Machine Learning (CE 40717)

Fall 2024

Ali Sharifi-Zarchi

CE Department
Sharif University of Technology

November 13, 2024

Attention is all you need!

Attention Is All You Need

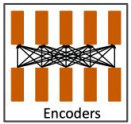
Ashish Vaswani* Google Brain avaswani@google.com
Noam Shazeer* Google Brain noam@google.com
Niki Parmar* Google Research nikip@google.com
Jakob Uszkoreit* Google Research usz@google.com
Llion Jones* Google Research llion@google.com
Aidan N. Gomez* † University of Toronto aidan@cs.toronto.edu
Łukasz Kaiser* Google Brain lukaszkaiser@google.com
Illia Polosukhin* ‡ illia.polosukhin@gmail.com

Abstract

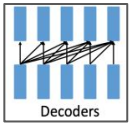
The dominant sequence transduction models are based on complex recurrent or convolutional neural networks that include an encoder and a decoder. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to be superior in quality while being more parallelizable and requiring significantly less time to train. Our model achieves 28.4 BLEU on the WMT 2014 English-to-German translation task, improving over the existing best results, including ensembles, by over 2 BLEU. On the WMT 2014 English-to-French translation task, our model establishes a new single-model state-of-the-art BLEU score of 41.0 after training for 3.5 days on eight GPUs, a small fraction of the training costs of the best models from the literature.

https://arxiv.org/abs/1706.03762

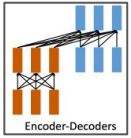
Transformer Architectures



Encoder-only (e.g., BERT): bidirectional contextual embeddings



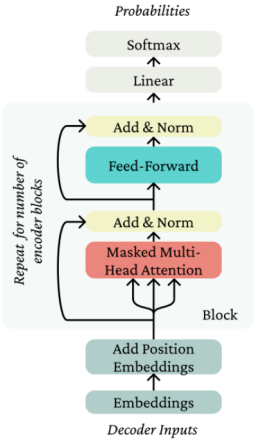
Decoder-only (e.g., GPT-x): unidirectional contextual embeddings, generate one token at a time



Encoder-decoder (e.g., T5): encode input, decode output

Transformer block

- Each block has two "sublayers"
  1. Multihead attention
  2. Feed-forward NNet (with ReLU)
- Residual: x + Sublayer(x)
- Layernorm changes input to have mean 0 and variance 1



https://web.stanford.edu/class/cs224n/slides/cs224n-2024-lecture19-open-problems.pdf

Layer normalization

Main Idea: Batch normalization is advantageous for stability but presents challenges with sequences of varying lengths.
Result: It provides a more stable input for the next layer.
Solution: "Layer normalization" functions similarly to batch normalization but doesn't normalize across the entire batch.

Batch norm

$$\mu = \frac{1}{B}\sum_{i=1}^{B} a_i, \quad \sigma = \sqrt{\frac{1}{B}\sum_{i=1}^{B}(a_i - \mu)^2}$$

$$\bar{a}_i = \frac{a_i - \mu}{\sigma}$$

d-dimensional vectors for each sample in batch

Layer norm

$$\mu = \frac{1}{d}\sum_{i=1}^{d} a_j, \quad \sigma = \sqrt{\frac{1}{d}\sum_{i=1}^{d}(a_j - \mu)^2}$$

$$\bar{a} = \frac{a - \mu}{\sigma}$$

Different dimensions of a

Why transformers?

Pros:

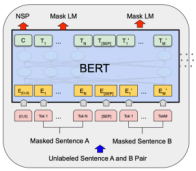
+ Much easier to parallelize
+ Much better long-range connections
+ In practice, can make it much deeper (more layers) than RNN

Cons:

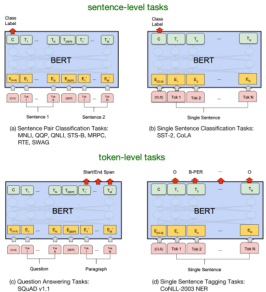
- Attention computations are technically $O(n^2)$
- Somewhat more complex to implement (positional encodings, etc.)

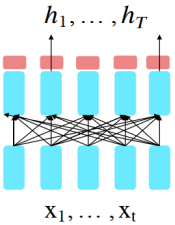
Encoder Language Model

$$P(x) = \prod_{i=1}^{n} P(x_i \mid x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n)$$



Pre-training





$$h_1, \ldots, h_t = \text{Decoder}(x_1, \ldots, x_t)$$

$$x_{\text{mask}} \sim A h_{\text{masked}} + b$$

https://arxiv.org/abs/1810.04805

Encoder Language Model

Encoder language models, like BERT, use masked tokens to learn bidirectional representations of text.

- **Masked Language Modeling:** Predicts randomly masked tokens in a sequence.
- **Bidirectional Context:** Considers information from both directions for each token.
- **Applications:** Used for classification, NER, and other NLP tasks.

BERT: Key Contributions

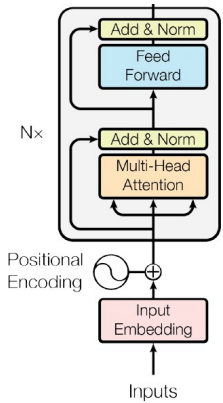
- It is a fine-tuning approach based on a deep Transformer encoder.
- The key: learn representations based on **bidirectional context**

  Why? Because both left and right contexts are important to understand the meaning of words.

  Example #1: we went to the river bank.
  Example #2: I need to go to bank to make a deposit.

- **Pre-training objectives:** masked language modeling + next sentence prediction
- State-of-the-art performance on a large set of **sentence-level** and **token-level** tasks

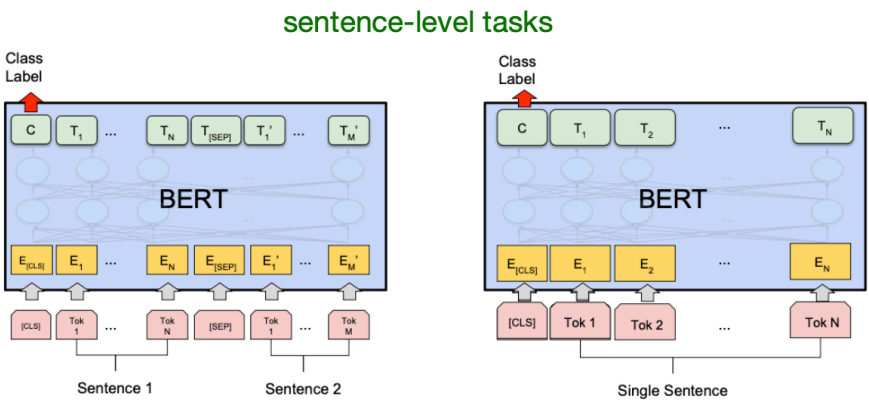
BERT pre-training: putting together



- **BERT-base:** 12 layers, 768 hidden size, 12 attention heads, 110M parameters
- **BERT-large:** 24 layers, 1024 hidden size, 16 attention heads, 340M parameters
- **Training corpus:** Wikipedia (2.5B) + BooksCorpus (0.8B)
- **Max sequence size:** 512 word pieces (roughly 256 and 256 for two non-contiguous sequences)
- **Trained for:** 1M steps, batch size 128k

https://arxiv.org/abs/1706.03762

Sentence-level tasks



https://arxiv.org/abs/1810.04805

Sentence-level tasks(cont.)

- Sentence pair classification tasks:

  MNLI

  - **Premise:** A soccer game with multiple males playing.
  - **Hypothesis:** Some men are playing a sport.
  - Result: {entailment, contradiction, neutral}

  QQP

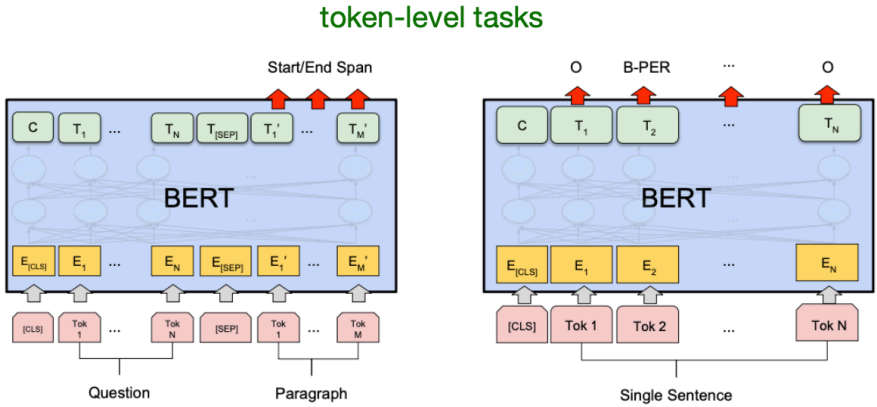
  - Q1: Where can I learn to invest in stocks?
  - Q2: How can I learn more about stocks?
  - Result: {duplicate, not duplicate}

- Single sentence classification tasks:

  SST2

  - Sentence: rich veins of funny stuff in this movie
  - Result: {positive, negative}

Token-level tasks



(c) Question Answering Tasks: SQuAD v1.1

(d) Single Sentence Tagging Tasks: CoNLL-2003 NER

https://arxiv.org/abs/1810.04805

Token-level tasks: Extractive Question Answering

- Extractive question answering e.g., SQuAD (Rajpurkar et al., 2016)

SQuAD

Question: The New York Giants and the New York Jets play at which stadium in NYC ?

Context: The city is represented in the National Football League by the New York Giants and the New York Jets , although both teams play their home games at MetLife Stadium in nearby East Rutherford , New Jersey , which hosted Super Bowl XLVIII in 2014 .

(Training example 29,883)

Result: MetLife Stadium

Token-level tasks: Named Entity Recognition

Token-level tasks

- Named entity recognition (Tjong Kim Sang and De Meulder, 2003)

CoNLL 2003 NER

John	Smith	lives	in	New	York
B-PER	I-PER	O	O	B-LOC	I-LOC

Masked Language Modeling (MLM)

- **Q:** Why we can't do language modeling with bidirectional models?

- **Solution:** Mask out k percent of the input words, and then predict the masked words.

store **gallon**

↓ ↓

the man went to [*MASK*] to buy a [*MASK*] of milk

MLM: Masking Rate and Strategy

- **Q: What is the value of k?**
  - They always use $k = 15\%$.
  - Too little masking: computationally expensive (we need to increase # of epochs)
  - Too much masking: not enough context
  - See (Wettig et al., 2022) for more discussion of masking rates:
    - Masking 40% outperforms 15% for BERT-large size models on GLUE and SQuAD
    - High masking rate of 80% can still preserve 95% fine-tuning performance

- **Q: How are masked tokens selected?**
  - 15% tokens are uniformly sampled
  - Is it optimal? See span masking (Joshi et al., 2020) and PMI masking (Levine et al., 2021)

Example: He **[MASK]** from Kuala **[MASK]**, Malaysia.

Next Sentence Prediction (NSP)

- Motivation: many NLP downstream tasks require understanding the relationship between two sentences (natural language inference, paraphrase detection, QA).
- NSP is designed to reduce the gap between pre-training and fine-tuning.

[CLS]: a special token always at the beginning

[SEP]: a special token used to separate two segments

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`

They sample two contiguous segments for 50% of the time and another random segment from the corpus for 50% of the time

BERT Training

Dataset. Let $\mathscr{D}$ be a set of examples $(x_{1:L}, c)$ constructed as follows:

- Let A be a sentence from the corpus.
- With probability 0.5, let B be the next sentence.
- With probability 0.5, let B be a random sentence from the corpus.
- Let $x_{1:L} = [\text{CLS}], A, [\text{SEP}], B$.
- Let c denote whether B is the next sentence or not.

Objective. Then the BERT objective is:

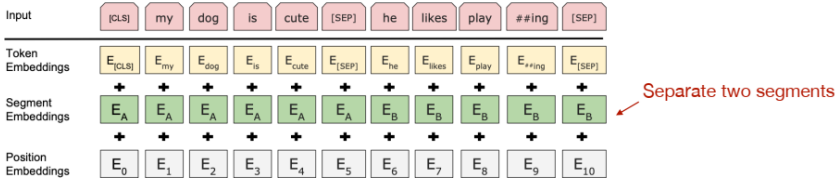
$$\mathcal{O}(\theta) = \sum_{(x_{1:L},c)\in\mathcal{D}} \underbrace{\mathbb{E}_{I,\tilde{x}_{1:L}\sim A(\cdot|x_{1:L},I)}\left[\sum_{i\in I} -\log p_\theta(\tilde{x}_i \mid x_{1:L})\right]}_{\text{masked language modeling}} + \underbrace{-\log p(c \mid \phi(x_{1:L})_1)}_{\text{next sentence prediction}}.$$

BERT Pre-training: Putting Together

- **Vocabulary size:** 30,000 wordpieces (common sub-word units) (Wu et al., 2016)

(Image: Stanford CS224N)

- **Input embeddings:**

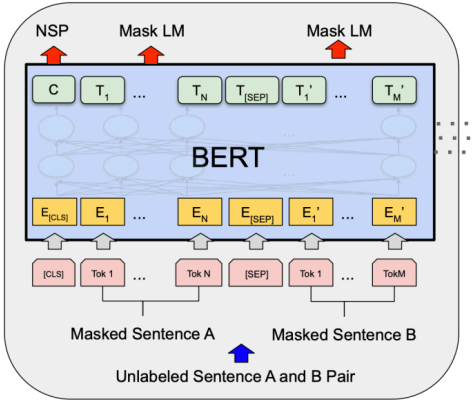


- Just two possible "segment embeddings": EA and EB.
- Positional embeddings are learned vectors for every possible position between 0 and 512-1.

https://cs330.stanford.edu/fall2019/presentations/presentation-10.23-1.pdf

BERT Pre-training: Putting Together

- MLM and NSP are trained together
- `[CLS]` is pre-trained for NSP
- Other token representations are trained for MLM



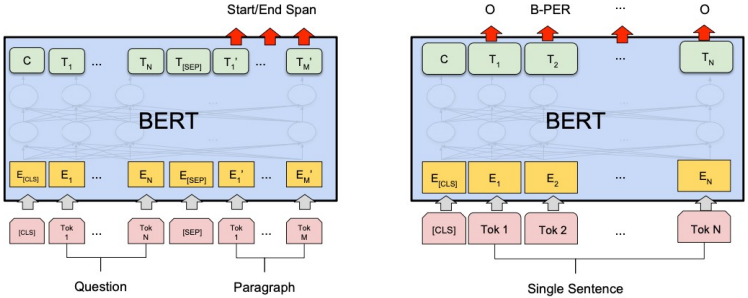
https://arxiv.org/abs/1810.04805

Figure 2: *

Fine-tuning BERT

“Pretrain once, finetune many times.”

token-level tasks



(c) Question Answering Tasks: SQuAD v1.1

(d) Single Sentence Tagging Tasks: CoNLL-2003 NER

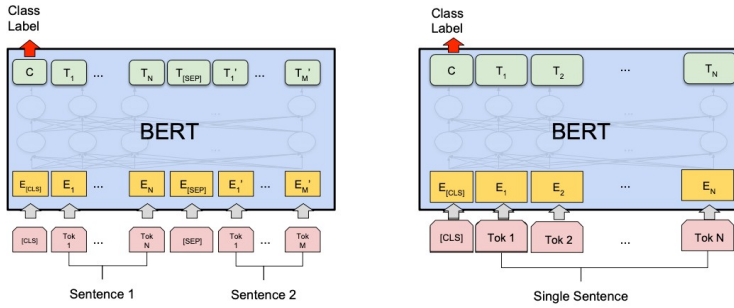
For token-level prediction tasks, add linear classifier on top of hidden representations

Q: How many new parameters?

Fine-tuning BERT

“Pretrain once, finetune many times.”

sentence-level tasks

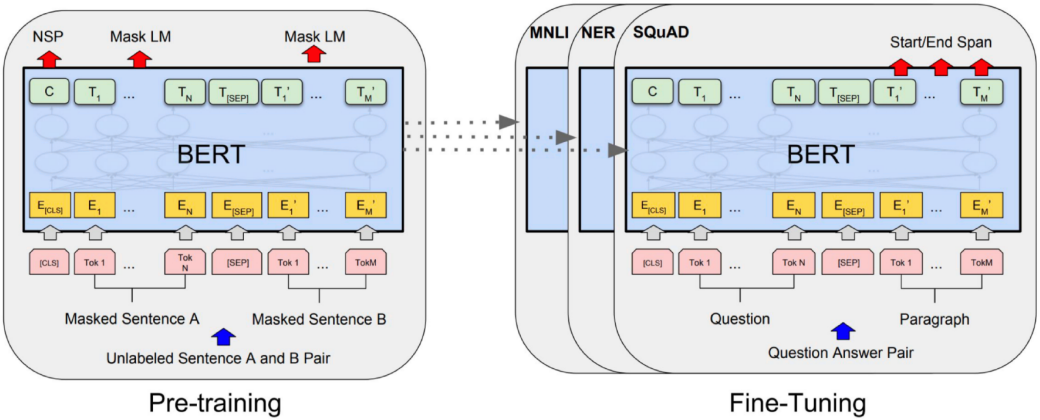


(a) Sentence Pair Classification Tasks: MNLI, QQP, QNLI, STS-B, MRPC, RTE, SWAG

(b) Single Sentence Classification Tasks: SST-2, CoLA

For sentence pair tasks, use [SEP] to separate the two segments with segment embeddings and add a linear classifier on top of [CLS] representation.

Finetuning Paradigm in NLP



BERT Extensions

- Models that handle long contexts (> 512 tokens)
  - Longformer, Big Bird, ...
- Multilingual BERT
  - Trained single model on 104 languages from Wikipedia. Shared 110k WordPiece vocabulary
- BERT extended to different domains
  - SciBERT, BioBERT, FinBERT, ClinicalBERT, ...
- Making BERT smaller to use
  - DistillBERT, TinyBERT, ...

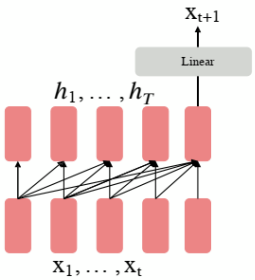
BERT Extensions

- **RoBERTa** (Liu et al., 2019)
  - Trained on 10x data & longer, no NSP
  - Much stronger performance than BERT (e.g., 94.6 vs 90.9 on SQuAD)
  - Still one of the most popular models to date
- **ALBERT** (Lan et al., 2020)
  - Increasing model sizes by sharing model parameters across layers
  - Less storage, much stronger performance but runs slower

Decoder Language Model

Autoregressive (AR) models use decoder stacks in generation, aiming to maximize log-likelihood via forward autoregressive factorization:

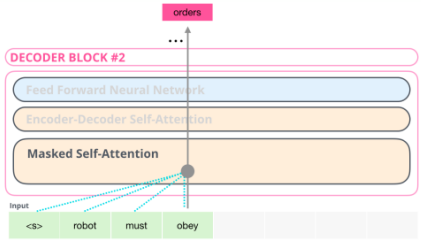
$$\max_{\theta} \log p_\theta(x_1, \ldots, x_T) \approx \sum_{t=1}^{T} \log p_\theta(x_t \mid x_1, \ldots, x_{t-1})$$

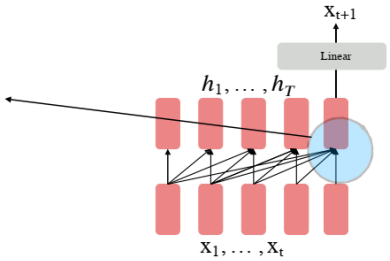


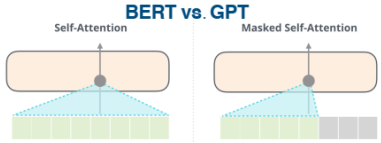
$$h_1, \ldots, h_t = \text{Decoder}(x_1, \ldots, x_t)$$
$$x_{t+1} \sim A h_t + b$$

Decoder Language Model

$$\max_{\theta} \log p_\theta(x_1, \ldots, x_T) \approx \sum_{t=1}^{T} \log p_\theta(x_t \mid x_1, \ldots, x_{t-1})$$





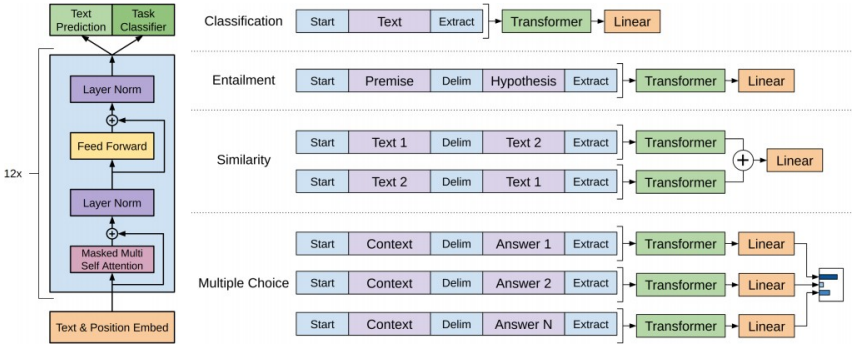


$$h_1, \ldots, h_t = \text{Decoder}(x_1, \ldots, x_t)$$
$$x_{t+1} \sim A h_t + b$$

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

Generative Pre-Trained Transformer (GPT)

- Transformer decoder with 12 layers.
- Byte-pair encoding with 40,000 merges.
- Trained on BooksCorpus: over 7000 unique books.
  - Contains long spans of contiguous text, for learning long-distance dependencies.

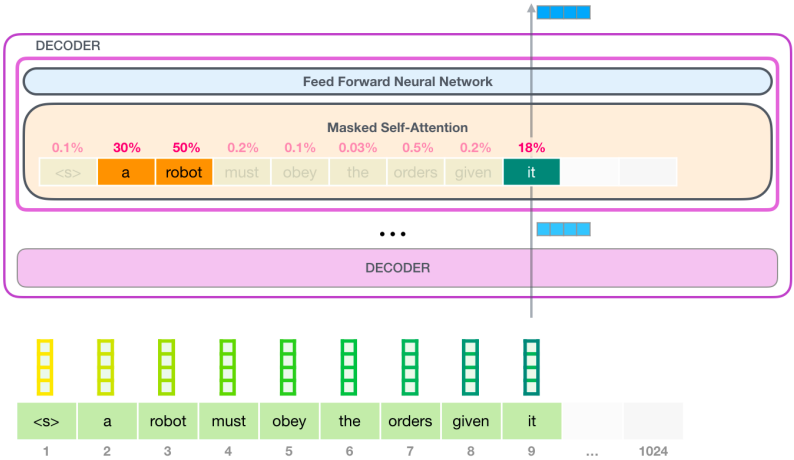


https://www.mdpi.com/2079-9292/13/17/3509

Generative Pre-Trained Transformer (GPT)

https://michaelfu1998-create.github.io/papers/vulrepair.pdf

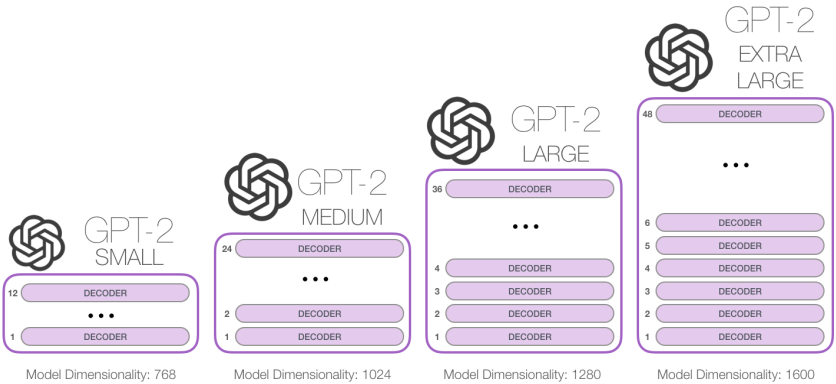
Generative Pre-Trained Transformer (GPT)



https://michaelfu1998-create.github.io/papers/vulrepair.pdf

Generative Pre-Trained Transformer (GPT)

- GPT released June 2018
- GPT-2 released Nov. 2019 with 1.5B parameters
- GPT-3: 175B parameters trained on 45TB texts



GPT Model Comparison

Model	Description	Data
GPT-2 (Radford et al., 2019)	Context size: 1024 tokens, 117M-1.5B parameters	WebText (45 million outbound links from Reddit with 3+ karma); 8 million documents (40GB)
GPT-3 (Brown et al., 2020)	Context size: 2048 tokens, 125M-175B parameters	Common Crawl + WebText + "two internet-based books corpora" + Wikipedia (400B tokens, 570GB)

Few-Shot, One-Shot, and Zero-Shot Learning in Language Models

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
```

Brown et al. (2020, "Language Models are Few-Shot Learners" https://arxiv.org/pdf/2005.14165.pdf

References I

- Asgari, E. "Natural language processing." Sharif University of Technology.
- Soleymani, M. "Machine learning." Sharif University of Technology.
- Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I. (2019). "Language Models are Unsupervised Multitask Learners." OpenAI Blog. Retrieved from https://openai.com/research/language-unsupervised
- Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., ... Amodei, D. (2020). "Language Models are Few-Shot Learners." *arXiv preprint arXiv:2005.14165*.
- Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... Stoyanov, V. (2019). "RoBERTa: A Robustly Optimized BERT Pretraining Approach." *arXiv preprint arXiv:1907.11692*.
- Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., Soricut, R. (2020). "ALBERT: A Lite BERT for Self-supervised Learning of Language Representations." In *Proceedings of the International Conference on Learning Representations (ICLR)*.
- Joshi, M., Chen, D., Liu, Y., Weld, D. S., Zettlemoyer, L., Levy, O. (2020). "SpanBERT: Improving Pre-training by Representing and Predicting Spans." *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 64-77.

References II

- Wettig, A., Baykal, C., Ruder, S., Søgaard, A. (2022). "Should All Tokens be Masked? A Pilot Study of Masked Language Model Performance on Diagnostic Classifiers." *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 10, pp. 10993-11001.
- Tjong Kim Sang, E. F., De Meulder, F. (2003). "Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition." In *Proceedings of the 7th Conference on Natural Language Learning at HLT-NAACL 2003*.
- Radford, A., Narasimhan, K., Salimans, T., Sutskever, I. (2018). "Improving Language Understanding by Generative Pre-Training." Retrieved from https://www.cs.ubc.ca/ amuham01/LING530/papers/radford2018improving.pdf