COS 597R Lecture 2 Notes: Pretraining I

Brendan Wang, Yinghui He, Haichen Dong

September 13, 2024

1 Introduction

This lecture focuses on the GPT-3 paper "Language Models are Few-Shot Learners" (Figure 1) [2]. This is a seminal paper with over 30k citations, one of the most cited papers in the past 5 years. In this lecture, we provide additional context and highlight important points of the paper, assuming students have read the paper carefully.

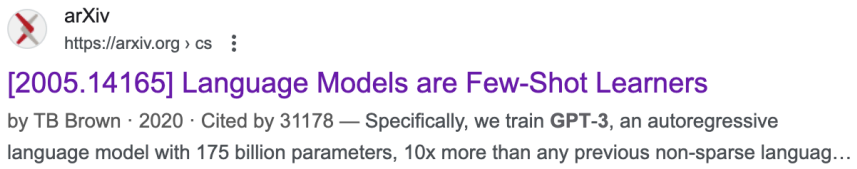

Figure 1: GPT-3 paper.

2 Brief History and Motivation

2.1 Word embeddings

Word embeddings. Word embeddings are single-layer representations that are learned using word vectors. Relevant works include word2vec [9] and GloVe [12].

Contextualized word embeddings. Contextualized word embeddings use RNNs with multiple layers of representations and contextual state to form stronger representations, extracting the hidden representations from RNN for every single word. This method was proposed five years after word embeddings. Contextualized word embeddings include ELMo [13] and CoVe [8].

Both types of word embeddings are used for task-specific (e.g., QA or translation) neural architectures.

2.2 Pre-trained models

An important idea emerged that we can just pretrain a single model, making minimal modifications (fine-tuning) to perform various downstream tasks. The fine-tuning process requires 10^3 to 10^5 examples to update a small number of parameters of the model.

Examples of pre-trained models are:

- BERT [3], RoBERTa [7] (encoder models)
- T5 [16], BART [6] (encoder-decoder models)

- GPT-1 [14], GPT-2 [15] (decoder models)

These models are all based on Transformers, mainly differing in the pre-training objectives. Their model sizes and pre-training data are also different.

2.3 Encoder vs decoder models

Figure 2 illustrates the different architecture of decoder models and encoder-decoder models. For decoder models, every layer has Masked Multi-Head Attention + Feed Forward component but no cross attention, while encoder-decoder models use the cross attention mechanism.

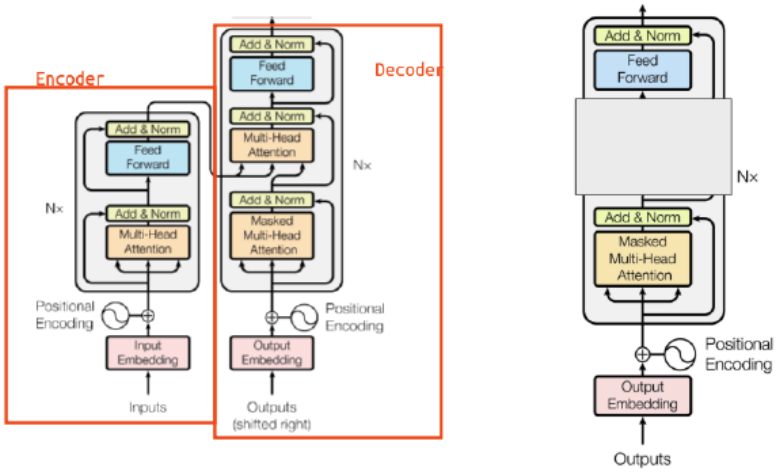

Figure 2: Architecture of encoder-decoder model (left) and decoder model (right).

Why encoder models like BERT or RoBERTa are not as popular today remains an open question. Scholar Yi Tay proposes several possible reasons[1]. First, encoder-only models can't generate text (easily) and are generally harder to scale up. Also, bidirectional attention may only be important at a smaller scale. Moreover, "masking objectives" can be still combined with autoregressive LMs.

3 GPT-3

GPT-3 is a transformer decoder-only model that was trained on large amounts of unlabeled text. Here we describe the main contribution, pretraining objectives, model architectures, training compute, training data and the approaches of GPT-3.

3.1 Overview

The GPT-3 paper has two main contributions. First, it trained an autoregressive language model of 175B parameters which is 10x larger than any previous LMs. Second, it introduced the concept of "in-context learning" and showed competitive performance compared to fine-tuned SOTA models.

Definition 1 (In-context learning)**.** A technique where an LLM performs a task from only a few examples or simple instructions without any gradient updates or fine-tuning.

[1] https://www.yitay.net/blog/model-architecture-blogpost-encoders-prefixlm-denoising

It is interesting to note that the idea of in-context learning starts from GPT-2, "though with much more limited results and no systematic study."

As the first to conduct a systematic analysis on few-shot learning (a form of in-context learning), the paper introduces several limitations of fine-tuning that can be addressed by in-context learning. First, collecting large supervised training sets is expensive or impractical. Second, fine-tuning can exploit spurious correlation and do not generalize well out-of-distribution. It is found in the Natural Language Inference task that models can take advantage of heuristics, such as depending on lexical overlap to determine if a premise entails a hypothesis. Finally, developing few-shot learning comes from the nature of human brain. Humans do not require large supervised datasets to learn most language tasks, which allows humans to seamlessly mix together or switch between many tasks and tasks when interacting with NLP systems. We want to interact with a single LM that possesses fluidity and generality, instead of many fine-tuned models. These three advantages serve as a strong rationale for the paper to dive into few-shot learning.

3.2 Pre-Training Objective

The model was trained using a next-token prediction training objective, which might be defined using the following loss function (or something similar):

$$L_1(\mathcal{U}) = \sum_i \log P(u_i | u_{i-k}, ..., u_{i-1}; \Theta) \tag{1}$$

This loss function is the sum of the log probabilities of a given sentence $\mathcal{U}$. Also, here, k refers to size of the previous context the model uses to predict the next token.

3.3 Model Architecture

GPT-3 has the same model architecture as GPT-2, "including the modified initialization, pre-normalization, and reversible tokenization" [2]. The only exception is that unlike GPT-2, GPT-3 also uses sparse attention, a modified attention mechanism in which the model attends to sparse subsets of the sentence instead of the whole sentence. The advantage of this approach is that it both speeds up computational training and inference.

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

Figure 3: Model architecture details for eight GPT-3 models.

Beyond the architecture details, we are also curious: do GPT-3 models obey scaling laws (which specify that the validation loss of models can be approximated as a smooth power law as a function of size)? To answer this question, the authors developed models across 8 different sizes to study the relationship between model size and model performance. The model architecture details are shown in Figure 3. The number of parameters ranged from 125 million parameters for the smallest model to 175 billion parameters for the largest, which the authors call "GPT-3 175B" or simply "GPT-3". All of these models were trained on the same training data which was comprised of 300 billion tokens using a context window size of 2048 tokens, model parallelism,

and an Adam optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.95$, and $\epsilon = 10^{-8}$. One interesting observation from the parameters the authors used was that larger models typically used a larger batch sizes, but required smaller learning rates. It is unclear why this is the case.

3.4 Training Compute

As expected, the larger of the GPT-3 models take a lot more train compute, measured both in terms of FLOPs and PFS-days. Interestingly, the authors illustrate that when comparing their GPT-3 models with a similarly-sized counterpart, the GPT-3 models can be trained using much fewer training tokens and epochs. For instance, although RoBERTa-Large (3B parameters) and GPT-3 2.7B are similar in size, GPT-3 2.7B took only 300 billion tokens and 1 training epoch compared to RoBERTa-Large which required 2000 billion training tokens and 40 epochs.

3.5 Training Data

GPT-3 was trained on a dataset that combined data from Common Crawl (nonprofit organization that crawls the web for content) with additional sets of high-quality curated data. These sets include WebText, Books1, Book2, and English Wikipedia.

Because data from Common Crawl (CC) contained lots of low-quality, duplicated, and noisy content, heavy filtering was required. One filtering technique involved filtering content based on its similarity to high-quality reference corpora, with respect to similarity metrics. In addition, fuzzy deduplication involved using key-word match techniques to dedepulicate CC.

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

Figure 4: Model architecture details for eight GPT-3 models.

Furthermore, the authors employ data sampling to create a data mix used for training. In particular, certain datasets were sampled according to their weight; when the weight of a dataset is greater than its proportion of the entire combined dataset, this leads to sampling from this dataset more frequently. Importantly, the authors found that sampling from high-quality data more frequently leads to better model performance. Additionally, when training the model, the number of epochs used also varied across the datasets. These final parameters were determined through experimentation and are shown in Figure 4. As there is a tradeoff between data quality and overfitting, how to precisely determine the optimal data mix remains a big area of research.

3.6 Training Approach

GPT-3 was evaluated with the use of in-context learning. The three in-context evaluation approaches are few-shot, one-shot, and zero-shot. As mentioned previously, these in-context learning approaches differ from fine-tuning in that they do not involve the update of the weights of the model.

- **Few-shot**: K demonstrations of the evaluation task are provided within the prompt given to the model. While the number of demonstrations K typically ranges from 10-100, K will depend on the number of demonstrations that fit in the context. An optional prompt

can also be added. Finally, a larger K does not necessarily lead to better performance; the optimal K can be determined using a development set.

- **One-shot**: A special case when $K = 1$. One-shot learning is thought to most closely resemble the way many tasks are communicated to humans. However, for certain tasks, it might be difficult to communicate the details without giving more examples.
- **Zero-shot**: No demonstrations are given to the model at inference time. While this can avoid the spurious correlation limitation of fine-tuning, zero-shot might be unfairly hard to the model.

Overall, the authors found that few-shot learning achieves stronger performance; on some tasks, it was only slightly behind SOTA fine-tuned models. However, with recent developments, researchers are increasingly curious for how to solve the LLM task with zero or one-shot learning since they make fair comparisons to human performance. For instance, Denny Zhou (Research Scientist at DeepMind) claimed that "few shot prompting will soon become obsolete." The

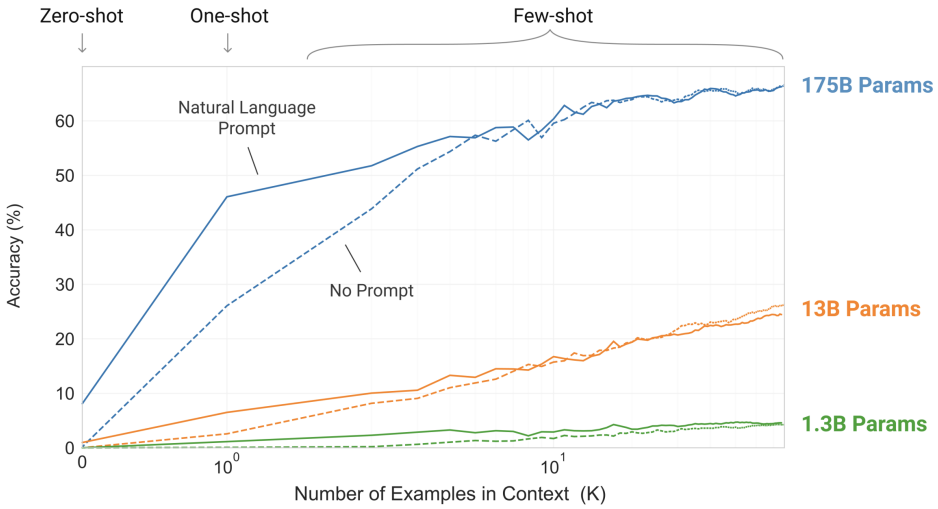

Figure 5: Accuracy performance of three GPT-3 models over K.

authors encapsulated the results across these approaches in the Figure 5. The solid and dashed lines refer to experiments where the input included and did not include a natural language prompt, respectively. Several key observations:

1. Accuracy increases with number of demonstration examples in context.
2. The slope of the green and orange lines is much smaller than the blue. This suggests that smaller models are less strong in their ability to do in-context learning.
3. The solid and dashed blue lines converge, which implies that after K reaches a certain point (we give the model sufficient context), inclusion of prompt is redundant.

4 Evaluation

There were a variety of evaluation tasks the authors used to benchmark the performance of GPT-3:

- Task similar to language modeling
- Closed-book question answering

- Machine translation
- Winograd schema and commonsense reasoning
- Reading comprehension
- SuperGLUE
- NLI
- Novel tasks: on-the-fly reasoning, adapation, open-ended text synthesis

Specifics on each evaluation task is described more in detail below.

4.1 Evaluation Protocol

GPT-3 was evaluated on evaluation tasks of various types. Specifically, the answers were generated in the following way:

- On tasks with free-form completion, beam search with a beam width of 4 and a length penalty of $\alpha = 0.6$ was applied.
- On tasks with multiple choice questions, the input prompt consisted of K in-context examples with questions and correct completions only, followed by the query context. Each completion option was feed separately, and per-token LM likelihood was compared. On a small number of datasets, there was additional benefits by measuring $\frac{P(\text{completion}|\text{context})}{P(\text{completion}|\text{answer\_context})}$, where answer_context is a string like `"Answer: "` to ensure that the model was outputting the answer.
- On tasks with binary classification, questions were treated like multiple choice questions with semantically meaningful options like `"True"` or `"False"` instead of using `"0"` or `"1"`.

As a sidenote, there are also other protocols to evaluate tasks with multiple choice questions. For example, some might put all options starting with labels like `"A: ..."` and `"B: ..."` in the context, and ask the model to output the chosen label followed by the option content. Further discussion on multiple choice question evaluation can be found in [18].

4.2 Language Modeling

In language modeling tasks, GPT-3 was given the context and was asked to predict the next words. GPT-3 set a new SOTA of zero-shot perplexity on the Penn Tree Bank (PTB) dataset. GPT-3 was also evaluated on the following datasets, with a summary of these results shown in Figure 6.

Setting	LAMBADA (acc)	LAMBADA (ppl)	StoryCloze (acc)	HellaSwag (acc)
SOTA	68.0[a]	8.63[b]	**91.8**[c]	**85.6**[d]
GPT-3 Zero-Shot	**76.2**	**3.00**	83.2	78.9
GPT-3 One-Shot	**72.5**	**3.35**	84.7	78.1
GPT-3 Few-Shot	**86.4**	**1.92**	87.7	79.3

Figure 6: Results on cloze and completion tasks.

- LAMBADA [11]: Given a paragraph of context, the model was asked to predict the last word which cannot be inferred from only the last sentense. This dataset tested the modeling of long-range dependencies in text, and an example can be found in Figure 7.

Context: He shook his head, took a step back and held his hands up as he tried to smile without losing a cigarette. "Yes you can," Julia said in a reassuring voice. "I 've already focused on my friend. You just have to click the shutter, on top, here."
Target sentence: He nodded sheepishly, through his cigarette away and took the ------.
Target word: camera

Figure 7: The LAMBADA dataset.

GPT-3 was tested under few-shot, one-shot, and zero-shot settings, and achived 86.4%, 72.5%, 76.2% accuracy, respectively, outperforming the previous state-of-the-art with 68% accuracy.

- StoryCloze [10]: Given a short 5-sentence story, the model was asked to choose the correct ending, as demonstrated in Figure 8. GPT-3 did not outperform the fune-tuned state-of-the-art, but improved over previous zero-shot results by roughly 10%.

Context	Right Ending	Wrong Ending
Karen was assigned a roommate her first year of college. Her roommate asked her to go to a nearby city for a concert. Karen agreed happily. The show was absolutely exhilarating.	Karen became good friends with her roommate.	Karen hated her roommate.

Figure 8: The StoryCloze dataset.

- HellaSwag [17]: The model was asked to pick the best ending to a story or set of instructions. These tasks required some commonsense reasoning, and were adversarially mined to be difficult for language models while remaining easy for humans.

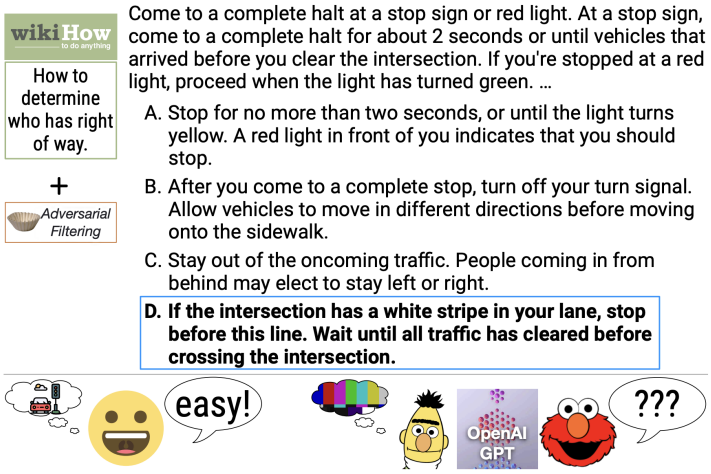

Figure 9: The HellaSwag dataset.

4.3 Open-domain Question Answering

The ability to answer questions about board factual knowledge of GPT-3 was measured. There are two settings as follows:

- Open-book: The model is allowed to search for and condition on text which potentially contains the answer.

- Close-book: The model has to answer the questions directly without conditioning on any auxiliary information.

GPT-3 was tested under the more restrictive and challenging close-book setting. Furthermore, dataset-specific fine-tuning was not allowed. The results are summarized in Figure 10. GPT-3's performance on open-domain QA tasks were mind-blowingly well, as it outperformed fine-tuned SOTA in the TriviaQA [4] dataset, and approached the performance of fine-tuned SOTA without any fine-tuning in WebQuestions [1] and Natural Questions [5] datasets.

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

Figure 10: Results on open-domain QA tasks.

4.4 Machine Translation

Despite the fact that GPT-3 was still primarily trained with English dataset (93% by word count), and it was not customized or designed for any translation tasks, GPT-3 showed some multilingual capability. Zero-shot GPT-3 underperformed previous unsupervised neural machine translation (NMT) works. However, by providing a few pairs on example demonstrations, one / few-shot GPT-3's performance improved and reached similar average performance to prior unsupervised NMT results. Unsepervised NMT used to be a smart topic that translated text without paired translation training data, but people lost interest after large language models like GPT-3 showed promising performance on those tasks. On the other hand, supervised machine translation requires paired training data, and GPT-3 still underperformed those supervised state-of-the-art.

A summary of results is shown in Figure 11. It is also worth mentioning that GPT-3 performed noticeably better when translating into English.

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

Figure 11: Results on machine translation. XLM, MASS, and mBART are unsupervised NMT.

References

[1] J. Berant, A. Chou, R. Frostig, and P. Liang. Semantic parsing on freebase from question-answer pairs. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1533–1544. ACL, 2013. URL `https://aclanthology.org/D13-1160/`.

[2] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei. Language models are few-shot learners. *CoRR*, abs/2005.14165, 2020. URL `https://arxiv.org/abs/2005.14165`.

[3] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019. URL `https://arxiv.org/abs/1810.04805`.

[4] M. Joshi, E. Choi, D. S. Weld, and L. Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *CoRR*, abs/1705.03551, 2017. URL `http://arxiv.org/abs/1705.03551`.

[5] T. Kwiatkowski, J. Palomaki, O. Redfield, M. Collins, A. P. Parikh, C. Alberti, D. Epstein, I. Polosukhin, J. Devlin, K. Lee, K. Toutanova, L. Jones, M. Kelcey, M. Chang, A. M. Dai, J. Uszkoreit, Q. Le, and S. Petrov. Natural questions: a benchmark for question answering research. *Trans. Assoc. Comput. Linguistics*, 7:452–466, 2019. doi: 10.1162/TACL_A_00276. URL `https://doi.org/10.1162/tacl_a_00276`.

[6] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension, 2019. URL `https://arxiv.org/abs/1910.13461`.

[7] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. Roberta: A robustly optimized bert pretraining approach, 2019. URL `https://arxiv.org/abs/1907.11692`.

[8] B. McCann, J. Bradbury, C. Xiong, and R. Socher. Learned in translation: Contextualized word vectors, 2018. URL `https://arxiv.org/abs/1708.00107`.

[9] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26, 2013.

[10] N. Mostafazadeh, N. Chambers, X. He, D. Parikh, D. Batra, L. Vanderwende, P. Kohli, and J. F. Allen. A corpus and evaluation framework for deeper understanding of commonsense stories. *CoRR*, abs/1604.01696, 2016. URL `http://arxiv.org/abs/1604.01696`.

[11] D. Paperno, G. Kruszewski, A. Lazaridou, Q. N. Pham, R. Bernardi, S. Pezzelle, M. Baroni, G. Boleda, and R. Fernández. The LAMBADA dataset: Word prediction requiring a broad discourse context. *CoRR*, abs/1606.06031, 2016. URL `http://arxiv.org/abs/1606.06031`.

[12] J. Pennington, R. Socher, and C. Manning. GloVe: Global vectors for word representation. In A. Moschitti, B. Pang, and W. Daelemans, editors, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar, Oct. 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1162. URL `https://aclanthology.org/D14-1162`.

[13] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. Deep contextualized word representations. *CoRR*, abs/1802.05365, 2018. URL `http://arxiv.org/abs/1802.05365`.

[14] A. Radford. Improving language understanding by generative pre-training. 2018.

[15] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[16] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer, 2023. URL `https://arxiv.org/abs/1910.10683`.

[17] R. Zellers, A. Holtzman, Y. Bisk, A. Farhadi, and Y. Choi. Hellaswag: Can a machine really finish your sentence? *CoRR*, abs/1905.07830, 2019. URL `http://arxiv.org/abs/1905.07830`.

[18] C. Zheng, H. Zhou, F. Meng, J. Zhou, and M. Huang. Large language models are not robust multiple choice selectors. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL `https://openreview.net/forum?id=shr9PXz7T0`.