



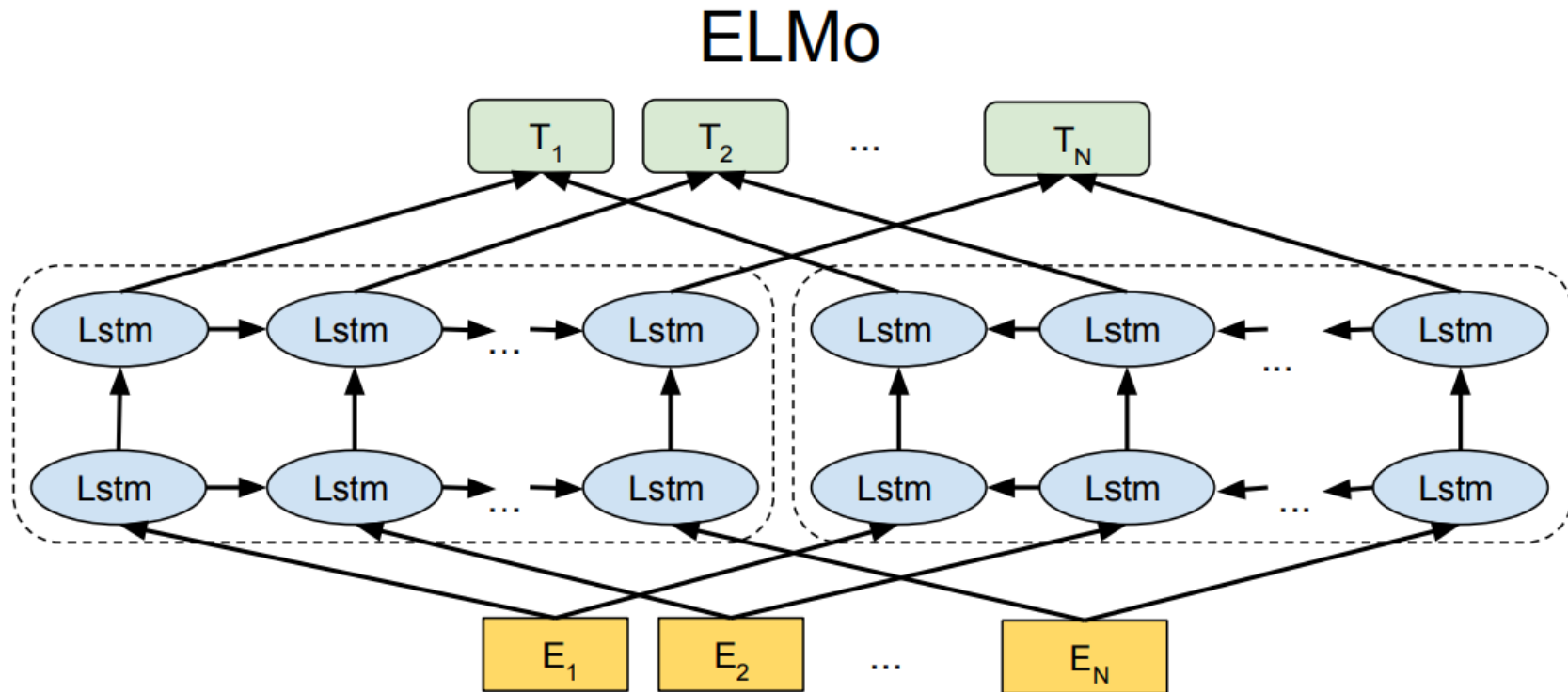
BERT **(Bidirectional Encoder** **Representations from** **Transformers)**

Камлык Эрик

Предобучение

- Обучение с нуля требует много времени и большое количество размеченных данных
- В задачах NLP модели должны выучить общие лингвистические знания
- Предлагается использовать предобученные модели
- Два подхода: feature based, fine-tuning

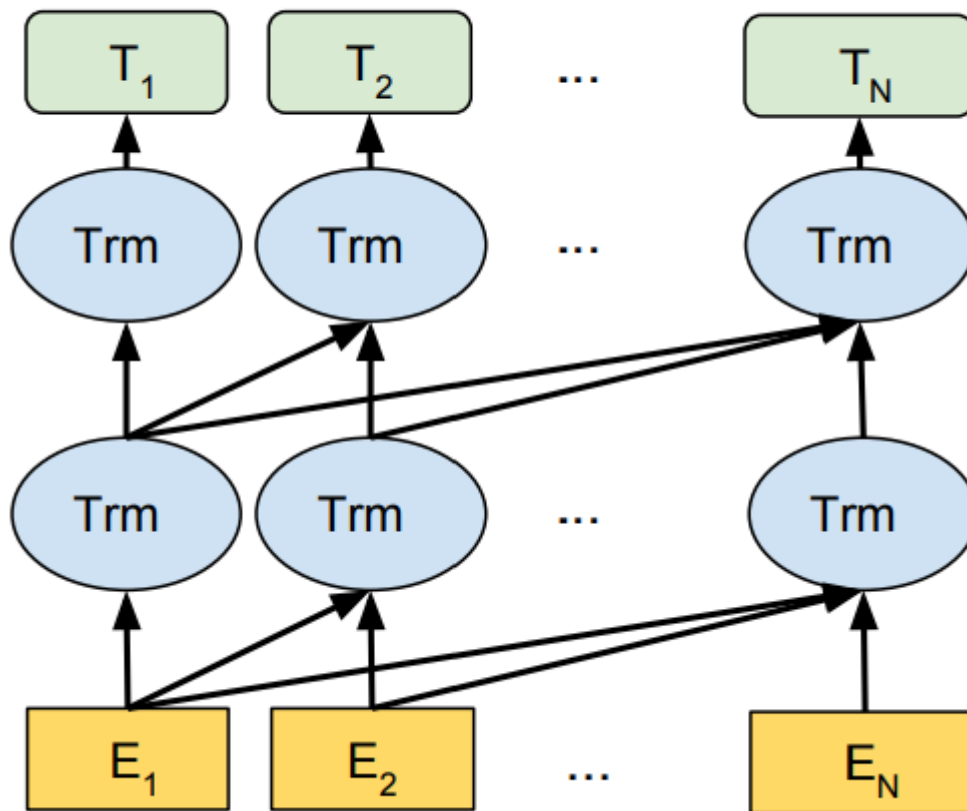
Feature based подход



- Двухнаправленная
- Выход слоев используются как признаки

Fine-tuning

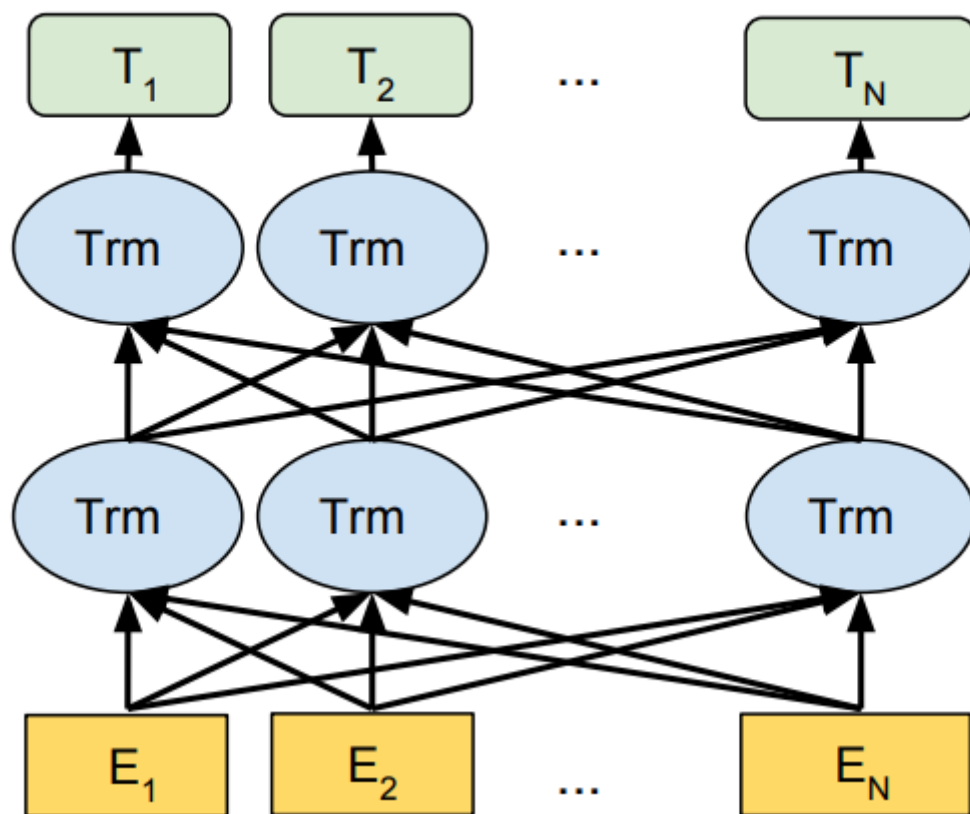
OpenAI GPT



- Обучалась на задаче предсказания следующего слова
- Отсюда однонаправленность

BERT

BERT (Ours)



- Обучалась на задаче заполнения пропусков (Masked LM)
- Учитывает контекст по двум направлениям

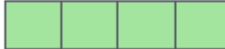
Self-attention

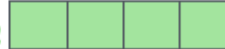
Input

Thinking

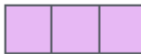
Machines

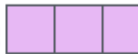
Embedding

x_1 

x_2 

Queries

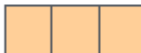
q_1 

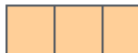
q_2 



W^Q

Keys

k_1 

k_2 



W^K

Values

v_1 

v_2 



W^V

Self-attention

Input

Embedding

Queries

Keys

Values

Score

Divide by 8 ($\sqrt{d_k}$)

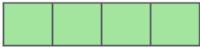
Softmax

Softmax

X
Value

Sum

Thinking

x_1 

q_1 

k_1 

v_1 

$q_1 \cdot k_1 = 112$

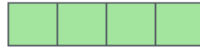
14

0.88

v_1 

z_1 

Machines

x_2 

q_2 

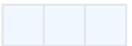
k_2 

v_2 

$q_1 \cdot k_2 = 96$

12

0.12

v_2 

z_2 

Multi-head attention

1) This is our input sentence*

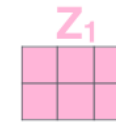
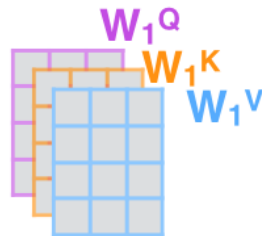
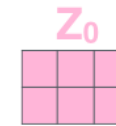
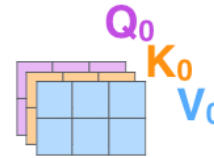
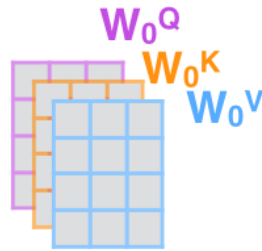
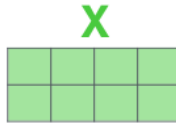
2) We embed each word*

3) Split into 8 heads. We multiply X or R with weight matrices

4) Calculate attention using the resulting $Q/K/V$ matrices

5) Concatenate the resulting Z matrices, then multiply with weight matrix W^O to produce the output of the layer

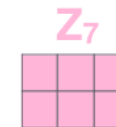
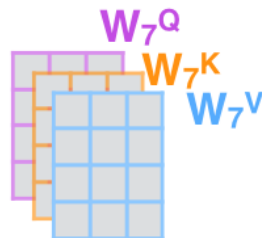
Thinking
Machines



...

...

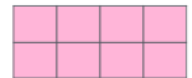
...



W^O

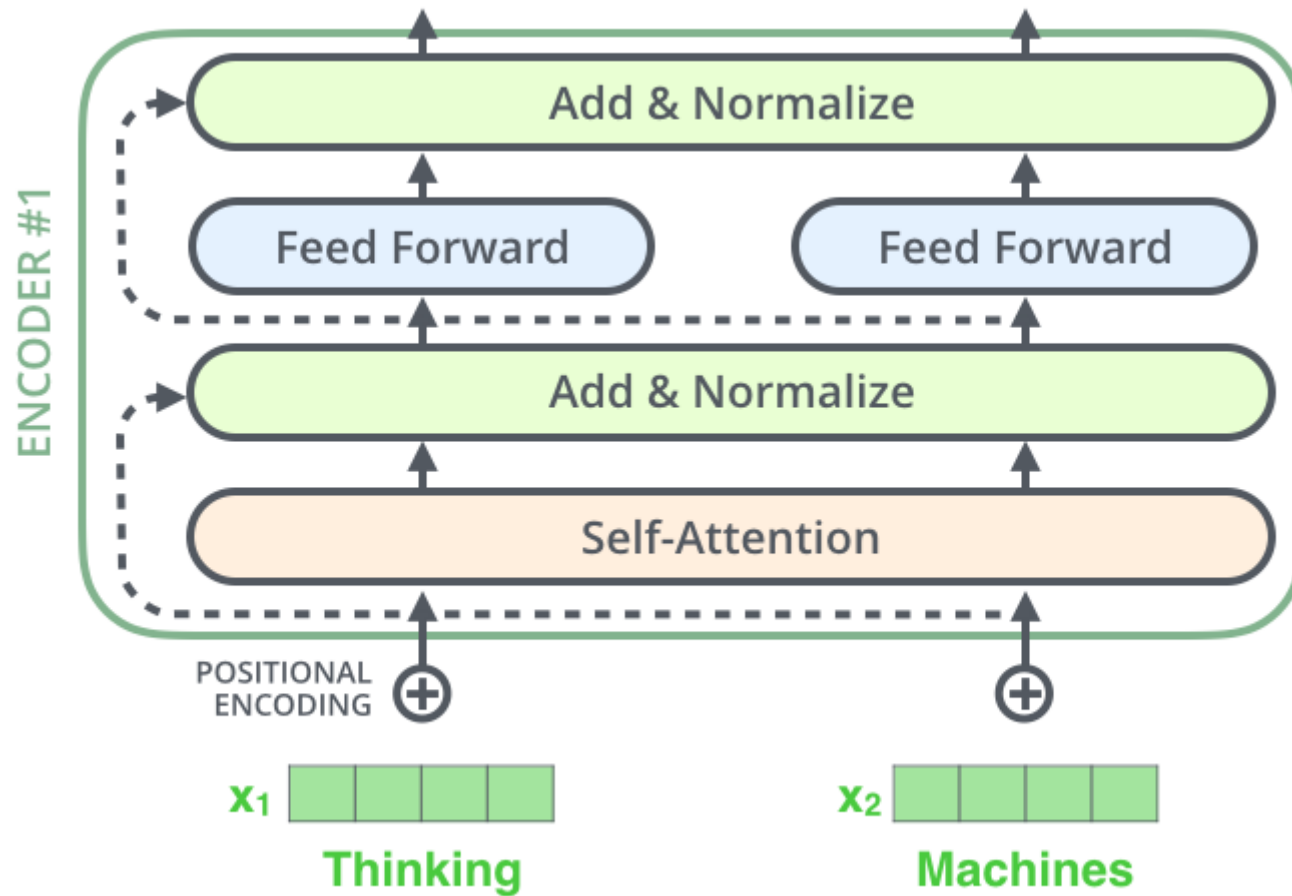


Z



* In all encoders other than #0, we don't need embedding. We start directly with the output of the encoder right below this one

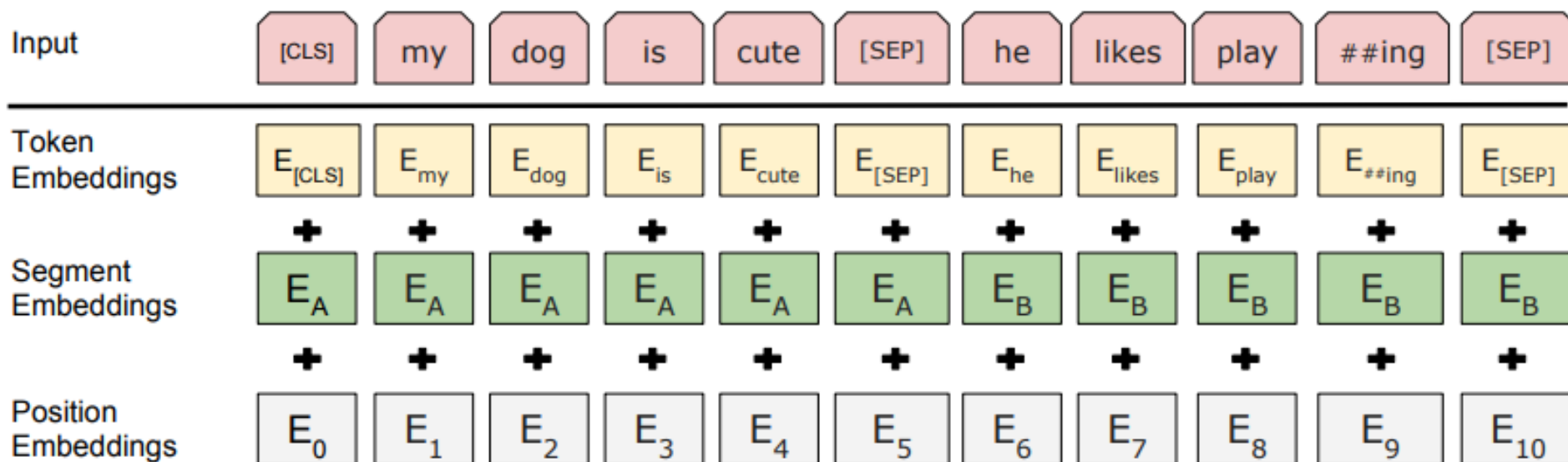
Блок трансформера



Параметры

- Обучили две модели – BERT base и BERT large
- BERT base: $L=12$, $H=768$, $A=12$, Parameters=110M
- BERT large: $L=24$, $H=1024$, $A=16$, Parameters=340M
- L – число слоёв
- H – размер скрытых представлений
- A – self-attention heads

Представление входа/выхода



- WordPiece эмбеддинги
- Токеты CLF, SEP
- Эмбеддинги сегмента и позиции

Предобучение на задаче Masked LM

- 15% слов маскируются
- Из них 80% заменяются на символ [MASK]
- 10% остаются неизменными
- 10% заполняются случайно

Masking Rates			Dev Set Results		
MASK	SAME	RND	MNLI Fine-tune	NER Fine-tune	NER Feature-based
80%	10%	10%	84.2	95.4	94.9
100%	0%	0%	84.3	94.9	94.0
80%	0%	20%	84.1	95.2	94.6
80%	20%	0%	84.4	95.2	94.7
0%	20%	80%	83.7	94.8	94.6
0%	0%	100%	83.6	94.9	94.6

Предобучение на задаче Next Sentence Prediction

Input = [CLS] the man went to [MASK] store [SEP]

he bought a gallon [MASK] milk [SEP]

Label = IsNext

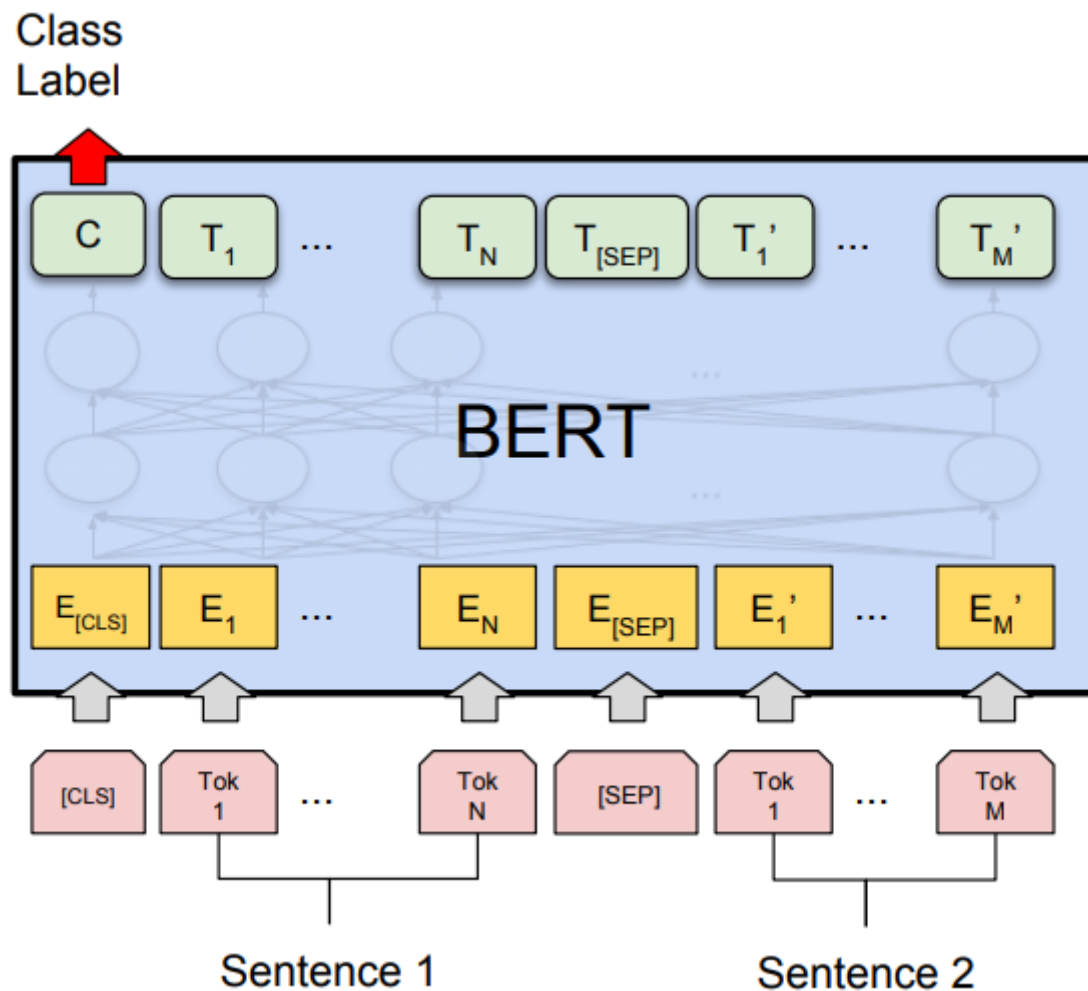
Input = [CLS] the man [MASK] to the store [SEP]

penguin [MASK] are flight ##less birds [SEP]

Label = NotNext

В 50% случаев предложения последовательные,
в 50% случайные

Предобучение на задаче Next Sentence Prediction



Данные

- BooksCorpus (800M слов), English Wikipedia (2,500M слов)
- Предложения - отрывки суммарной длиной ≤ 512

Эксперименты: бенчмарк GLUE

12 задач классификации:

- Multi-Genre Natural Language Inference (есть два предложения; согласуются, противоречат, или связаны нейтрально)
- Семантическая близость предложений
- Содержит ли предложение ответ на вопрос
- Тональность рецензии (положительная/отрицательная)
- Лингвистическая “правильность предложений”

Эксперименты: бенчмарк GLUE

System	MNLI-(m/mm) 392k	QQP 363k	QNLI 108k	SST-2 67k	CoLA 8.5k	STS-B 5.7k	MRPC 3.5k	RTE 2.5k	Average -
Pre-OpenAI SOTA	80.6/80.1	66.1	82.3	93.2	35.0	81.0	86.0	61.7	74.0
BiLSTM+ELMo+Attn	76.4/76.1	64.8	79.8	90.4	36.0	73.3	84.9	56.8	71.0
OpenAI GPT	82.1/81.4	70.3	87.4	91.3	45.4	80.0	82.3	56.0	75.1
BERT _{BASE}	84.6/83.4	71.2	90.5	93.5	52.1	85.8	88.9	66.4	79.6
BERT _{LARGE}	86.7/85.9	72.1	92.7	94.9	60.5	86.5	89.3	70.1	82.1

Эксперименты: задача SQuAD

Stanford Question Answering Dataset 1.1:

- Есть вопрос и отрывок текста. Нужно предсказать где начинается и заканчивается ответ.
- Вероятность того, что T_i начало = $S * T_i$, конец – $E * T_i$
- Берётся softmax и ответ – макс. $S * T_i + E * T_j$

SquAD 2.0:

- Допускается вариант, что ответа нет
- Вероятность этого $S * C + E * C$ (C – выход CLF)

Эксперименты: задача SQuAD

System	Dev		Test	
	EM	F1	EM	F1
Top Leaderboard Systems (Dec 10th, 2018)				
Human	-	-	82.3	91.2
#1 Ensemble - nlnet	-	-	86.0	91.7
#2 Ensemble - QANet	-	-	84.5	90.5
Published				
BiDAF+ELMo (Single)	-	85.6	-	85.8
R.M. Reader (Ensemble)	81.2	87.9	82.3	88.5
Ours				
BERT _{BASE} (Single)	80.8	88.5	-	-
BERT _{LARGE} (Single)	84.1	90.9	-	-
BERT _{LARGE} (Ensemble)	85.8	91.8	-	-
BERT _{LARGE} (Sgl.+TriviaQA)	84.2	91.1	85.1	91.8
BERT _{LARGE} (Ens.+TriviaQA)	86.2	92.2	87.4	93.2

SquAD v.1.1

System	Dev		Test	
	EM	F1	EM	F1
Top Leaderboard Systems (Dec 10th, 2018)				
Human	86.3	89.0	86.9	89.5
#1 Single - MIR-MRC (F-Net)	-	-	74.8	78.0
#2 Single - nlnet	-	-	74.2	77.1
Published				
unet (Ensemble)	-	-	71.4	74.9
SLQA+ (Single)	-	-	71.4	74.4
Ours				
BERT _{LARGE} (Single)	78.7	81.9	80.0	83.1

SquAD v.2.0

Эксперименты: задача SWAG

Situations With
Adversarial
Generations:

- Есть предложение и 4 варианта его продолжения. Нужно предсказать наиболее вероятный

System	Dev	Test
ESIM+GloVe	51.9	52.7
ESIM+ELMo	59.1	59.2
OpenAI GPT	-	78.0
BERT _{BASE}	81.6	-
BERT _{LARGE}	86.6	86.3
Human (expert) [†]	-	85.0
Human (5 annotations) [†]	-	88.0

Оценка значимости Masked LM и NSP

- Без NSP качество падает на задачах, где оценивается связь предложений
- Без двунаправленности качество резко падает на SQuAD

Tasks	Dev Set				
	MNLI-m (Acc)	QNLI (Acc)	MRPC (Acc)	SST-2 (Acc)	SQuAD (F1)
BERT _{BASE}	84.4	88.4	86.7	92.7	88.5
No NSP	83.9	84.9	86.5	92.6	87.9
LTR & No NSP	82.1	84.3	77.5	92.1	77.8
+ BiLSTM	82.1	84.1	75.7	91.6	84.9

Влияние размера модели на качество

- Качество строго увеличивается на всех задачах
- Качество сильно увеличивается даже на таких маленьких задачах, как MRPC

Hyperparams				Dev Set Accuracy		
#L	#H	#A	LM (ppl)	MNLI-m	MRPC	SST-2
3	768	12	5.84	77.9	79.8	88.4
6	768	3	5.24	80.6	82.2	90.7
6	768	12	4.68	81.9	84.8	91.3
12	768	12	3.99	84.4	86.7	92.9
12	1024	16	3.54	85.7	86.9	93.3
24	1024	16	3.23	86.6	87.8	93.7

Использование BERTa в feature based подходе

- Возможность использовать для задач, где трансформер справляется плохо
- Экономия вычислительных мощностей (можно переиспользовать полученные представления)

System	Dev F1	Test F1
ELMo (Peters et al., 2018a)	95.7	92.2
CVT (Clark et al., 2018)	-	92.6
CSE (Akbik et al., 2018)	-	93.1
Fine-tuning approach		
BERT _{LARGE}	96.6	92.8
BERT _{BASE}	96.4	92.4
Feature-based approach (BERT _{BASE})		
Embeddings	91.0	-
Second-to-Last Hidden	95.6	-
Last Hidden	94.9	-
Weighted Sum Last Four Hidden	95.9	-
Concat Last Four Hidden	96.1	-
Weighted Sum All 12 Layers	95.5	-

Результаты для задачи Named Entity Recognition



Эмбеддинги BERTa

- Образуют кластеры по смыслу
- Одно слово будет представлено по разному в зависимости от положения предложения
- Зависимость от контекста повышается на поздних слоях
- Эмбеддинги слов в предложении схожи на ранних слоях, расходятся на поздних
- Стоп-слова наиболее различаются в зависимости от контекста



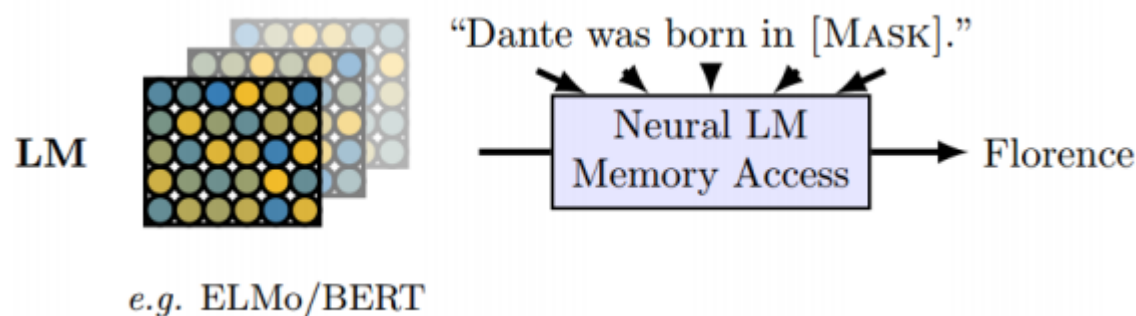
Анализ знаний, которые запоминает BERT

Способы:

- Анализ заполненных пропусков в Masked LM
- Анализ весов слоя self-attention
- Использование скрытых представлений в задачах классификации

Способность запоминать факты и аргументировать

- Хорошо запоминает связи между объектами



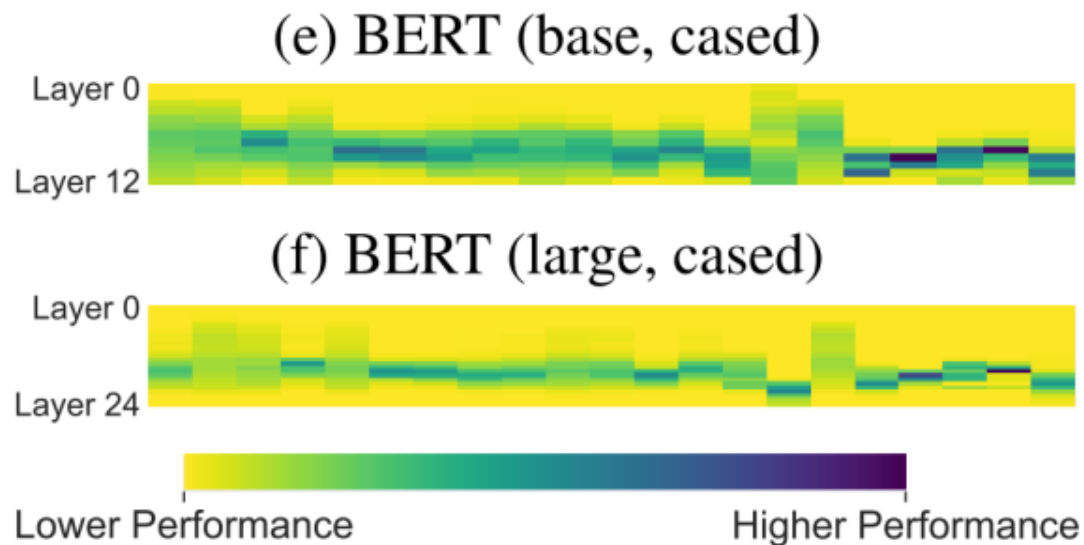
- Сложности с аргументацией:
- Может понять простые свойства объектов (напр. дом большой, человек может зайти в дом)
- Но не может понять как они взаимодействуют (напр. что дом больше человека)



Синтаксические и семантические знания

- Синтаксическая структура хранится иерархически
- Запоминаются части речи, части предложения
- Не чувствителен к перестановкам слов
- Семантические знания: запоминает типы объектов, их связи
- Плохо запоминает числа

Где хранятся знания



- Средние слои лучше работают для transfer learning
- Синтаксическая структура на средних слоях, семантическая на всех

Выводы

- Двунаправленная модель, основанная на трансформерах
- Предобучена на задачах Masked LM и Next Sentence Prediction
- Показывает SOTA результаты на большом числе задач
- Хорошо запоминает знания, но не умеет аргументировать