LLM intro

Attention is all we need

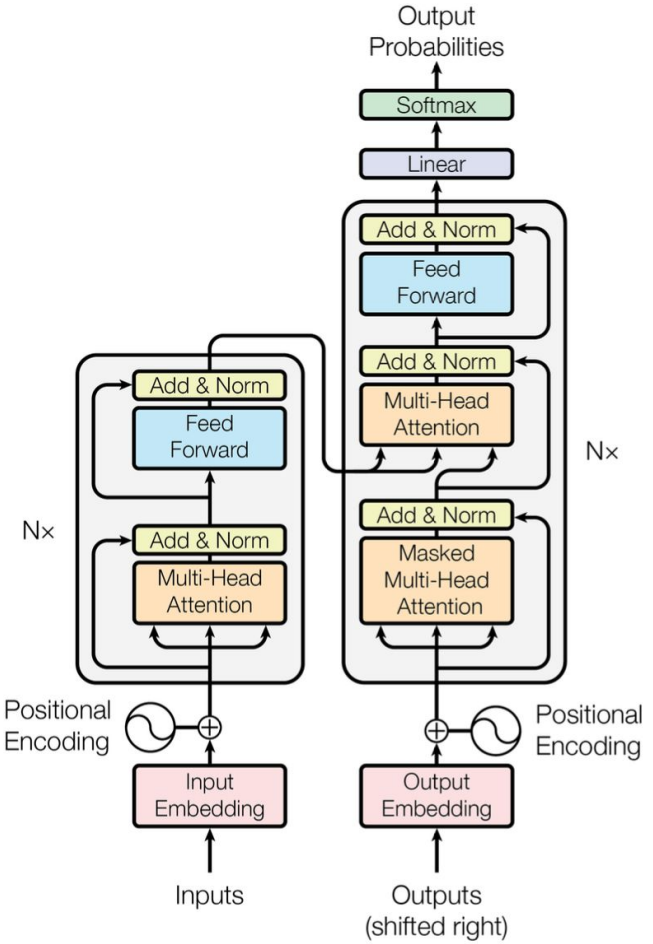


Figure 1: The Transformer - model architecture.

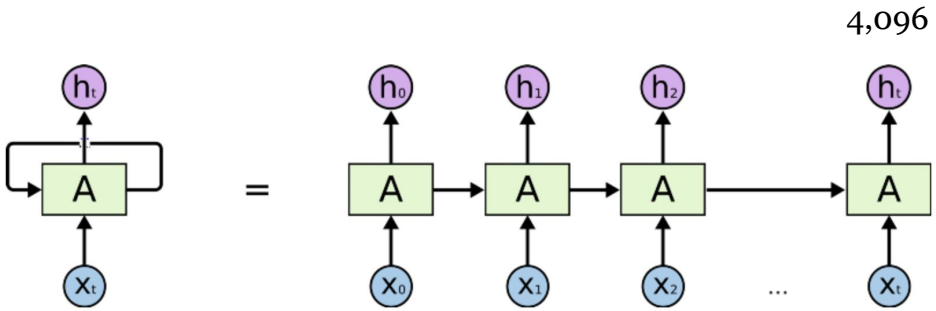
More data !

- Improved Generalization
- Enhanced Performance on Specific Tasks
- Few-Shot and Zero-Shot Learning

Unsupervised sentiment neuron

We first trained a multiplicative LSTM with 4,096 units on a corpus of 82 million Amazon reviews to predict the next character in a chunk of text. Training took one month across four NVIDIA Pascal GPUs, with our model processing 12,500 characters per second.

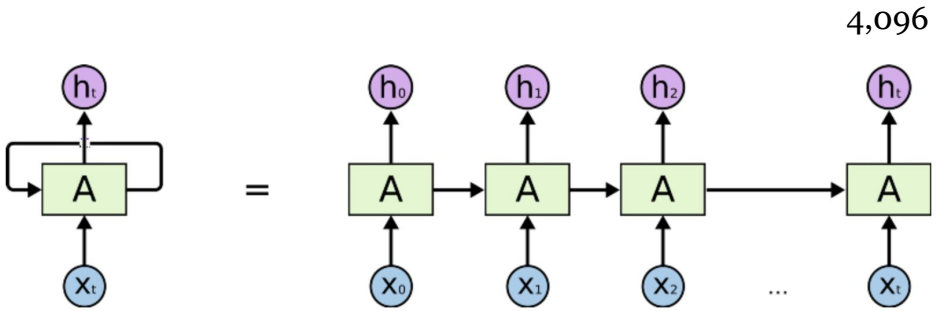
82 million Amazon reviews, 38GB
Не используем метки ! (Положительный отрицательный или сколько звезд)



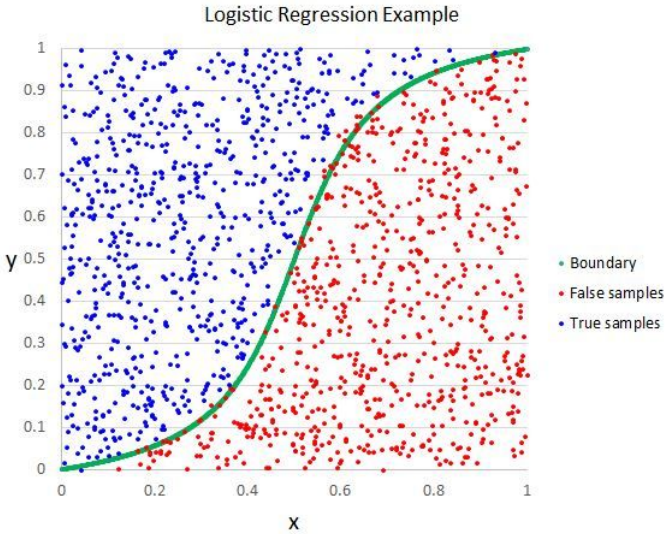
https://openai.com/research/unsupervised-sentiment-neuron

Unsupervised sentiment neuron

While training the linear model with L1 regularization, we noticed it used surprisingly few of the learned units. Digging in, we realized there actually existed a single “sentiment neuron” that’s highly predictive of the sentiment value.



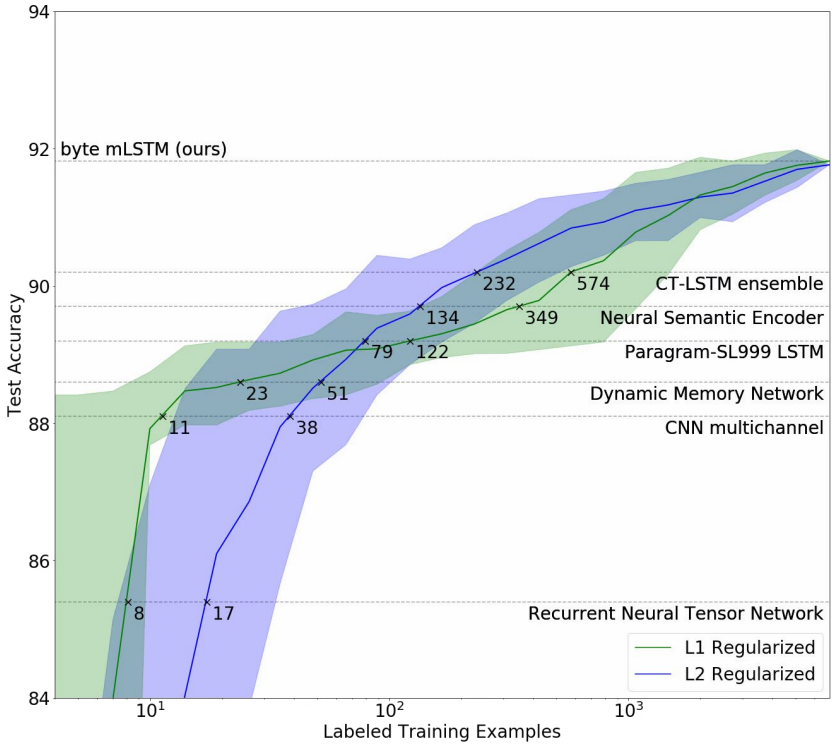
+



https://openai.com/research/unsupervised-sentiment-neuron

Unsupervised sentiment neuron

The number of labeled examples it takes two variants of our model (the green and blue lines) to match fully supervised approaches, each trained with 6,920 examples (the dashed gray lines). Our L1-regularized model (pretrained in an unsupervised fashion on Amazon reviews) matches [multichannel CNN](https://arxiv.org/abs/1408.5882) performance with only 11 labeled examples, and state-of-the-art CT-LSTM Ensembles with 232 examples.



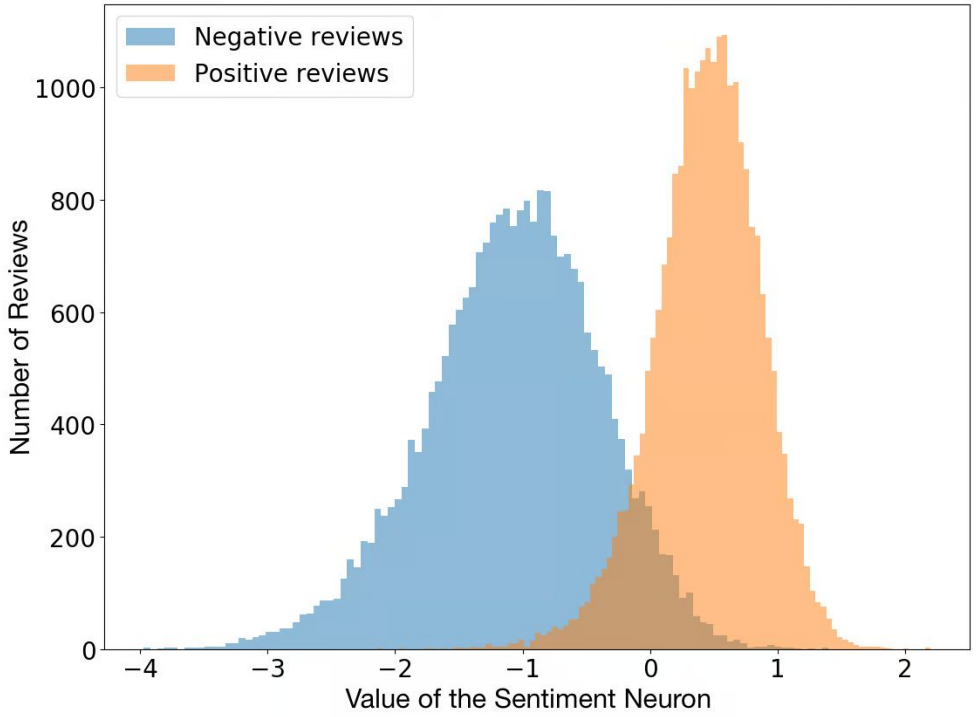
stanford sentiment treebank

https://openai.com/research/unsupervised-sentiment-neuron

Unsupervised sentiment neuron

Just like with similar models, our model can be used to generate text. Unlike those models, we have a direct dial to control the sentiment of the resulting text: we simply overwrite the value of the sentiment neuron.

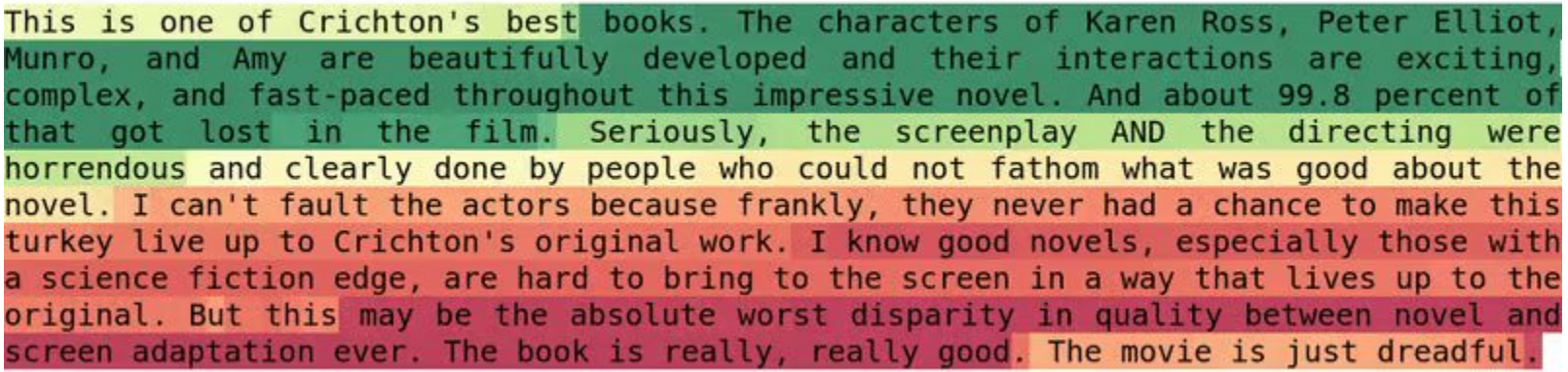
Sentiment fixed to positive	Sentiment fixed to negative
Just what I was looking for. Nice fitted pants, exactly matched seam to color contrast with other pants I own. Highly recommended and also very happy!	The package received was blank and has no barcode. A waste of time and money.
This product does what it is supposed to. I always keep three of these in my kitchen just in case ever I need a replacement cord.	Great little item. Hard to put on the crib without some kind of embellishment. My guess is just like the screw kind of attachment I had.
Best hammock ever! Stays in place and holds it's shape. Comfy (I love the deep neon pictures on it), and looks so cute.	They didn't fit either. Straight high sticks at the end. On par with other buds I have. Lesson learned to avoid.



Распределение активации одного нейрона
IMDB reviews

https://openai.com/research/unsupervised-sentiment-neuron

Unsupervised sentiment neuron



The sentiment neuron adjusting its value on a character-by-character basis.

ЭТО НЕ РАБОТАЕТ ДЛЯ МАЛЕНЬКИХ МОДЕЛЕЙ

https://openai.com/research/unsupervised-sentiment-neuron

Attention is all we need

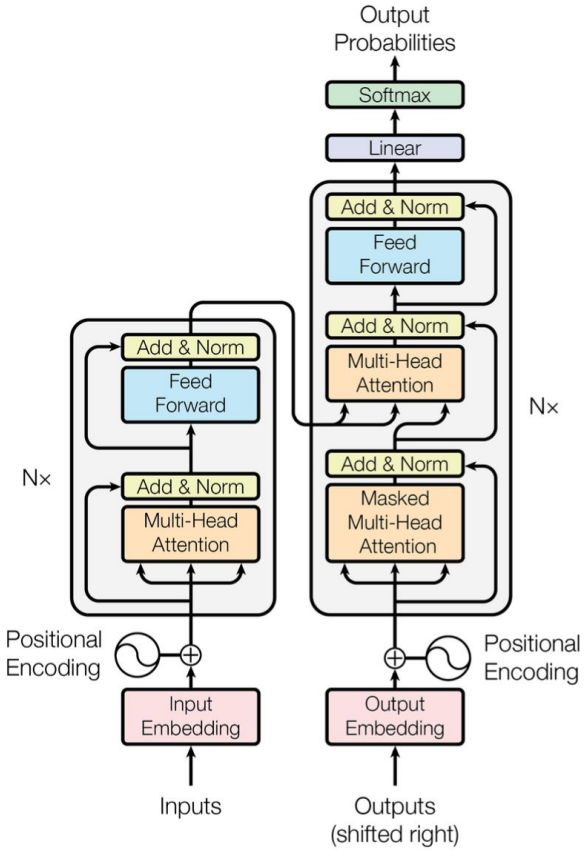


Figure 1: The Transformer - model architecture.

Attention is all we need

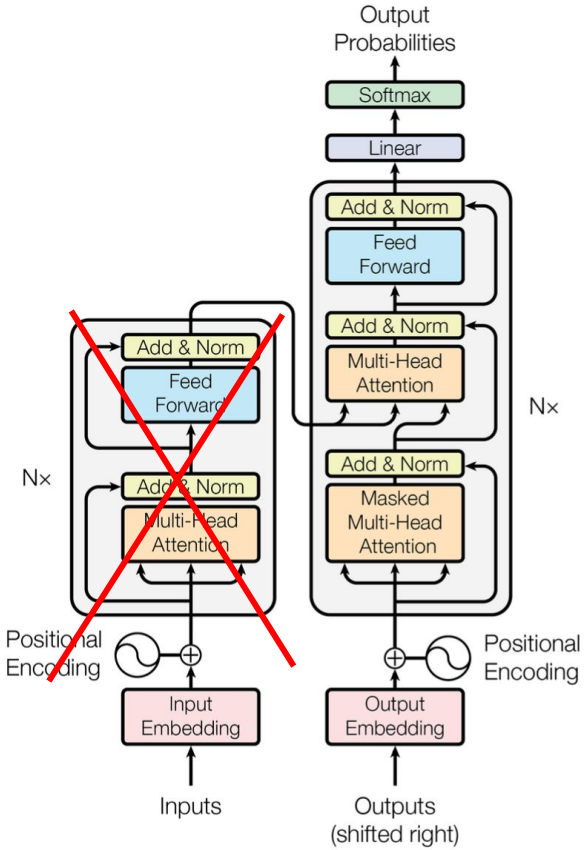


Figure 1: The Transformer - model architecture.

Attention is all we need

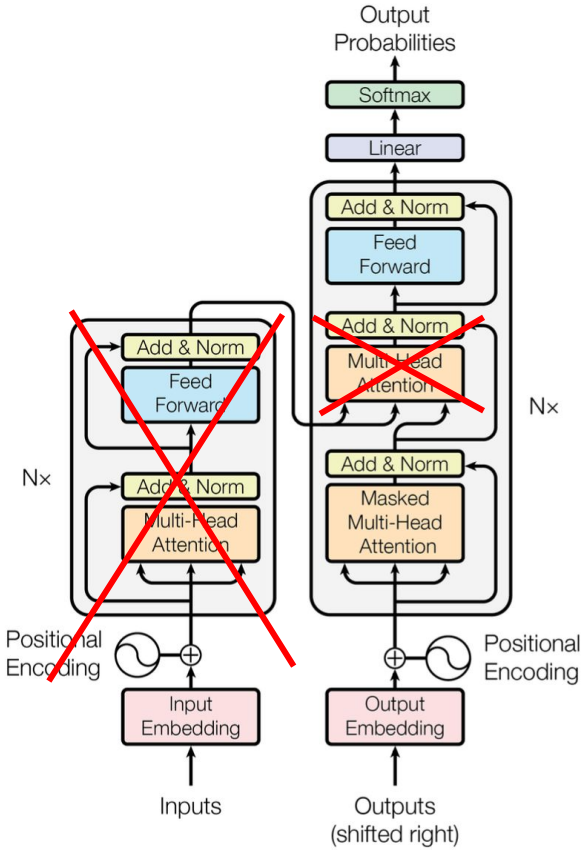
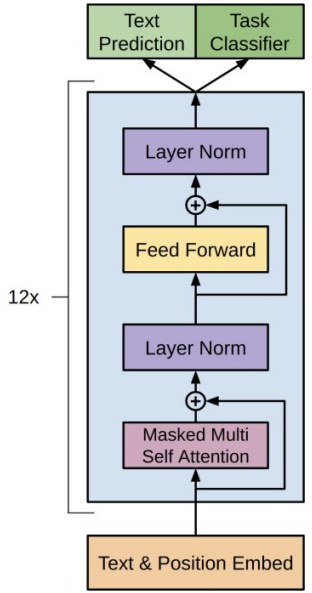


Figure 1: The Transformer - model architecture.

Attention is all we need



GPT-1 Improving Language Understanding by Generative Pre-Training

Scores
(before softmax)

0.11	0.00	0.81	0.79
0.19	0.50	0.30	0.48
0.53	0.98	0.95	0.14
0.81	0.86	0.38	0.90

Apply Attention Mask →

Masked Scores
(before softmax)

0.11	-inf	-inf	-inf
0.19	0.50	-inf	-inf
0.53	0.98	0.95	-inf
0.81	0.86	0.38	0.90

```
import torch

softmax = torch.nn.Softmax()
input = torch.Tensor([1,0,0,0,0])
print(input)

output = softmax(input)
print(output)
```

```
tensor([1., 0., 0., 0., 0.])
tensor([0.4046, 0.1488, 0.1488, 0.1488, 0.1488])
```

Masked Scores
(before softmax)

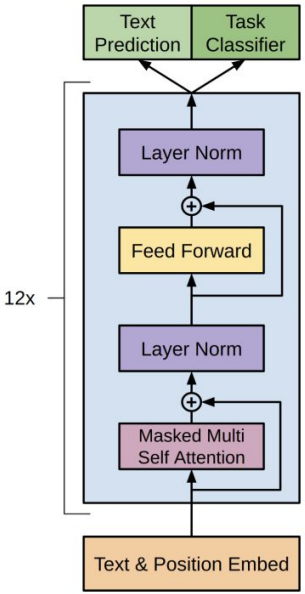
0.11	-inf	-inf	-inf
0.19	0.50	-inf	-inf
0.53	0.98	0.95	-inf
0.81	0.86	0.38	0.90

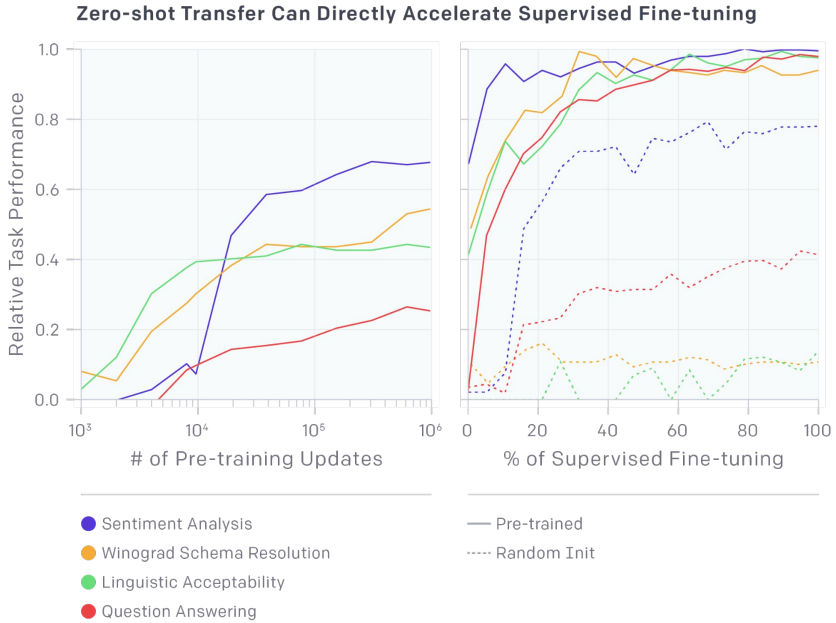
Softmax
(along rows) →

Scores

1	0	0	0
0.48	0.52	0	0
0.31	0.35	0.34	0
0.25	0.26	0.23	0.26

GPT-1 Improving Language Understanding by Generative Pre-Training





https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf
https://openai.com/research/language-unsupervised
https://sannaperzon.medium.com/paper-summary-gpt-1-improving-language-understanding-by-generative-pre-training-c43bd7ff242a

GPT-1 Данные

Данные и вычисления
Токенизация: BPE с размером словаря 40k
Корпус: BooksCorpus, 7000 неопубликованных книг (~5GB)
Мощности: 8 x Nvidia P6000 x 30 дней x 33% утилизация (~1 pfs-day)
Размер: 12 блоков (~117M параметров), 512 токенов контекста

GPT-1 fine-tune

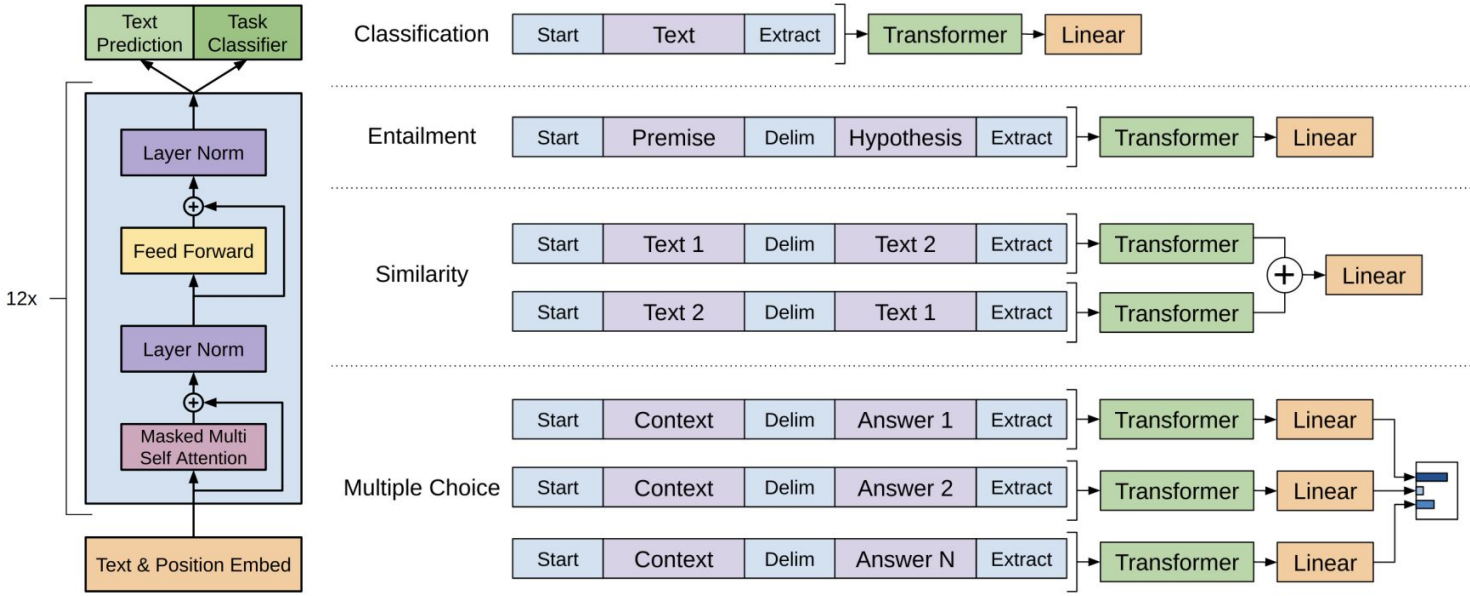
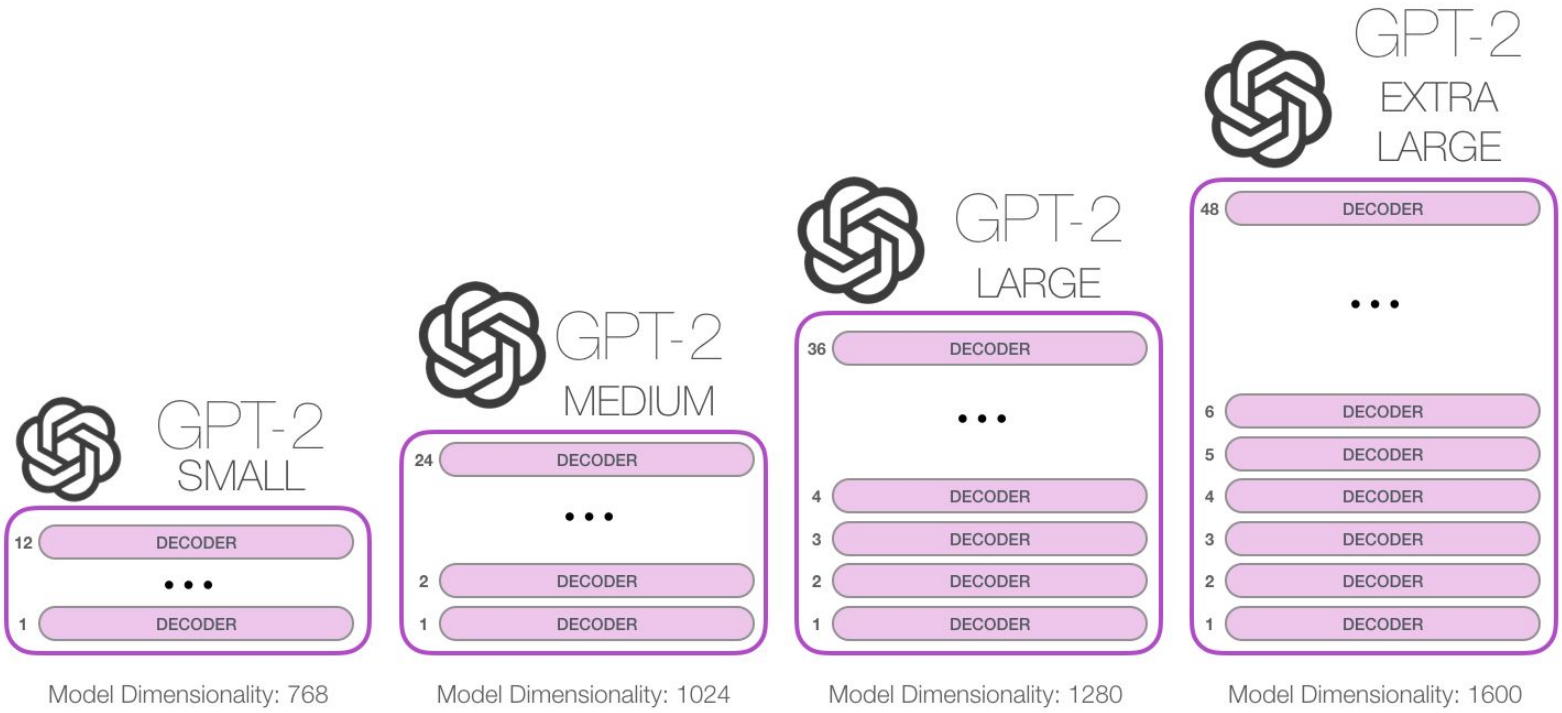


Figure 1: **(left)** Transformer architecture and training objectives used in this work. **(right)** Input transformations for fine-tuning on different tasks. We convert all structured inputs into token sequences to be processed by our pre-trained model, followed by a linear+softmax layer.

GPT-1 Метрики

Dataset	Task	SOTA	Ours
SNLI	Textual entailment	89.3	89.9
MNLI matched	Textual entailment	80.6	82.1
MNLI mismatched	Textual entailment	80.1	81.4
SciTail	Textual entailment	83.3	88.3
QNLI	Textual entailment	82.3	88.1
RTE	Textual entailment	61.7	56.0
STS-B	Semantic similarity	81.0	82.0
QQP	Semantic similarity	66.1	70.3
MRPC	Semantic similarity	86.0	82.3
RACE	Reading comprehension	53.3	59.0
ROCStories	Commonsense reasoning	77.6	86.5
COPA	Commonsense reasoning	71.2	78.6
SST-2	Sentiment analysis	93.2	91.3
CoLA	Linguistic acceptability	35.0	45.4
GLUE	Multi task benchmark	68.9	72.8

GPT-2



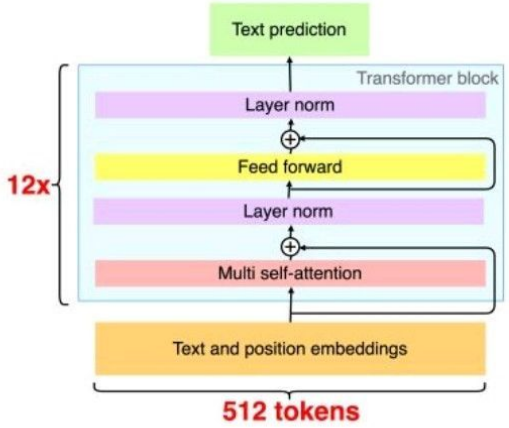
https://jalammar.github.io/illustrated-gpt2/

GPT-2 Данные

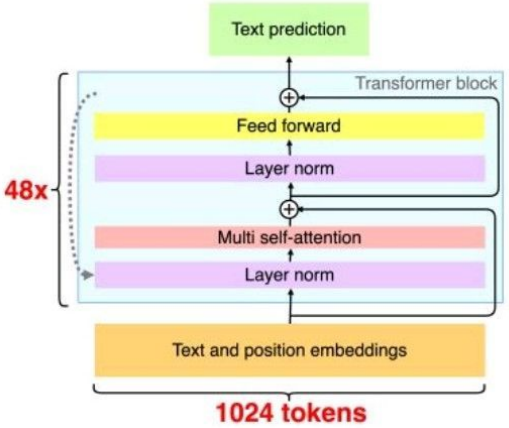
Данные и вычисления
Токенизация: Based on byte-level Byte-Pair-Encoding BBPE, 50257
Датасет: 45M ссылок с reddit (8M документов 40GB текста)
Мощности: 100 x Nvidia Volta GPU x 1 неделю

GPT-1 vs GPT-2 vs GPT-3

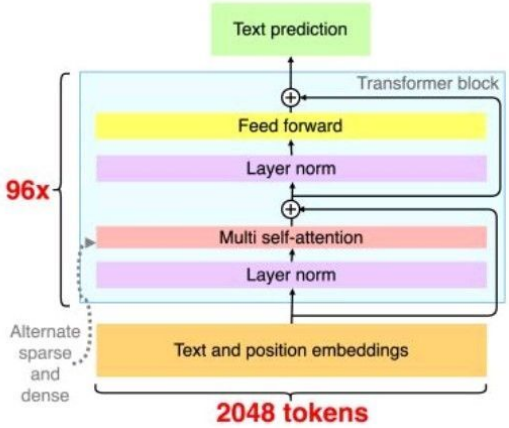
GPT-1



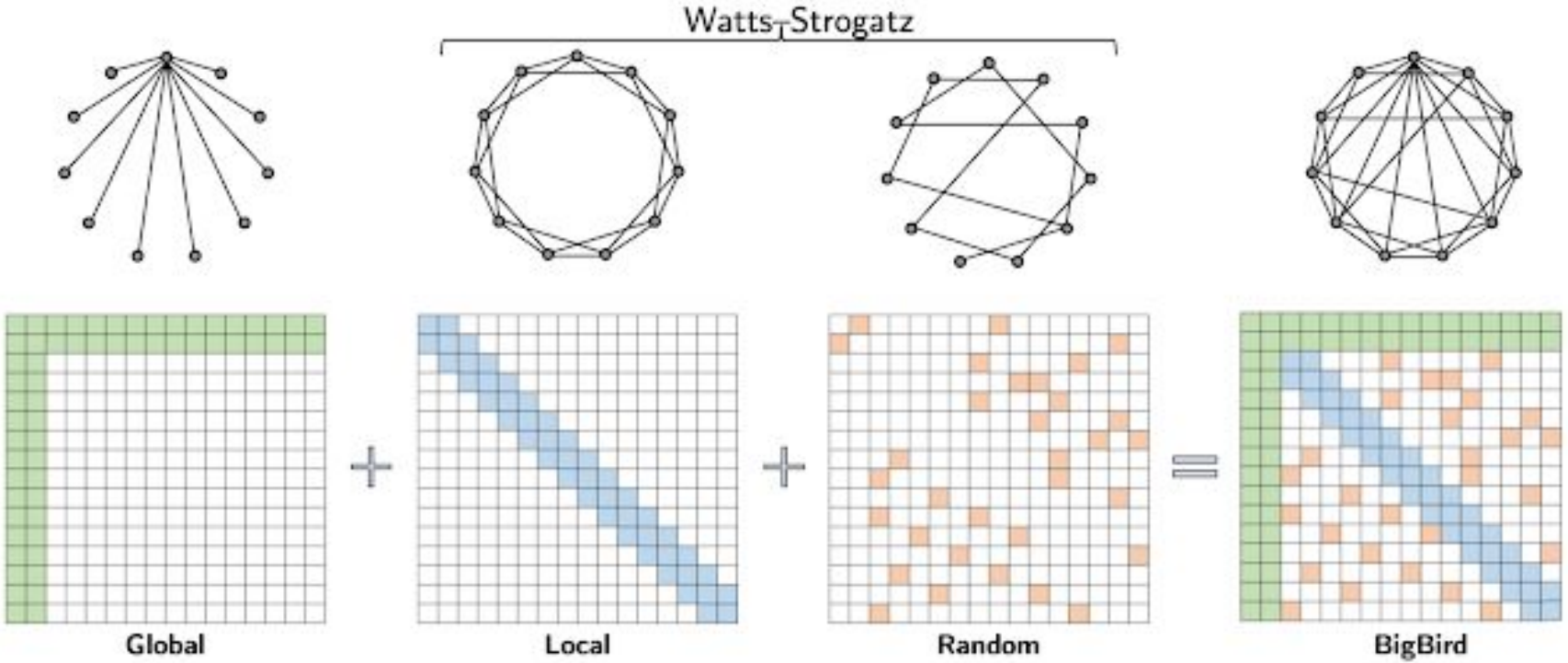
GPT-2



GPT-3



Attention



https://blog.research.google/2021/03/constructing-transformers-for-longer.html

Attention is all we need

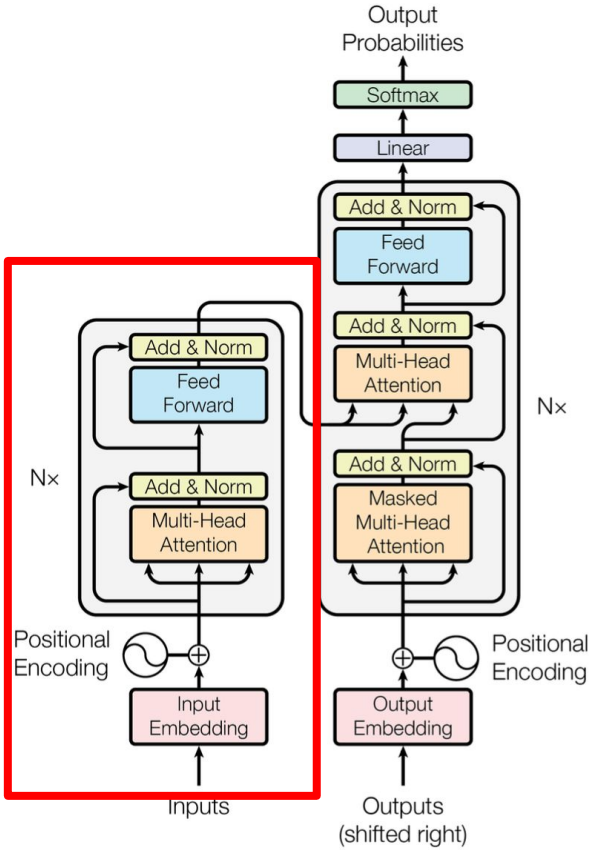
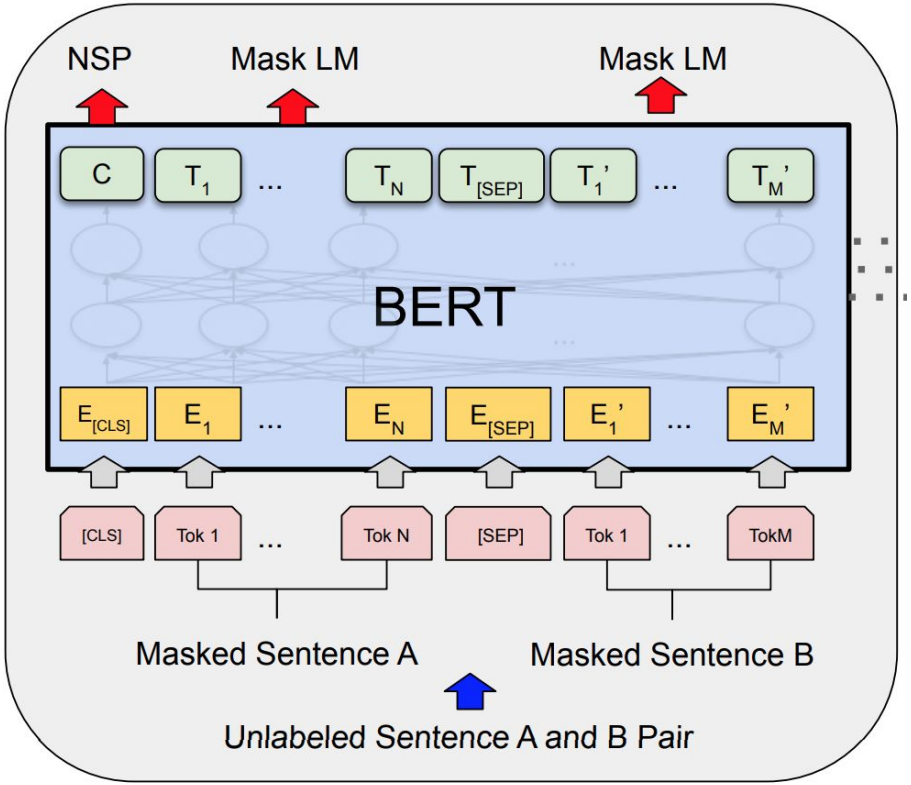


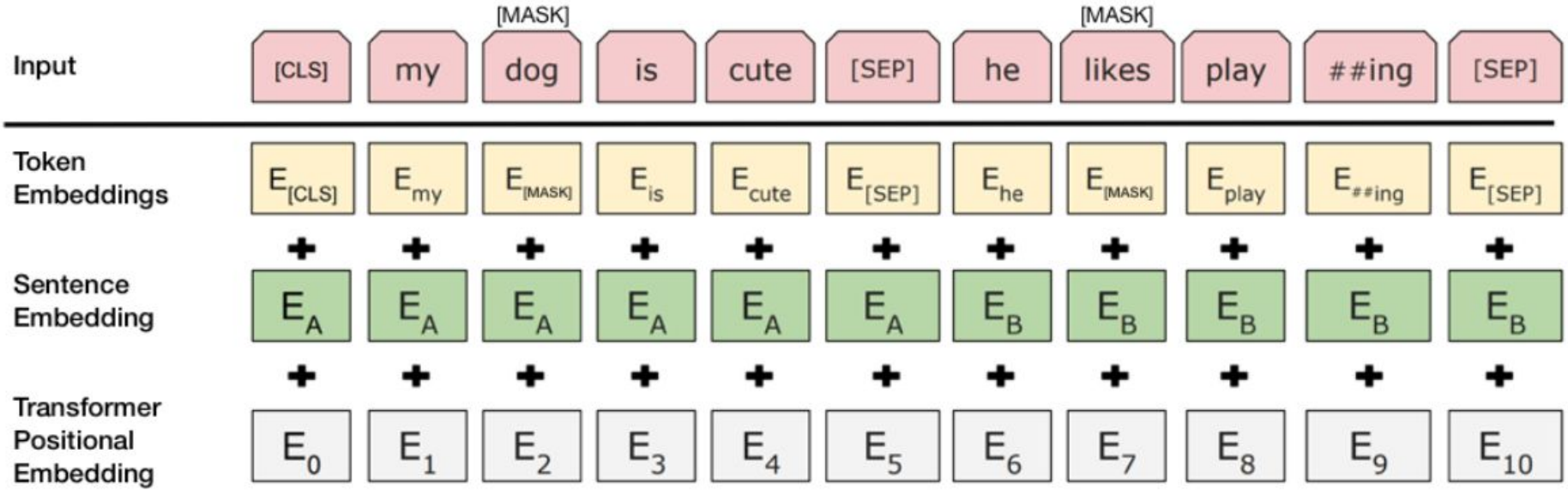
Figure 1: The Transformer - model architecture.

BERT Pre-training



NSP - next sentence prediction
Mask LM - Masked Language Model

BERT embeding



BERT for text generation ?

Начальное предложение предложение

`although he had already eaten a large meal, he was still very hungry.`

Маска

`although he had already eaten a large meal, he was still very [MASK].`

https://mayhewsw.github.io/2019/01/16/can-bert-generate-text/