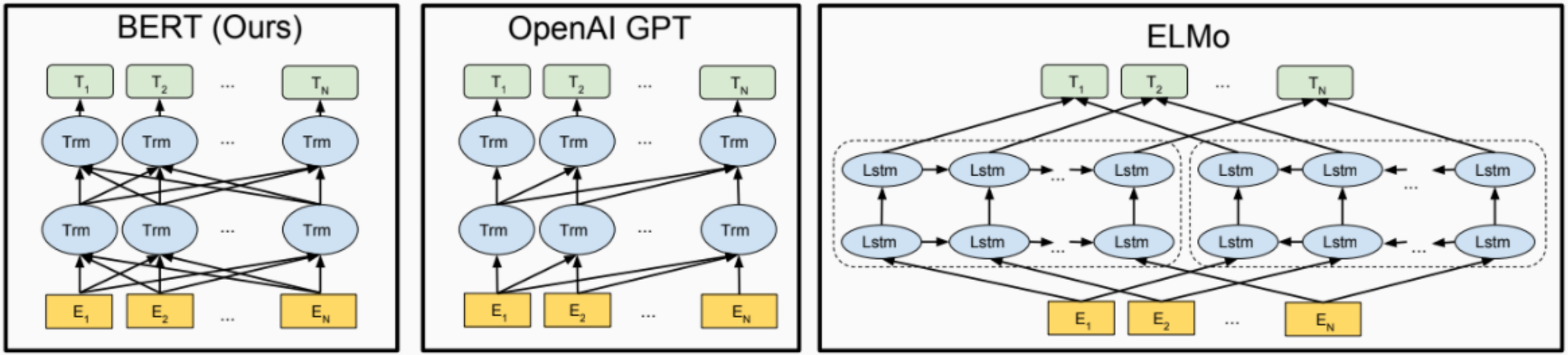


Figure 1: Among three, only BERT representations are jointly conditioned on both left and right context in all layers.

Нейронная дегенерация текста

Обзор на основе работ

Ari Holtzman, Jan Buys, Maxwell Forbes, Yejin Choi «The Curious Case of Neural Text Degeneration» // https://arxiv.org/abs/1904.09751

Sean Welleck, et. al. «Neural Text Generation with Unlikelihood Training» // https://arxiv.org/pdf/1908.04319.pdf

Проблемы «дегенерации текстов»

Рассматриваем задачу генерации текстов с помощью нейронной сети… есть огромные проблемы

Open-ended Generation

есть «контекст» $x_1, \ldots, x_m, m < T$

$$p(x_t, \ldots, x_T) = \prod_{t=1}^{T} p(x_t \mid x_1, \ldots, x_{t-1})$$

open – т.к. не ответ на вопрос, не суммаризация и т.п.

Проблемы «дегенерации текстов»

- **Огромная статистическая разница (distributional differences) между человеческим и машинным текстом [1]**
  Человеческий язык очень специфичен!

- **Декодирование (decoding strategies) сильно влияет на качество [1]**
  выход: Nucleus Sampling

- **Использование правдоподобия и стандартное декодирование приводит к повторам в тексте и неестественности [2]**
  высокочастотные токены – слишком часто
  низкочастотные – слишком редко

Проблемы «дегенерации текстов»

- **Архитектура трансформера приводит к повторениям**
  следующее слово встречается среди 128 предыдущих в 63% случаях, в обычной речи – 49% (+ дальше)

- **Обучение на стандартных корпусах никак не учитывает специфику решаемой задачи**

Проблемы GPT-2

Context:

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.

Continuation (BeamSearch, b=10):

"The unicorns were able to communicate with each other, they said unicorns. a statement that the unicorns. Professor of the Department of Los Angeles, the most important place the world to be recognition of the world to be a of the world to be a of the world to be a of the world to be a of the world to be a of the world to be a of the world to be a of the world to be a of the world to be a of the...

Figure 1: Beam search leads to degenerate text, even when generated from GPT-2-117M, in stark contrast with the admirable quality of the text decoded using *top-k* sampling (Radford et al., 2019). The *continuation* is machine generated, conditioned on the *context* provided by a human. Blue text highlights decoded words that have occurred previously in the text.

Стратегии декодирования, которые максимизировали вероятность текста провалились

$$\prod_{t=m+1}^{T} p(x_t \mid x_1, \ldots x_m, \ldots, x_{t-1}) \to \max$$

жадная (Greedy decoding)

beam search (успешен в non-open-ended generation tasks: machine translation, data-to-text generation, summarization)

Проблемы GPT-2

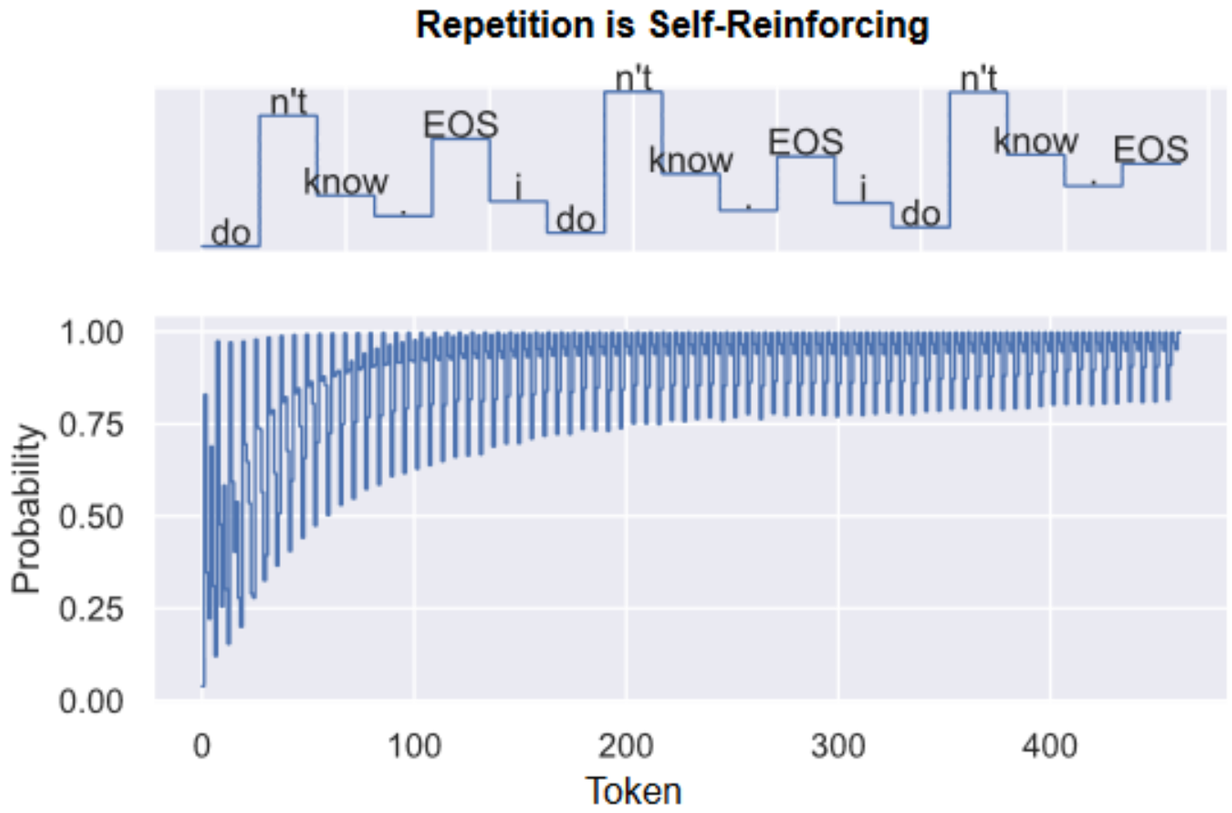


Figure 3: The probability of repetition increases with each instance of repetition, creating a positive-feedback loop.

Генерация превращается в повтор «I don't know»

Это связывают с архитектурой трансформера

Проблемы GPT-2

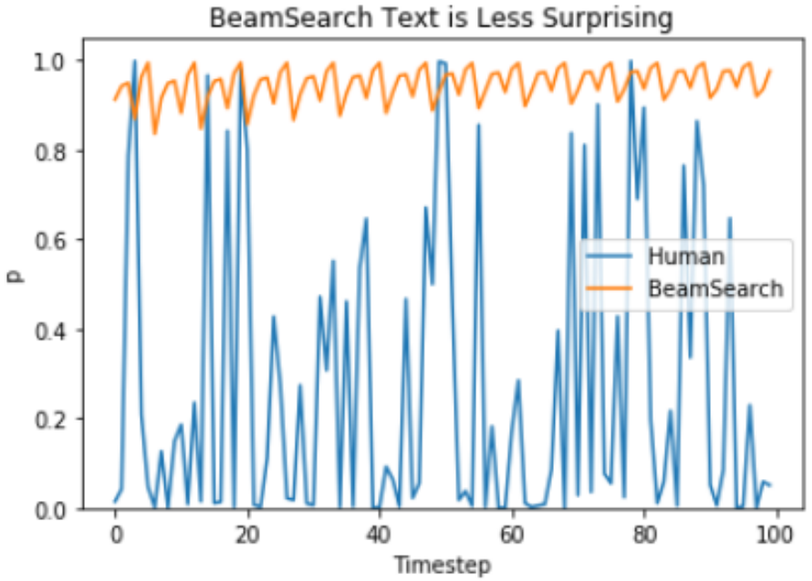


Figure 2: The probability assigned to tokens generated by humans and beam search using GPT-2-117M. Note the increased variance that characterizes the richness of human text.

Human	BeamSearch
…get your hopes up. I saw him once and I have no intention of being near him anytime soon. He sat on the edge, the wind tossing around his hair. It was going to be seriously wind-blown later. I sat down next to him and I was trying to forget the dwarfs mangled body. I shook and hugged myself. Are you cold? He asked, his voice full of concern. I just shrugged and squeezed my eyes shut. I saw Kojas glowing eyes and sword, the…	…looked at the clouds. He looks at the clouds. He looks at the clouds. He looks at the clouds. He looks at the clouds. He looks at the clouds. He looks at the clouds. He looks at the clouds. He looks at the clouds. He looks at the clouds. He looks at the clouds. He looks at the clouds. He looks at the clouds. He looks at the clouds. He looks at the clouds. He looks at the clouds. He looks at the clouds. He looks at the clouds…

Человеческий текст более непредсказуем!

В сгенерированных текстах на 40% меньше уникальных токенов

Это не лечится увеличением обучения (Radford et al., 2019)

Prefix	*Lyrically the song has excerpts of different languages including French , Spanish ,*
GPT2-greedy	German , Italian , Portuguese , Spanish , Portuguese , Portuguese , Portuguese , Portuguese , Portuguese , Portuguese , Portuguese , Portuguese , Portuguese , Portuguese , Portuguese , Portuguese , ...
GPT2-top3	German , Italian , Portuguese , Spanish , Portuguese , Portuguese, Italian and Spanish. It is also available in English, French and Spanish. In addition to its lyrics, the album features a number of original songs, ...
GPT2-nucleus-0.3	German , Italian , Portuguese , Spanish , Portuguese , Italian , Portuguese , Spanish , Portuguese , Spanish , Portuguese , Spanish , Portuguese , Spanish , Portuguese , Spanish , Portuguese , Spanish , Portuguese , Spanish , Portuguese , ...
Prefix	*The first pair of pereiopods is armed with a large , asymm*
GPT2-greedy	etrical, and long-range laser cannon. The second pair is armed with a large , asymmetrical, and long-range laser cannon. The third pair is armed with a large , asymmetrical, and long-range laser cannon. The fourth pair is armed with a large ...
GPT2-top3	etrical, and highly mobile head, and the second pair has a small , asymmetrical, and highly mobile head. The second pair has a large and highly mobile head, and the third pair is armed with a large and highly mobile head. The first pair ...
GPT2-nucleus-0.3	etrical head and a large body. The first pair of pereiopods is armed with a large , asymmetrical head and a large body. The first pair of pereiopods is armed with a large , asymmetrical head and a large body. The first pair of pereiopods is armed ...

Table 1: Top: Degenerate repetition in completions from a state-of-the-art large-scale language model (Radford et al., 2019). The examples contain single-word repetitions, phrase-level repetitions, and structural repetitions where some tokens within a repeating phrase vary. Recently proposed stochastic samplers (top-k, nucleus) exhibit degeneration based on hyper-parameter settings.

Проблемы GPT-2

генерация наиболее вероятных текстов приводит к повторениям в сгенерированных текстах и их неестественности

Широкое и узкое распределения

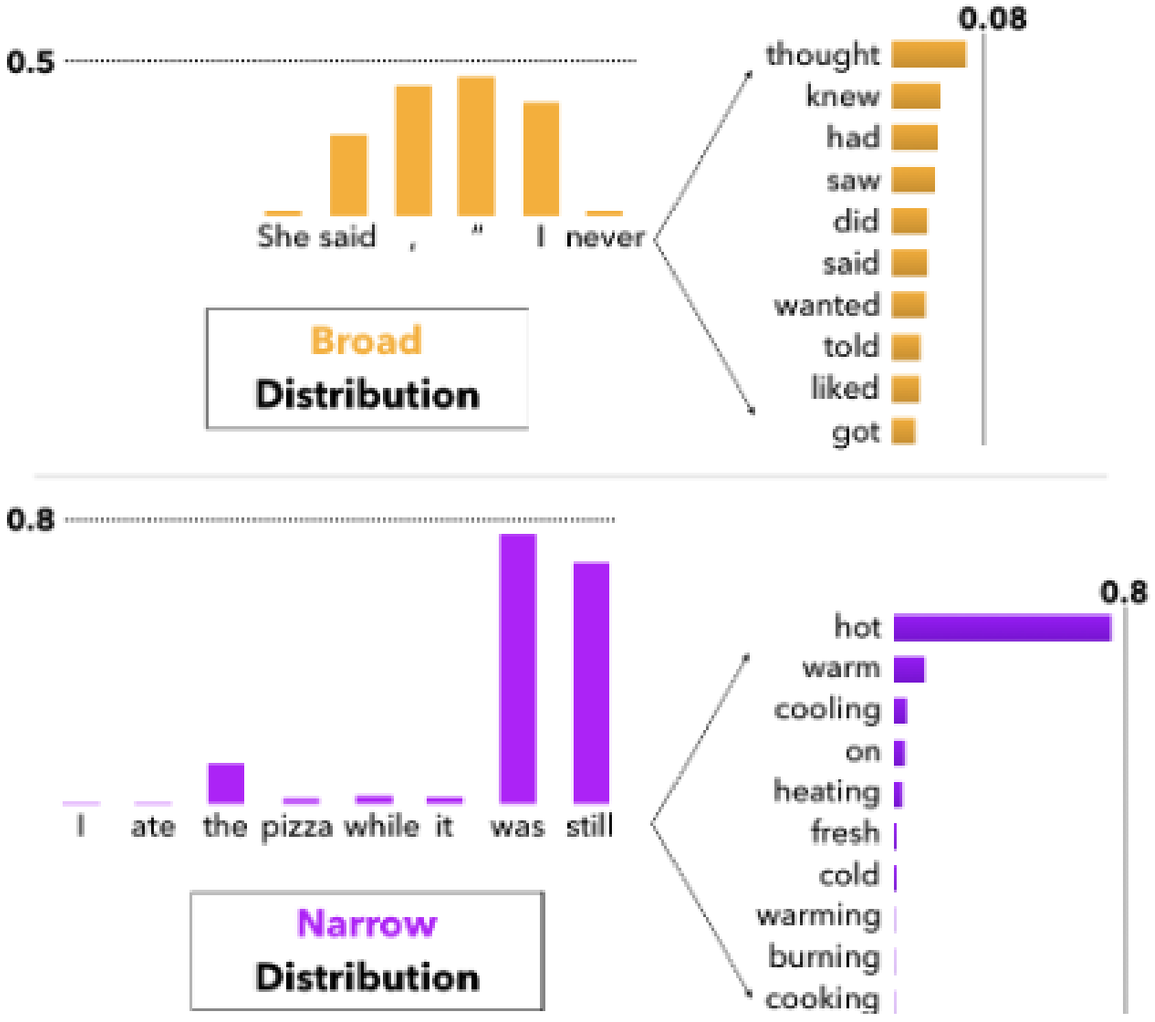


Figure 5: Examples of the probability mass assigned two partial human sentences by GPT, and the resulting **broad** and **narrow** distributions. Broad distributions lead to a large number of tokens with moderate shares of probability mass. In contrast, narrow confidence distributions (less common in open-ended generation) concentrate the overwhelming majority of probability mass into just a few tokens.

Стратегии сэмплирования – для борьбы с дегенерацией
Stochastic Decoding

1. Сэмплирование с температурой

$$p(x_t = x \mid x_1,\ldots,x_{t-1}) = \text{softmax}(u_1 / \tau,\ldots,u_l / \tau)$$

2. Топ-k (Top-k Sampling)

$$p'(x_t = x \mid x_1,\ldots,x_{t-1}) = \begin{cases} p(x_t = x \mid x_1,\ldots,x_{t-1}) / p', & x \in \text{top}(k), \\ 0, & \text{иначе.} \end{cases}$$

3. Nucleus (Top-p) Sampling

вместо используем $\text{top}(k)$

$$\sum_{x \in \text{sort}} p(x_t = x \mid x_1,\ldots,x_{t-1}) \geq p$$

но есть мнение [2], что сами вероятности неадекватны

Стратегии сэмплирования

В идеале детерминистические и стохастические – в зависимости от задачи

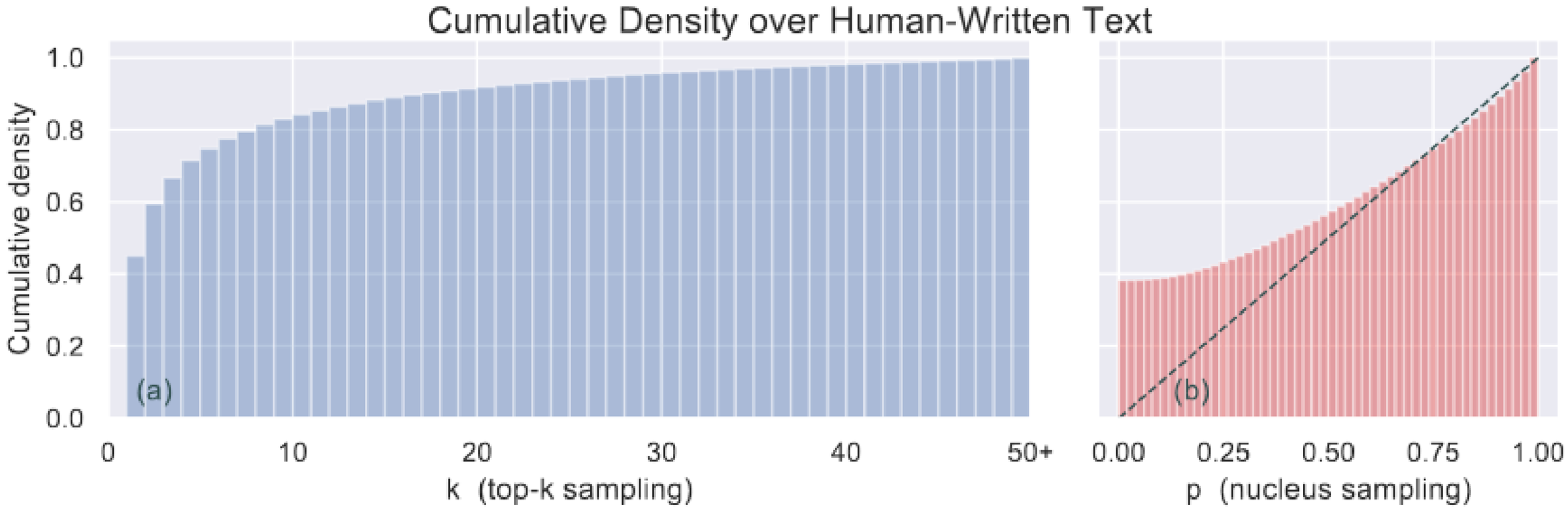


Figure 7: The left-hand side graph illustrates the diminishing returns received as the k increases in top-k, which contrasts with the increasing returns of Nucleus Sampling (right) that allows values of p close to 1 to act very similarly to pure sampling without risk of sampling from the low-confidence tail. The height of a bar encodes the cumulative density of the minimum value of k (for top-k sampling) or p (for Nucleus sampling) required to assign a non-zero probability to the *gold* next word prediction over a corpus of human-written text.

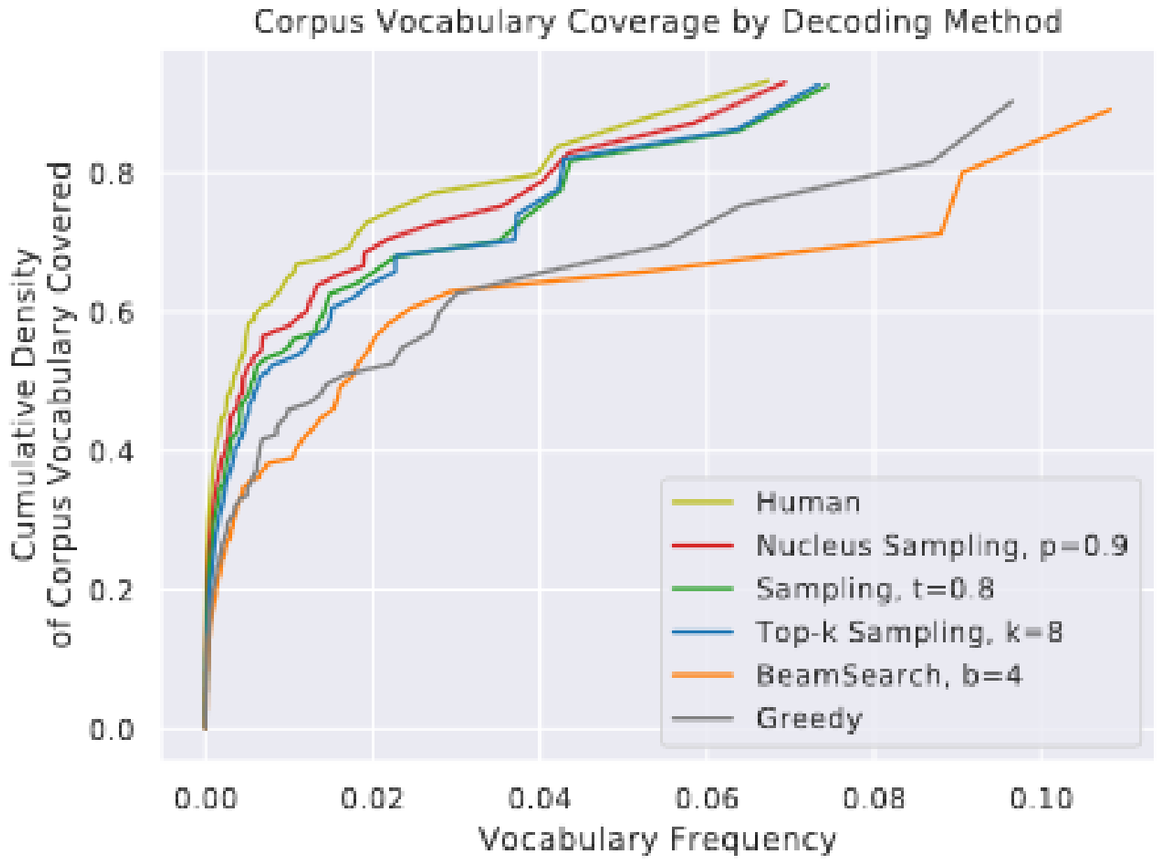


Figure 8: A chart describing the distributional differences between n-gram frequencies of human and machine text. The complete separation of likelihood maximization and stochastic methods, stochastic clearly closer to human, indicates an inherent issue with a likelihood maximization as a decoding objective.

🗣 **Context**: On days when he woke early, the president liked to have coffee in the oval office. There was something about watching the sky grow lighter and lighter as you sat your pajama'd behind in the most powerful chair in the free world, sipping marine strength coffee, that you just couldn't achieve anywhere else.

💰 **Greedy**: He'd been in the oval office for over a year now, and he'd never been so happy to be there. He'd been in the oval office for almost a year, and he'd never been so happy to be there.

🚦 **BeamSearch** (*k*=4): The President was a man of his word. He was a man of his word. He was a man of his word. He was a man of his word. He was a man of his word. He was a man of his word.

🥄 **Sampling** (*t*=1.0): You couldn't be sure if that's what you were really doing, and If you decided to take the day off. The president wanted you to take the day off, but he wanted to maintain a curfew and use his influence wisely.

🇰 **Top-*k* Sampling** (*k*=8): "I'm sorry, sir." "No, it's okay, son," the president said. "I understand." "You're going to have to make a special trip down there to get that kid. He has no idea where he's going."

⚛ **Nucleus Sampling** (*p*=0.9): But that wasn't what drew the president's attention. He'd been seated for maybe a minute when he noticed the other man. What was the guy doing here?

🗣 **Gold**: He was therefore disagreeably surprised to find a man in an understated grey suit sitting in that selfsame chair sipping tea. The president turned around and went looking for his chief of staff.

Unlikelihood training [2]

Ориентация на оптимизацию правдоподобия – ошибка!

из-за неё повторы и неестественность

и глобально, не умеем максимизировать правдоподобие всей последовательности

Unlikelihood loss [2]

Идея: определяем «запрещённое множество» – штрафуем слова оттуда

The key idea behind the unlikelihood loss is decreasing the model's probability of certain tokens, called *negative candidates*. Given a sequence $(x_1, \ldots, x_T)$ and a set of negative candidate tokens $\mathcal{C}^t = \{c_1, \ldots, c_m\}$, where each $c_j \in \mathcal{V}$, we define the unlikelihood loss for step t as:

$$\mathcal{L}_{\text{UL}}^t(p_\theta(\cdot|x_{<t}), \mathcal{C}^t) = -\sum_{c \in \mathcal{C}^t} \log(1 - p_\theta(c|x_{<t})). \tag{4}$$

The loss decreases as $p_\theta(c|x_{<t})$ decreases.

Token level objective

Реализация идеи:

Given a sequence $(x_1, \ldots, x_T)$, the token-level objective applies the unlikelihood loss to a set of negative candidates at each time-step of maximum likelihood training:

$$\mathcal{L}^t_{\text{UL-token}}(p_\theta(\cdot|x_{<t}), \mathcal{C}^t) = -\alpha \cdot \underbrace{\sum_{c \in \mathcal{C}^t} \log(1 - p_\theta(c|x_{<t}))}_{\text{unlikelihood}} - \underbrace{\log p_\theta(x_t|x_{<t})}_{\text{likelihood}}. \quad (5)$$

We propose a candidate set which uses the previous context tokens:

$$\mathcal{C}^t_{\text{prev-context}} = \{x_1, \ldots, x_{t-1}\} \setminus \{x_t\}. \quad (6)$$

Intuitively, the unlikelihood loss with this candidate set makes (i) incorrect repeating tokens less likely, as the previous context contains potential repeats, and (ii) frequent tokens less likely, as these tokens appear often in the previous context. This candidate set is also efficient to compute and requires no additional supervision.

Sequence level objective

Штраф за повторы последовательностей

Intuitively, the negative candidates can identify problematic tokens for the loss to penalize. We choose to penalize repeating n-grams in the continuation:

$$\mathcal{C}^t_{\text{repeat-n}} = \{x_t\} \text{ if } (x_{t-i}, \dots, x_t, \dots, x_{t+j}) \in x_{<t-i} \text{ for any } (j-i) = n, i \le n \le j, \qquad (10)$$

which says that the token x_t is the (single) negative candidate for step t if it is part of a repeating n-gram.

Evaluation metrics

Repetition As a token-level metric for repetition, we use the fraction of next-token (top-1) predictions that occur in the previous ℓ tokens (**rep/**ℓ). That is, given a validation set $\mathcal{D}$ of length-T sequences,

$$\frac{1}{|\mathcal{D}|T}\sum_{\mathbf{x}\in\mathcal{D}}\sum_{t=1}^{T}\mathbb{I}\left[\arg\max p_\theta(x|\mathbf{x}_{<t}) \in \mathbf{x}_{t-\ell-1:t-1}\right]. \tag{11}$$

We use the portion of duplicate n-grams (**seq-rep-n**) in a generated sequence to measure sequence-level repetition. That is, for a continuation $\mathbf{x}_{k+1:k+N}$ we compute,

$$1.0 - \frac{|\text{unique n-grams}(\mathbf{x}_{k+1:k+N})|}{|\text{n-grams}|}, \tag{12}$$

and average over continuations. **seq-rep-n** is zero when the continuation has no repeating n-grams, and increases towards 1.0 as the model repeats. We compute **seq-rep-n** on the continuation rather than the full completion since we are interested in measuring degenerate repeats in the continuation.

Evaluation metrics

Token Distribution We quantify a model's predicted token distribution using the number of unique tokens. As a token-level metric (**uniq**), we use the number of unique next-token predictions on the validation set, i.e. $|\{\arg\max p(x_t|x_{<t}) \mid x_{<t} \in \mathcal{D}_{\text{valid}}\}|$. As a sequence-level metric (**uniq-seq**) we use the number of unique tokens in continuations of prefixes from the validation set (subsection 6.1).

Language Modeling Quality To quantify a model's language modeling quality we use the standard perplexity metric (**ppl**), and next-token greedy prediction accuracy (**acc**).

Experiments

		seq-rep-4
Prefix	*... Lyrically the song has excerpts of different languages including French , Spanish*	
$\mathcal{L}_{\text{MLE}}$	, Italian , Spanish , Italian , Spanish , Italian , Spanish , Spanish , Portuguese	0.744
$\mathcal{L}_{\text{UL-token+seq}}$	, and German . In the first verse , the protagonist sings about being a " girl who 's been in love with someone else " , while the second verse describes the relationship between the protagonist and her lover . In the third verse , the protagonist sings about	0.063
Prefix	*... with timely advice from General Lee , adopted a strong defensive position that was virtually*	
$\mathcal{L}_{\text{MLE}}$	impregnable . Lee 's forces were well prepared for the night , and the battle was a disaster . Lee 's forces were well prepared for the night , and the battle was a disaster . Lee 's forces were well prepared for the night , and the battle was	0.617
$\mathcal{L}_{\text{UL-token+seq}}$	impregnable . The Americans were also able to use the boats to bombard the fort from the rear , and the guns fired at the British ships in the harbor . The British bombardment began at 9 : 30 am , when the first shots were fired on the	0.000

Experiments

Prefix	*... starboard engines and was going to crash . " We 're going in ,"*	
$\mathcal{L}_{\text{MLE}}$	he said . " We 're going to crash . We 're going to crash . We 're going to crash . We 're going to crash . We 're going to crash . We 're going to crash . We 're going to crash . We 're going to	0.787
$\mathcal{L}_{\text{UL-token+seq}}$	Hood said . " I 'm going to make sure we 're going to get back to the water . " The order to abandon ship was given by Admiral Beatty , who ordered the remaining two battlecruisers to turn away . At 18 : 25 , Hood turned his	0.000
Prefix	*... career - high 27 points on 8 - for - 11 shooting with three rebounds*	
$\mathcal{L}_{\text{MLE}}$	and two assists . On January 3 , 2012 , he was named to the 2012 13 All - Atlantic 10 first team . On February 3 , 2012 , he was named to the Atlantic 10 first team . On February 5 , 2012 , he was named	0.277
$\mathcal{L}_{\text{UL-token+seq}}$	and a career - high 7 assists against the Minnesota Timberwolves . On February 3 , 2012 , he was named to the 2012 All - NBA First Team . On March 7 , 2012 , he was named one of five finalists for the Naismith Award , which is	0.064

Table 2: Example greedy completions, showing the last 15 tokens of a 50 token prefix, and 50-token continuations. The completions show representative examples of the MLE model's degenerate single token repetition (top), phrase-level repetition (middle two), and 'structural' repetition (bottom), as well as the proposed method's ability to fix these degenerate behaviors.

Model	search	seq-rep-4	uniq-seq	ppl	acc	rep	wrep	uniq
$\mathcal{L}_{\text{MLE}}$	greedy	.442	10.2k	**24.52**	**.401**	.619	.345	11.5k
	beam	.507	9.2k					
$\mathcal{L}_{\text{UL-token}}$	greedy	**.267**	**12.0k**	25.68	.397	**.568**	**.304**	**12.3k**
	beam	**.330**	**11.0k**					
$\mathcal{L}_{\text{UL-seq}}$	greedy	.134	11.7k	**23.95**	**.408**	.606	.331	12.4k
	beam	.015	16.1k					
$\mathcal{L}_{\text{UL-token+seq}}$	greedy	**.051**	**14.6k**	25.37	.401	**.553**	**.288**	**13.3k**
	beam	**.012**	**16.9k**					
Human	-	.005	18.9k	-	-	.479	-	18.9k

Table 3: Results for token-level objectives (upper) and sequence-level fine-tuning (lower) according to sequence-level (left) and token-level (right) metrics using the **validation subset of wikitext-103**. The best metrics achieved by both token-level and sequence-level models using both greedy and beam search are shown in bold. rep and wrep use $\ell = 128$; relative rankings hold for other ℓ.

UL-token – было

UL-seq – (10)

UL-token+seq – их комбинация

Model	search	seq-rep-4	uniq-seq	ppl	acc	rep	wrep	uniq
$\mathcal{L}_{\text{MLE}}$	greedy	.453	10.4k	**25.701**	**.394**	.629	.355	11.7k
	beam	.528	9.4k					
$\mathcal{L}_{\text{UL-token}}$	greedy	**.276**	**12.5k**	27.020	.390	**.575**	**.309**	**12.6k**
	beam	**.336**	**11.6k**					
$\mathcal{L}_{\text{UL-seq}}$	greedy	.144	12.1k	**25.112**	**.401**	.613	.338	12.7k
	beam	.014	17.5k					
$\mathcal{L}_{\text{UL-token+seq}}$	greedy	**.059**	**15.2k**	26.839	.394	**.559**	**.293**	**13.6k**
	beam	**.012**	**18.1k**					

Table 4: Results for token-level objectives (upper) and sequence-level fine-tuning (lower) according to sequence-level (left) and token-level (right) metrics using the **test subset of Wikitext-103**.

Instructions: You will be shown an excerpt from a Wikipedia article, with two possible continuations. **DO NOT** try to find the original article on Wikipedia.

Please read the excerpt and the continuations, and select which continuation is **more natural**. Focus on the **quality of the writing**, and try to **disregard factual errors**.

Excerpt:
...(who left in early 1980). The organization flew" the first international relief airlift to Cambodia since 1975" , delivering medicine to Phnom - Penh. Operation California had airlifted more than $ 3 million worth of aid by October 1979. Since then,...

... Operation USA has become a highly acclaimed aid organization that is involved in helping people in different ways around the world. In 1982, Operation California sent" the first private airlift from the U.S. to Poland" , delivering 200, 000 of medical supplies and medicine ; that year Operation California also airlifted medical supplies to Lebanon. In 1983, Operation California delivered aid to the children of Vietnam and Cambodia. Operation California provided aid to the earthquake victims in Mexico City in 1985, as well as working in cooperation with the

...the UN has provided humanitarian assistance to the country. The UN also provides humanitarian assistance to the country.

Humanitarian aid

The UN also provides humanitarian assistance to the country. The UN also provides humanitarian assistance to the country.

Humanitarian aid

The UN also provides humanitarian assistance to the country.

Humanitarian aid

The UN also provides humanitarian assistance to the country.

Which of these two continuations is **more natural**?

○ Continuation A is **more natural**.

○ Continuation B is **more natural**.

Please enter a very brief reason (a few words or a sentence) explaining your choice:

(If you do not give a reason, your hit may be rejected)

You must ACCEPT the HIT before you can submit the results.

Model 1		Model 2	Win rate
$\mathcal{L}_{\text{MLE}}$ baseline		$\mathcal{L}_{\text{UL-token}}$	62%
$\mathcal{L}_{\text{MLE}}$ baseline		$\mathcal{L}_{\text{UL-seq}}$	*70%
$\mathcal{L}_{\text{MLE}}$ baseline	*beaten by*	$\mathcal{L}_{\text{UL-token+seq}}$	*84%
$\mathcal{L}_{\text{UL-token}}$		$\mathcal{L}_{\text{UL-token+seq}}$	*72%
$\mathcal{L}_{\text{UL-seq}}$		$\mathcal{L}_{\text{UL-token+seq}}$	58%
$\mathcal{L}_{\text{MLE}}$ baseline		Reference	*74%
$\mathcal{L}_{\text{UL-token}}$	*beaten by*	Reference	*68%
$\mathcal{L}_{\text{UL-seq}}$		Reference	56%
$\mathcal{L}_{\text{UL-token+seq}}$		Reference	52%

Table 5: **Human evaluation results**. Human evaluators preferred generations from our models over the baseline model, and $\mathcal{L}_{\text{UL-token+seq}}$ outperformed our other variants. The sequence-tuned models approach human-level performance. Comparisons marked with * are statistically significant (one-sided binomial test, $p < .05$).

WritingPrompts dataset of (Fan et al., 2018).

Each example consists of a contextof 5 sentences with a maximum of 200 tokens;the task is to continue the text by generating the200 next tokens (the continuation).

XLNet: Generalized Autoregressive Pretraining for Language Understanding

	BERT	RoBERTa	DistilBERT	XLNet
Size (millions)	**Base**: 110 **Large**: 340	**Base**: 110 **Large**: 340	**Base:** 66	**Base**: ~110 **Large**: ~340
Training Time	**Base**: 8 x V100 x 12 days* **Large:** 64 TPU Chips x 4 days (or 280 x V100 x 1 days*)	**Large:** 1024 x V100 x 1 day; 4-5 times more than BERT.	**Base**: 8 x V100 x 3.5 days; 4 times less than BERT.	**Large:** 512 TPU Chips x 2.5 days; 5 times more than BERT.
Performance	Outperforms state-of-the-art in Oct 2018	2-20% improvement over BERT	3% degradation from BERT	2-15% improvement over BERT
Data	16 GB BERT data (Books Corpus + Wikipedia). 3.3 Billion words.	160 GB (16 GB BERT data + 144 GB additional)	16 GB BERT data. 3.3 Billion words.	**Base**: 16 GB BERT data **Large**: 113 GB (16 GB BERT data + 97 GB additional). 33 Billion words.
Method	BERT (Bidirectional Transformer with MLM and NSP)	BERT without NSP**	BERT Distillation	Bidirectional Transformer with Permutation based modeling

Итог

ULMfit	01.2018	fast.ai	1 GPU-дней
GPT	06.2018	OpenAI	240 GPU-дней
BERT	10.2018	Google AI	265 TPU-дней
GPT-2	02.2019	OpenAI	>2048 TPU-дней

Языковые модели – предсказывают следующее слово

есть простые n-граммные, если рекуррентные / трансформерные – позволяют учесть весь контекст

Ссылки

хороший курс «Natural Language Processing with Deep Learning»

http://web.stanford.edu/class/cs224n/