GPT-NeoX-20B: An Open-Source Autoregressive Language Model

Sid Black*, Stella Biderman*, Eric Hallahan*,
Quentin Anthony, Leo Gao, Laurence Golding, Horace He, Connor Leahy, Kyle McDonell,
Jason Phang, Michael Pieler, USVSN Sai Prashanth, Shivanshu Purohit, Laria Reynolds,
Jonathan Tow, Ben Wang, and Samuel Weinbach

EleutherAI

Abstract

GPT-NeoX-20B is a 20 billion parameter autoregressive language model whose weights will be made freely and openly available to the public through a permissive license. It is, to the best of our knowledge, the largest dense autoregressive model that has publicly available weights. In this paper, we describe the model architecture and training, evaluate its performance, and discuss the broader impacts of its release. We are open-sourcing the training and evaluation code, as well as the model weights, at `https://github.com/EleutherAI/gpt-neox`.

1 Introduction

Over the past several years there has been an explosion in research surrounding large language models (LLMs) for natural language processing, catalyzed largely by the impressive performance of transformer based language models like BERT [Devlin et al., 2018], GPT-2 [Radford et al., 2019], GPT-3 [Brown et al., 2020], and T5 [Raffel et al., 2019]. One of the most impactful outcomes of this research has been the finding that the performance of LLMs scales predictably as a power-law with the number of parameters, with architecture details such as width/depth ratio having a minimal effect within a wide range [Kaplan et al., 2020]. A consequence of this has been an abundance of research focusing on scaling transformer models up to never before seen scales, resulting in models of up to 530B parameters [Smith et al., 2022], a scale that would have been almost unthinkable just a few years prior.

Today, there are dozens of publicly acknowledged LLMs in existence. The largest have more than two orders of magnitude more parameters than GPT-2, and even at that scale there are nearly a dozen different models. However, these models are almost universally the protected intellectual property of large tech companies, and are gated behind a commercial API, available only upon request, or not available for outsider use at all. To our knowledge, the only freely and publicly available dense autoregressive language models larger than GPT-2 are GPT-Neo (2.7B parameters) [Black et al., 2021b], GPT-J-6B [Wang and Komatsuzaki, 2021], Megatron-11B[1], Pangu-α-13B [Zeng et al., 2021], and the recently released Fairseq 6.7 and 13B [Artetxe et al., 2021] models.

In this paper, we release GPT-NeoX-20B, motivated by the belief that open access to LLMs is critical to advancing research in a wide range of areas—particularly in AI safety, mechanistic interpretability, and the study of how LLM capabilities scale. Many of the most interesting capabilities of LLMs only emerge above a certain number of parameters, and they have many properties that simply cannot be studied in smaller models. Although safety is often cited as a justification for keeping model weights private, we believe this is insufficient to prevent misuse, and is largely a limitation on the ability to probe and study LLMs for researchers not based at the small number of organizations that have access to state of the art language models.

*Lead authors. Authors after the first three are listed in alphabetical order. See Appendix A for individual contribution details.

[1] `https://github.com/pytorch/fairseq/tree/main/examples/megatron_11b`

In the following sections, we will give a broad overview of GPT-NeoX-20B's architecture and training hyperparameters, as well as detailing the hardware and software setup used for training and evaluating, and the choices made when designing the dataset and tokenization. We also address of some of the difficulties and unknowns we encountered in training such a large model, and close by dissecting the broader impacts its release may have.

We are making the model weights at evenly spaced 1000 step intervals throughout the whole of training available, as well as open-sourcing the training and evaluation code, at `https://github.com/EleutherAI/gpt-neox`. We hope that making a wide range of checkpoints throughout training freely available will facilitate research into the training dynamics of LLMs, as well as being impactful in the aforementioned areas of AI safety and interpretability.

2 Model Design and Implementation

GPT-NeoX-20B is an autoregressive transformer decoder model whose architecture largely follows that of GPT-3 [Brown et al., 2020], with a few notable deviations described below.

Params	Non-embedding	Layers	Model Dim	Heads	Batch Size	Learning Rate
20 B	19.9 B	44	6144	64	3.1 M	9.7×10^{-5}

Table 1: "Params" refers to all parameters while "Non-embedding" refers to non-embedding parameters and should be the parameter count used for scaling laws research. Batch size is presented in tokens rather than the number of contexts, following Brown et al. [2020]. For a full list of hyperparameters, see Appendix B.

2.1 Model Architecture

Although our architecture is largely similar to GPT-3, there are some notable differences. In this section we give a high-level overview of those differences, but ask the reader to refer to Brown et al. [2020] for full details of the model architecture.

Rotary Positional Embeddings Following on from our previous positive experiences [Biderman et al., 2021, Biderman, 2021, Wang and Komatsuzaki, 2021], we use rotary embeddings [Su et al., 2021] instead of the learned positional embeddings that OpenAI's GPT models use [Radford et al., 2018]. Rotary embeddings are a form of static relative positional embeddings. In brief, they twist the embedding space so that the attention of a token at position m to token at position n is linearly dependent on $m - n$. More formally, they modify the standard multiheaded attention equations from

$$\text{softmax}\left(\frac{1}{\sqrt{d}}\sum_{n,m}\mathbf{x}_m^T\mathbf{W}_q^T\mathbf{W}_k\mathbf{x}_n\right),$$

where $\mathbf{x}_m$, $\mathbf{x}_n$ are (batched) embeddings of tokens at position m and n respectively and $\mathbf{W}_q^T$, $\mathbf{W}_k$ are the query and key weights respectively to

$$\text{softmax}\left(\frac{1}{\sqrt{d}}\sum_{n,m}\mathbf{x}_m^T\mathbf{W}_q^T R^d_{\Theta,(n-m)}\mathbf{W}_k\mathbf{x}_n\right),$$

where $R^d_{\Theta,x}$ is a $d \times d$ block diagonal matrix with the ith block being a 2D rotation by $x\theta_i$ for hyperparameters $\Theta = \{\theta_i = 10000^{-2(i-1)/d} \mid i \in \{0, 1, 2, \ldots, (d-1)/2\}\}$. For a visual diagram of what rotary embeddings do, see Figure 1.

While Su et al. [2021] applies rotary embeddings to every embedding vector, we follow Wang and Komatsuzaki [2021] and instead apply it only to the first 25% of embedding vectors. Our experiments indicate that this strikes the best balance of performance and computational efficiency.[2]

[2]See the Weights & Biases reports here and here for further details.

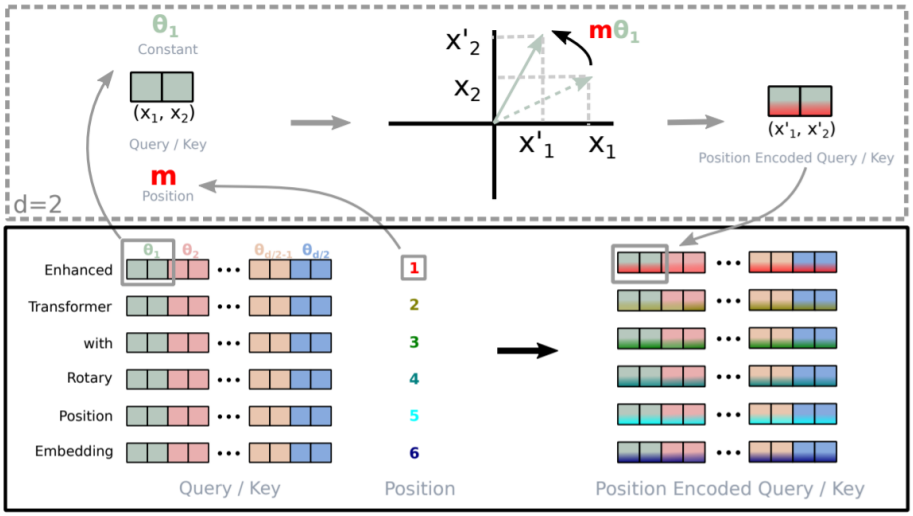

Figure 1: A pictorial representation of rotary embeddings, from Su et al. [2021].

Parallel Attention + FF Layers As in Wang [2021], we compute the Attention and Feed-Forward (FF) layers in parallel[3] and add the results, rather than running them in series. This is primarily for efficiency purposes, as each residual addition with op-sharding requires one all-reduce in the forward pass and one in the backwards pass [Shoeybi et al., 2019]. By computing the Attention and FFs in parallel, the results can be reduced locally before performing a single all-reduce. In Mesh Transformer JAX [Wang and Komatsuzaki, 2021], this led to a 15% throughput increase, while having comparable loss curves with running them in series during early training.

Initialization For the Feed-Forward output layers before the residuals, we used the initialization scheme introduced in Wang [2021],

$$\frac{2}{L\sqrt{d}}$$

This prevents activations from growing with increasing depth and width, with the factor of 2 compensating for the fact that the parallel and feed-forward layers are organized in parallel.

For all other layers, we use the *small init* scheme from Nguyen and Salazar [2019],

$$\sqrt{\frac{2}{d+4d}}$$

All Dense Layers While GPT-3 alternates between dense and sparse layers using the technique introduced in Child et al. [2019], we instead opt to exclusively use dense layers to reduce implementation complexity.

2.2 Software Libraries

Our model is trained using a custom codebase that we call GPT-NeoX [Black et al., 2021a]. GPT-NeoX builds on Megatron [Shoeybi et al., 2019] and DeepSpeed [Rasley et al., 2020] to facilitate efficient and straightforward training of large language models with tens of billions of parameters. We use the official PyTorch v1.10.0 release binary package compiled with CUDA 11.1. This package is bundled with NCCL 2.10.3 for distributed communications.

[3]See `https://github.com/EleutherAI/gpt-neox/blob/ac3d8087f1762213880523893a52329d66d2d1a9/megatron/model/transformer.py#L593` for implementation details.

2.3 Hardware

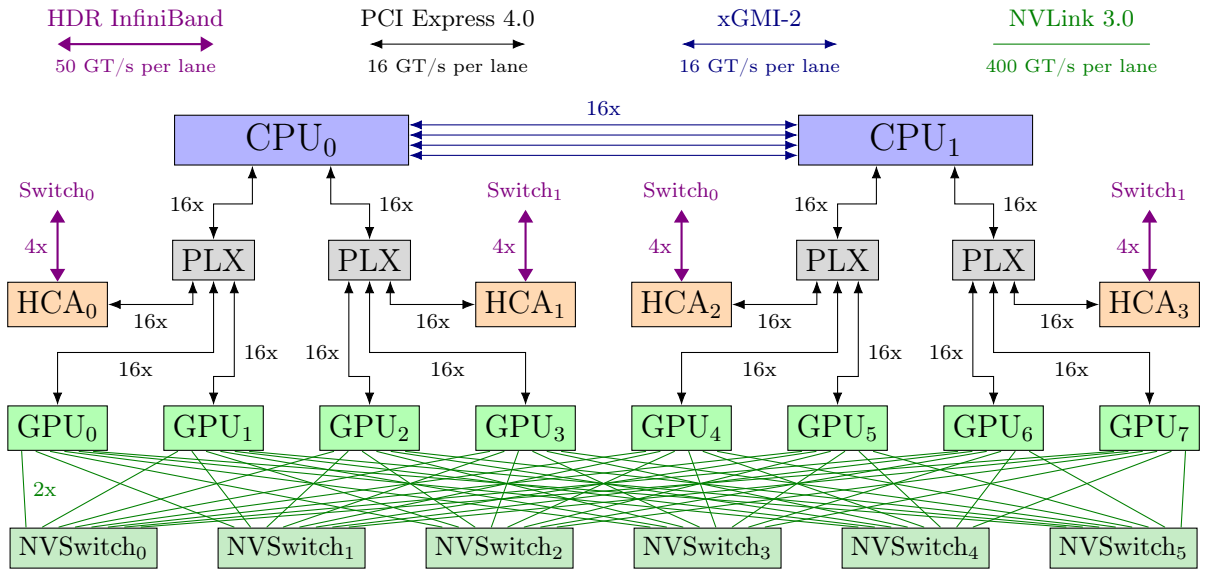

Figure 2: Architecture diagram of a single training node.

We trained GPT-NeoX-20B on twelve Supermicro AS-4124GO-NART servers, each with eight NVIDIA A100-SXM4-40GB GPUs and configured with two AMD EPYC 7532 CPUs. All GPUs can directly access the InfiniBand switched fabric through one of four ConnectX-6 HCAs for GPUDirect RDMA. Two NVIDIA MQM8700-HS2R switches—connected by 16 links—compose the spine of this InfiniBand network, with one link per node CPU socket connected to each switch. Figure 2 shows a simplified overview of a node as configured for training.

3 Training

Due to the intractability of performing a hyperparameter sweep for a 20 billion parameter model, we opted to use the values from Brown et al. [2020] to guide our choice of hyperparameters. As Brown et al. [2020] did not train a model at our exact scale, we interpolate between the learning rates of their 13B and 175B models to arrive at a learning rate of 0.97E−5. Based on the results of smaller scale experiments, we select a weight decay of 0.01. To achieve a higher training throughput, we opt to use the same batch size as OpenAI's 175B model–approximately 3.15M tokens, or 1538 contexts of 2048 tokens each, and train for a total of 150,000 steps, decaying the learning rate with a cosine schedule to 10% of its original value at the end of training.

We use the AdamW [Loshchilov and Hutter, 2017] optimizer, with beta values of 0.9 and 0.95 respectively, and an epsilon of 1.0E−8. We extend AdamW with the *ZeRO* optimizer [Rajbhandari et al., 2019] to reduce memory consumption by distributing optimizer states across ranks. Since the weights and optimizer states of a model at this scale do not fit on a single GPU, we use the tensor parallelism scheme introduced in Shoeybi et al. [2019] in combination with pipeline parallelism [Harlap et al., 2018] to distribute the model across GPUs. To train GPT-NeoX-20B, we found the most efficient way to distribute the model given our hardware setup to be a tensor parallel size of 2, and a pipeline parallel size of 4—allowing the most communication intensive processes, tensor and pipeline parallelism, to occur within a node, and data parallel communication to occur across node boundaries.

3.1 Training Data

GPT-NeoX-20B was trained on The Pile [Gao et al., 2020], a massive curated dataset that we designed specifically for training large language models. It consists of data from 22 data sources, coarsely broken down into 5 categories:

- **Academic Writing**: Pubmed Abstracts and PubMed Central, arXiv, FreeLaw,[4] USPTO Backgrounds,[5] PhilPapers,[6] NIH Exporter[7]
- **Web-scrapes and Internet Resources**: CommonCrawl, OpenWebText2, StackExchange,[8] Wikipedia (English)
- **Prose**: BookCorpus2, Bibliotik, Project Gutenberg [PG-19; Rae et al., 2019]
- **Dialogue**: Youtube subtitles, Ubuntu IRC,[9] OpenSubtitles [Lison and Tiedemann, 2016], Hacker News,[10] EuroParl [Koehn, 2005]
- **Miscellaneous**: GitHub, the DeepMind Mathematics dataset [Saxton et al., 2019], Enron Emails [Klimt and Yang, 2004]

In aggregate, the Pile consists of over 825GiB of raw text data. The diverse data sources reflects our desire for a general-purpose language model. Certain components are up-sampled to obtain a more balanced data distribution. In contrast, GPT-3's training data consists of web-scrapes, books datasets, and Wikipedia. When comparing results in this work to GPT-3, the training data is almost certainly the biggest known unknown factor. Full details of the Pile can be found in our technical report [Gao et al., 2020] and the associated datasheet [Biderman et al., 2022].

It is particularly notable that the Pile contains a scrape of StackExchange preprocessed into a Q/A form. Recent work [Biderman and Raff, 2022] has shown that this formulation heavily influences code generation, with prompts such as "write a Java program that accepts 10 integers and shows them in reversed order." producing not only a code solution but also a natural language discussion of that code, as one might see on Stack Exchange.

3.2 Tokenization

For GPT-NeoX-20B, we use a BPE-based tokenizer similar to that used in GPT-2, with the same total vocabulary size of 50257. We make three major changes to the tokenizer. First, we train a new BPE tokenizer based on the Pile, taking advantage of its diverse text sources to construct a more general-purpose tokenizer. Second, in contrast to the GPT-2 tokenizer which treats tokenization at the start of a string as a non-space-delimited token, the GPT-NeoX-20B tokenizer applies consistent space delimitation regardless. This resolves an inconsistency regarding the presence of prefix spaces to a tokenization input.[11] An example can be seen in Figure 3. Third, our tokenizer contains tokens for repeated space tokens (all positive integer amounts of repeated spaces up to and including 24). This allows the GPT-NeoX-20B tokenizer to tokenize text with large amounts of whitespace using fewer tokens; for instance, program source code or arXiv LaTeXsource files.

3.2.1 Tokenizer Comparisons on Pretraining Corpora

Both tokenizers share 36938 out of 50257 tokens, a ∼73.5% overlap in tokens. In this section, we perform comparison between the GPT-NeoX-20B tokenizer to the GPT-2 tokenizer using the validation set of the Pile.

In Table 2a, we show the resulting number of tokens from tokenizing each component of the Pile's validation set with both tokenizers, and the ratio of GPT-NeoX-20B tokens to GPT-2 tokens.

We see that the GPT-NeoX-20B tokenizer represents all Pile components using fewer or very closely comparable numbers of tokens. The largest percentage improvement in token counts are in the EuroParl, GitHub, and PubMed Central components, with a more than 20% savings in the number of tokens needed to represent that component. We highlight that arXiv, GitHub, and StackExchange—subsets with large

[4] https://www.courtlistener.com/
[5] https://bulkdata.uspto.gov/
[6] https://philpapers.org/
[7] https://exporter.nih.gov/
[8] https://archive.org/details/stackexchange
[9] https://irclogs.ubuntu.com/
[10] https://news.ycombinator.com/
[11] https://discuss.huggingface.co/t/bpe-tokenizers-and-spaces-before-words/475/2

Figure 3: GPT-2 tokenization vs. GPT-NeoX-20B tokenization. GPT-NeoX-20B tokenization handles whitespace better, which can be particularly useful for text such as source code. For more examples, see Appendix C.

	GPT-2	GPT-NeoX-20B	$\frac{\text{GPT-NeoX-20B}}{\text{GPT-2}}$
arXiv	41,020,155	34,704,315	0.84603
BookCorpus2	2,336,388	2,365,633	1.01252
Books3	42,819,036	43,076,832	1.00602
DM Mathematics	7,699,527	7,413,775	0.96289
Enron Emails	480,500	433,867	0.90295
EuroParl	3,519,584	2,808,275	0.79790
FreeLaw	21,098,168	18,687,364	0.88573
GitHub	42,986,216	33,021,839	0.76820
Gutenberg (PG-19)	6,729,187	6,428,946	0.95538
HackerNews	2,578,933	2,551,720	0.98945
NIH ExPorter	776,688	739,558	0.95219
OpenSubtitles	5,431,529	5,446,485	1.00275
OpenWebText2	31,993,480	30,813,744	0.96313
PhilPapers	1,879,206	1,750,928	0.93174
Pile-CC	53,415,704	53,392,389	0.99956
PubMed Abstracts	8,708,180	8,215,529	0.94343
PubMed Central	56,874,247	43,534,166	0.76545
StackExchange	22,708,643	19,000,198	0.83669
USPTO Backgrounds	10,217,886	9,727,223	0.95198
Ubuntu IRC	3,341,287	2,771,066	0.82934
Wikipedia (en)	12,614,087	12,692,048	1.00618
YoutubeSubtitles	3,883,103	3,311,907	0.85290
Total	383,111,734	342,887,807	0.89501

(a) All tokens

	GPT-2	GPT-NeoX-20B	$\frac{\text{GPT-NeoX-20B}}{\text{GPT-2}}$
arXiv	38,932,524	33,561,364	0.86204
BookCorpus2	2,233,367	2,262,609	1.01309
Books3	40,895,236	41,198,424	1.00741
DM Mathematics	7,214,874	6,929,066	0.96039
Enron Emails	374,978	373,498	0.99605
EuroParl	3,482,120	2,780,405	0.79848
FreeLaw	17,766,692	17,434,708	0.98131
GitHub	29,338,176	27,558,966	0.93936
Gutenberg (PG-19)	5,838,580	5,827,408	0.99809
HackerNews	2,312,116	2,299,848	0.99469
NIH ExPorter	776,619	739,543	0.95226
OpenSubtitles	5,428,118	5,445,721	1.00324
OpenWebText2	30,849,218	29,723,143	0.96350
PhilPapers	1,872,347	1,743,627	0.93125
Pile-CC	51,305,080	51,281,909	0.99955
PubMed Abstracts	8,676,790	8,185,417	0.94337
PubMed Central	44,508,570	40,722,151	0.91493
StackExchange	17,414,955	16,712,814	0.95968
USPTO Backgrounds	9,882,473	9,601,385	0.97156
Ubuntu IRC	3,220,797	2,659,225	0.82564
Wikipedia (en)	11,874,878	11,986,567	1.00941
YoutubeSubtitles	3,589,042	3,046,451	0.84882
Total	337,787,550	322,074,249	0.95348

(b) Excluding whitespace tokens

Table 2: Number of tokens from tokenizing the Pile validation set. (a) shows the full token count, (b) exclude whitespace tokens

code components—can be represented with meaningfully fewer tokens with the GPT-NeoX-20B tokenizer compared to the GPT-2 tokenizer. Overall, the GPT-NeoX-20B tokenizer represents the Pile validation set with approximately 10% fewer tokens compared to the GPT-2 tokenizer.

As our tokenizer is tweaked to better tokenize whitespace, we also perform a comparison between the two tokenizers excluding whitespace. We perform the same analysis as the above, but exclude all whitespace tokens from our computations, only counting the non-whitespace tokens. A token is considered a whitespace token if it consists only of whitespace characters. The results are shown in Table 2b. We observe that the GPT-NeoX-20B tokenizer still uses 5% fewer tokens to represent the Pile validation set compared to the GPT-2 tokenizer. As expected, the token ratios for certain components such as GitHub and StackExchange become closer to even once the whitespace characters are excluded.

	GPT-2	GPT-NeoX-20B	$\frac{\text{GPT-NeoX-20B}}{\text{GPT-2}}$
C4 Tokens	173,669,294	173,768,876	1.00057
C4 Tokens excl. Space	168,932,391	171,003,008	1.01226

Table 3: Number of tokens from tokenizing the AllenAI C4 (`en`) validation set.

Given that the GPT-NeoX-20B is trained on the Pile, the Pile components would be considered in-domain for the tokenizer, and hence it may not provide the most informative comparison between the two. To perform

an out-of-domain comparison, we perform the same analysis using the AllenAI replication of C4,[12], another popular pretraining corpus for large language models. As above, we use the validation set for our analysis. Our results are shown in Table 3. We find that the GPT-NeoX-20B tokenizer tokenizes the C4 validation set to approximately the same number of tokens as the GPT-2 tokenizer. When excluding all whitespace tokens, the GPT-NeoX-20B requires approximately 1% more tokens to represent the corpus compared to the GPT-2 tokenizer.

3.2.2 Longest Tokens

We show in Table 4 the 10 longest tokens in each tokenizer vocabulary. We exclude consideration of tokens that comprise only symbols or whitespace characters. We observe that for the GPT-2 tokenizer, many of the longest tokens appear to reflect artifacts in the tokenizer training data, likely with certain websites or web-scrapes being overrepresented in the training data. For the GPT-NeoX-20B tokenizer, we observe that most of the longest tokens are scientific terms, likely arising from the PubMed components of the Pile.

GPT-2	GPT-NeoX-20B
`rawdownloadcloneembedreportprint`	`Ġimmunohistochemistry`
`BuyableInstoreAndOnline`	`Ġimmunohistochemical`
`cloneembedreportprint`	`Ġtelecommunications`
`ĠRandomRedditorWithNo`	`Ġimmunofluorescence`
`Ġtelecommunications`	`Ġimmunosuppressive`
`channelAvailability`	`ĠBytePtrFromString`
`Ġdisproportionately`	`Ġmultidisciplinary`
`ĠTelecommunications`	`Ġhistopathological`
`ĠguiActiveUnfocused`	`Ġneurodegenerative`
`ItemThumbnailImage`	`Ġindistinguishable`

Table 4: Ten longest tokens (excluding tokens comprising mainly symbols, numbers and spaces) in tokenizer vocabularies. "Ġ" indicates a word delimiter.

3.2.3 Worst Case Word Tokenization Comparison

GPT-2 Worst-case Tokenization				**GPT-NeoX-20B Worst-case Tokenization**			
Word	GPT-2 Tokenization		GPT-NeoX-20B Tokenization	Word	GPT-2 Tokenization		GPT-NeoX-20B Tokenization
hematopoietic	(6)	`hem|at|op|oiet|ic`	(1) `hematopoietic`	Schwarzenegger	(1)	`Schwarzenegger`	(5) `Schw|arzen|egger`
adenocarcinoma	(6)	`aden|oc|arc|in|oma`	(1) `adenocarcinoma`	Bolshevik	(1)	`Bolshevik`	(4) `Bol|she|v|ik`
MERCHANTABILITY	(5)	`MERCH|ANT|ABILITY`	(1) `MERCHANTABILITY`	crowdfunding	(1)	`crowdfunding`	(4) `crow|d|fund|ing`
CONSEQUENTIAL	(5)	`CONSEQU|ENTIAL`	(1) `CONSEQUENTIAL`	misogyny	(1)	`misogyny`	(4) `mis|o|gyn|y`
oligonucleotides	(5)	`oligon|ucle|otides`	(1) `oligonucleotides`	McAuliffe	(1)	`McAuliffe`	(4) `McA|ul|iff|e`
cytoplasmic	(5)	`cyt|opl|asm|ic`	(1) `cytoplasmic`	unstoppable	(1)	`unstoppable`	(4) `unst|opp|able`
corticosteroids	(4)	`cort|ic|oster|oids`	(1) `corticosteroids`	Timberwolves	(1)	`Timberwolves`	(4) `Timber|w|olv|es`
neurodegenerative	(4)	`neur|odeg|ener|ative`	(1) `neurodegenerative`	excruciating	(1)	`excruciating`	(4) `excr|uci|at|ing`
asymptotic	(4)	`as|ymp|tot|ic`	(1) `asymptotic`	Kaepernick	(1)	`Kaepernick`	(4) `Kaep|ern|ick`
aneurysm	(4)	`an|eur|ys|m`	(1) `aneurysm`	Valkyrie	(1)	`Valkyrie`	(4) `Valk|y|r|ie`

Table 5: Worst case word tokenization with respective tokenizers. We show cases where one tokenizer requires many more tokens to represent a word compared to the other tokenizer.

We consider the words for which there is the greatest discrepancy in the resulting token length between the two tokenizers, where one tokenizer needs many tokens to represent while the other tokenizer uses relatively few tokens. We define a word as a contiguous string delimited by whitespace or punctuation (as defined by `strings.punctuation` in Python). We perform this analysis at the component level. We only consider words that occur at least 10 times within the given component. We show in Table 5 a representative example from the Pile-CC corpus.

[12] `https://github.com/allenai/allennlp/discussions/5056`

4 Performance Evaluations

To evaluate our model we use Gao et al. [2021b], an open source codebase for language model evaluation that supports a number of model APIs. We compare with the GPT-3 API [Brown et al., 2020] [13], the open source FairSeq dense models [Artetxe et al., 2021], and GPT-J [Wang and Komatsuzaki, 2021]. We do not compare against T5 [Raffel et al., 2019] as our evaluation methodology assumes the models are autoregressive, or T0 [Sanh et al., 2021] because the T0 API does not allow for computing log-likelihoods.

While it is common to display "scaling laws" curves of best fit, we opt to not do so as the small number of OpenAI API models give DaVinci an outsized influence on the slope of the curve. Instead, we connect the points with lines directly. A dashed line between GPT-J and GPT-NeoX-20B acknowledges the fact that, while the models are very similar, they are not the same model trained at two different scales the way the FairSeq and OpenAI models are.

When we were able to obtain the relevant information, we report two baselines: human level performance and random performance. All plots contain error bars representing two standard errors, meaning that each interval is a 95% confidence interval around the point. For some plots, the standard error is so small that the interval is not visible.

4.1 Natural Language Tasks

We evaluate our model on a diverse collection of standard language modeling datasets. In general, we find that GPT-NeoX-20B performs on par with or marginally worse than FairSeq 13B and approximately on the linear interpolation between the performance of OpenAI's Curie and DaVinci models.

4.2 Knowledge-Based Tasks

In addition to evaluating on natural language processing benchmarks, we are also interested in the ability of our models to answer factual questions requiring advanced knowledge. To do this, we use a dataset of multiple choice questions in a variety of diverse domains developed by Hendrycks et al. [2020]. As individual subjects are rather noisy, we follow [Hendrycks et al., 2020] by focusing on results aggregated by subject area: Humanities, Social Sciences, STEM, and Miscellaneous as presented in Figure 5. We report full results in the appendix.

4.3 Mathematical Competency

Due to the fact that large language models tend to perform quite poorly on both arithmetic tasks and mathematical problems, we opted to include mathematical texts in various forms (arXiv, DM Mathematics, Math Stack Exchange and Math Overflow) as a significant portion of our training data in an attempt to improve performance in these areas.

We evaluate on the MATH test dataset [Hendrycks et al., 2021]. Note that this is an evaluation metric that is generally finetuned on, but due to computational limitations we only evaluate models zero-shot here.

We also find that GPT-J, also trained on the Pile, matches GPT-3 175B's performance despite being 30x smaller. While we leave finetuning GPT-J and GPT-NeoX to future work, we view this as a strong indicator that pretraining on the Pile is an effective way to improve performance on mathematics tasks.

5 Discussion

Whilst the performance of the released 20B parameter model is impressive in many respects, outperforming our previous best performing model, GPT-J-6B, on most benchmarked datasets, it is clear from comparing to Fairseq's 13B parameter model [Artetxe et al., 2021], and extrapolating based on OpenAI's models' performance, that the performance on natural language tasks in particular could be improved, whilst the performance in other areas, such as scientific literature and mathematics excels. In this section we will try to

[13] The numbers do not always agree with the numbers reported in Brown et al. [2020], because our prompt formatting is slightly different from theirs.

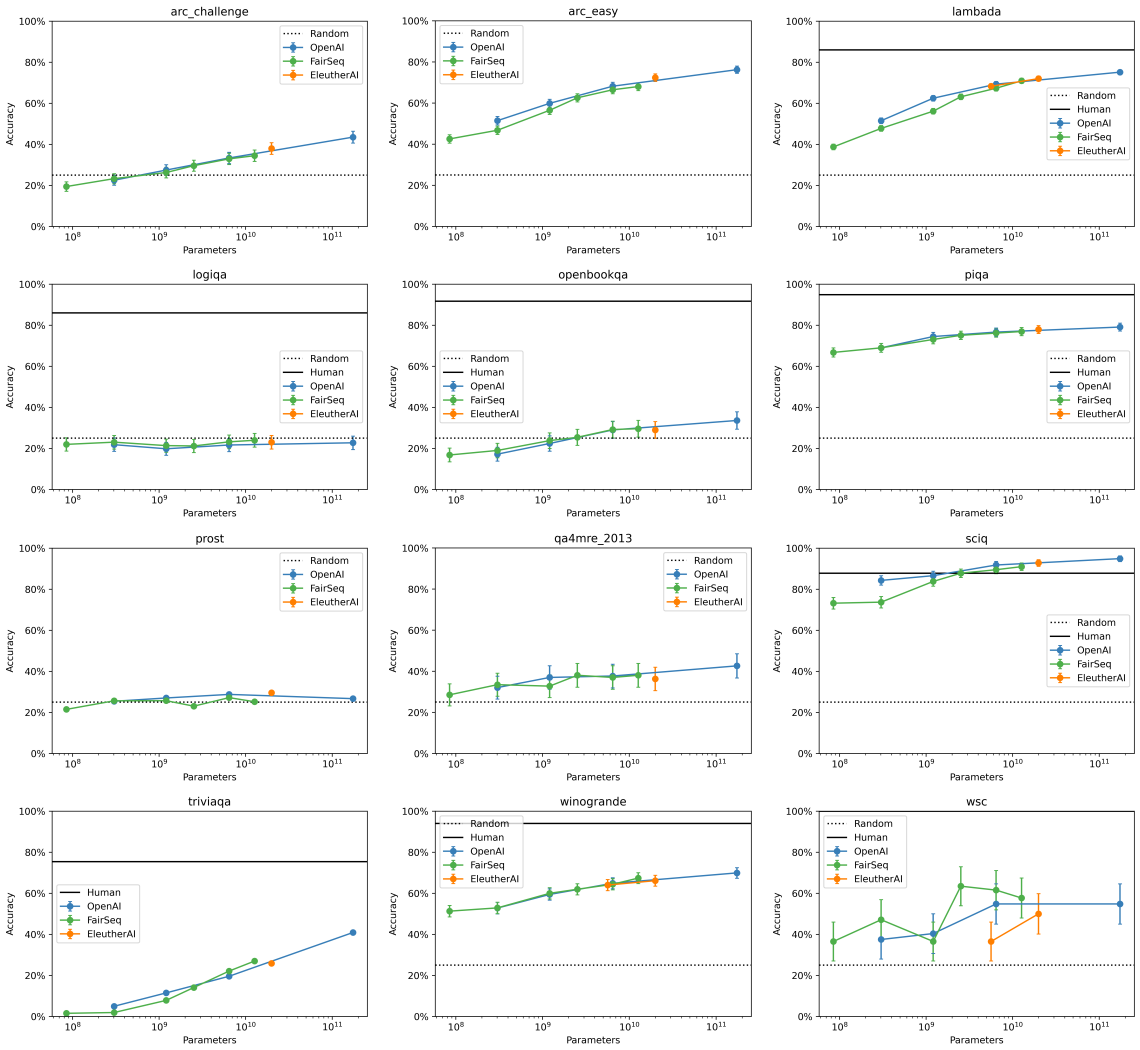

Figure 4: Zero-shot performance of GPT-NeoX-20B compared to GPT-J-6B and FairSeq and OpenAI models on a variety of language modeling benchmarks. Length-Normalized plots can be seen in Figure 13

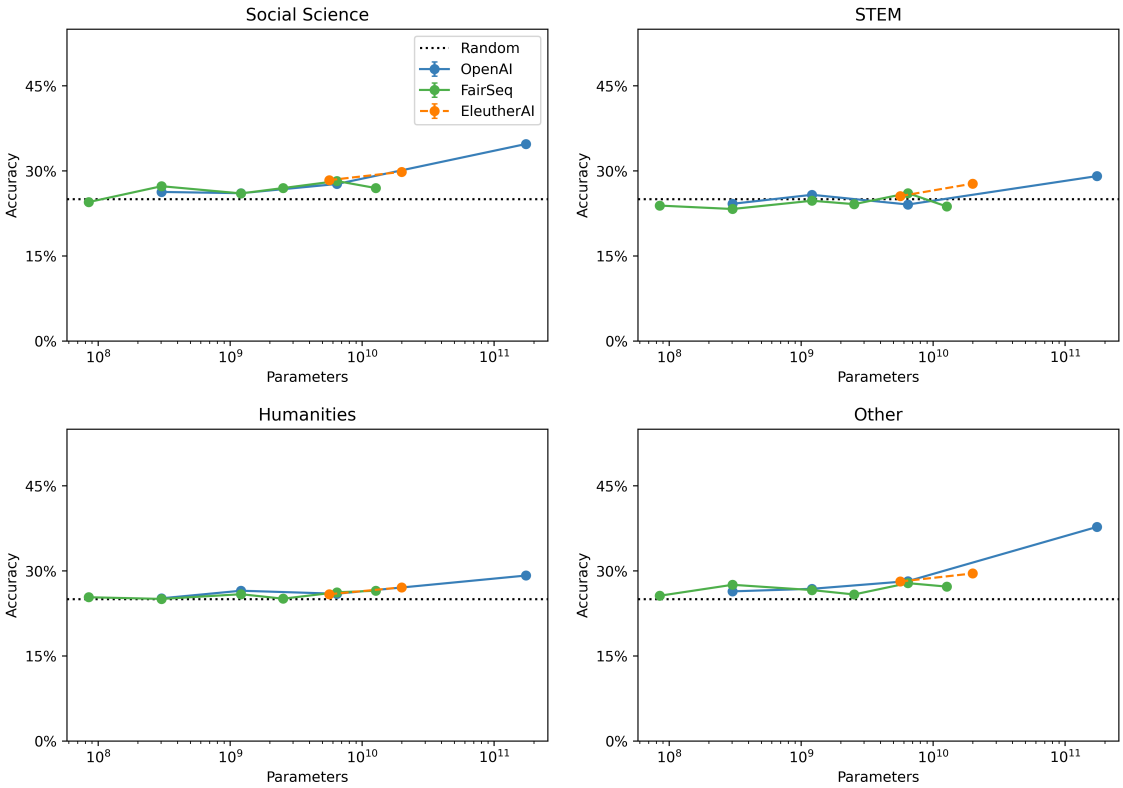

Figure 5: Zero-shot performance of GPT-NeoX-20B compared to GPT-J-6B and FairSeq and OpenAI models on Hendrycks et al. [2020]. We were unable to find information on median human performance.

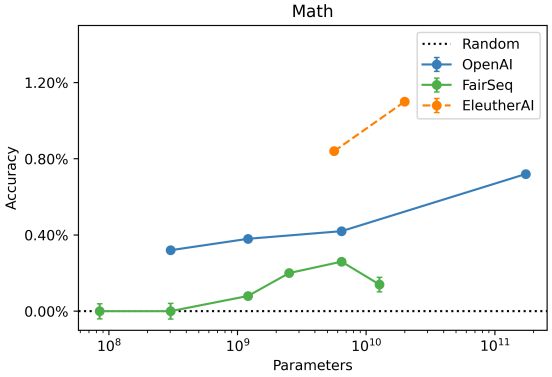

Figure 6: Zero-shot performance of GPT-NeoX-20B, GPT-J-6B, and GPT-Neo 2.7B compared to FairSeq and OpenAI models on the MATH dataset. Random performance on this task is 0%, and we were unable to find information on median human performance.

dissect what the causes for these performance differences could be, and how we might be able to address them in future iterations.

Tokenizer and Dataset. We expect some performance regression on web- and book-based language modeling benchmarks due to our tokenizer design favoring scientific and programming documents [Section 3.2] over web documents, as well as the former being much more prevalent in the Pile when compared to OpenAI's training set. However, we did not have the resources to quantify the expected performance differences from these design choices, and leave this to future work.

Hyperparameter Tuning. Hyperparameter tuning is an expensive process, which for multi-billion parameter networks is often infeasible to do at full scale. Due to the aforementioned limitations, we opted to choose hyperparameters based on a mixture of experiments at smaller scales and by interpolating parameters appropriate for our model size based on previously published work [Brown et al., 2020]. However, several aspects of both our model architecture [Section 2.1] and our training setup, including the data [Section 3.1] and the tokenizer [Section 3.2], diverge significantly from Brown et al. [2020]. As such, it is almost certainly the case that the hyperparameters used for those models are no longer optimal, and potentially never were.

The effects of certain hyperparameter choices on language models are sadly understudied, to the extent that many aspects of training large transformer models are effectively a folk art, only accessible to staff at a select number of private companies. For example, literature rigorously studying the effect of weight decay in language models is particularly scant [Zhou et al., 2021], [Zhang et al., 2018], and likewise, there is very little research investigating the effects of choices in designing a tokenizer—in particular the choice of vocabulary size—on downstream performance.

It is possible that our choice of 0.01 for weight decay was sub-optimal compared to GPT-3's choice of 0.1, however, we note that Fairseq's recently released LMs [Artetxe et al., 2021] also used a weight decay of 0.01. Another possible source of performance regression could have been our choice of batch size. For efficiency reasons, we opted to use a fairly large batch size of 3.2M tokens during training, the same as was used by the 175B GPT-3 model. The critical batch size [McCandlish et al., 2018] for the model may have been lower, possibly resulting in a negative impact on training. In addition, the optimal learning rate—which we selected by interpolating between the learning rates for models in [Brown et al., 2020]—can often be affected by the batch size, and failing to quantify this effect accurately could also have led to suboptimal performance. During the training of the model, however, new research has been released which may significantly alleviate the costs of hyperparameter tuning for future training runs [Yang et al., 2021].

Dataset Deduplication. Finally, the lack of dataset deduplication could also have had an impact on downstream performance. Recent research has shown that deduplicating training data can have a large effect on perplexity scores [Lee et al., 2021]. Although GPT-J-6B was trained on the same data and performed on par with OpenAI's models of a similar size, it seems plausible that this effect could become particularly apparent with a larger model size.

6 Broader Impacts

We believe that Transformative Artificial Intelligence (TAI) [Karnofsky et al., 2016] is approaching [Cotra, 2020, Grace et al., 2018], and that these systems will cause catastrophic damage if they are misaligned with human values [Fox and Shulman, 2013, Omohundro, 2008]. As such, we believe it is essential to prioritize and help facilitate technical research that ensures TAI's values will be aligned with ours.

AI Alignment generally refers to the problem of how to ensure increasingly powerful and autonomous AI systems perform the users' wishes faithfully and without unintended consequences. Alignment is especially critical as we approach human and superhuman levels of intelligence, as powerful optimization processes amplify small errors in goal specification into large misalignments [Goodhart, 1984, Manheim and Garrabrant, 2019, Fox and Shulman, 2013], and misalignments in this regime will result in runaway optimization processes that evade alteration or shutdown [Omohundro, 2008, Benson-Tilsen and Soares, 2016, Turner et al., 2021], posing a significant existential risk to humanity. Additionally, even if the goal is specified correctly, superhuman models may still develop deceptive subsystems that attempt to influence the real world to satisfy their objectives [Hubinger et al., 2021]. While current systems are not yet at the level where the consequences of misalignment pose an existential threat, rapid progress in the field of AI has increased the concern that the alignment problem may be seriously tested in the not-too-distant future.

Much of the alignment literature focuses on the more theoretical aspects of alignment [Demski and Garrabrant, 2020, Yudkowsky and Soares, 2018, Taylor, 2016, Garrabrant et al., 2016, Armstrong and Mindermann, 2018, Hubinger et al., 2021], abstracting away the specifics of how intelligence will be implemented, due to uncertainty over the path to TAI. However, with the recent advances in capabilities, it may no longer be the case that the path to TAI is completely unpredictable. In particular, recent increases in the capabilities of large language models (LLMs) raises the possibility that the first generation of transformatively powerful AI systems may be based on similar principles and architectures as current large language models like GPT. This has motivated a number of research groups to work on "prosaic alignment" [Christiano, 2016, Askell et al., 2021, Ouyang et al., 2021], a field of study that considers the AI alignment problem in the case of TAI being built primarily with techniques already used in modern ML. We believe that due to the speed of AI progress, there is a significant chance that this assumption is true, and, therefore, that contributing and enabling contributions to prosaic alignment research will have a large impact.

The open-source release of this model is motivated by the hope that it will allow alignment researchers who would not otherwise have access to LLMs to use them. While there are negative risks due to the potential acceleration of capabilities research, which may place further time pressure on solving the alignment problem, we believe the benefits of this release outweigh the risks of accelerating capabilities research.

6.1 The Usefulness of Large Language Models in Alignment

LLMs represent a different paradigm than the AI systems generally studied by alignment researchers because they are not well-described as coherent agents or expected utility maximizers. Though trained to optimize a log-likelihood loss function, at a high level the goals a LLM pursues are varied and contradictory, depending on the way it is prompted. This introduces additional challenges, but may also enable new approaches to alignment.

GPT-NeoX-20B itself is not the system we need to align, but we hope it can serve as a publicly available platform for experiments whose results might generalize to crucial future work.

The following is a non-exhaustive list of potential approaches we consider promising for further investigation.

Mechanistic interpretability. Mechanistic interpretability research [Cammarata et al., 2020] hopes to gain an understanding into *how* models accomplish the tasks they do, in part in the hopes of detecting problematic or deceptive algorithms implemented by models before these failures manifest in the real world. Being able to interpret and inspect the detailed inner workings of trained models would be a powerful tool to ensure models are optimizing for the goals we intended [Hubinger et al., 2021, Koch et al., 2021]. Reverse engineering transformer language models has already yielded insights about the inner functioning of LMs [Elhage et al., 2021, nostalgebraist, 2020, Anonymous, 2022, Dai et al., 2021].

Using a LLM as a reward model. Because they are trained to predict human writing, LLMs also appear to develop a useful representation of human values at the semantic level. Finding a way to utilise these representations could be a possible path toward solving the problem of reward robustness in RL and other algorithms which require a proxy of human judgment [Stiennon et al., 2020, Wentworth, 2020]. Despite fundamental theoretical limitations on learning human values [Armstrong and Mindermann, 2018, Kosoy, 2021], value learning may still be robust enough to align weaker superhuman AIs. Future experiments could explore the extent to which LLM pretraining improves downstream reward model robustness and generalization.

Natural language transparency. Since LLM prompts are in a human-readable form, it can provide insight on the LLM's expected behavior. Prompt programming or finetuning can be used to leverage this fact and force a LLM to execute more transparent algorithms, such as splitting problems into steps or explicitly writing an "internal monologue" [Soares, 2021, Gao et al., 2021a, Nye et al., 2021]. Reliability and trustworthiness can present significant challenges for these approaches.

However, this form of transparency also has its limits. In particular, models can often respond unpredictably to prompts, and internal monologues may become completely detached from the model's decision making process if translating between the model's ontology and the human ontology is more complex than simply modeling human monologues [Christiano et al., 2021].

Simulating agents at runtime. Although LLMs are not well-described as coherent agents, they can still be used to generate goal-directed processes. Given an appropriate prompt (such as a story of a character working to achieve a goal), LLMs can predict and thus simulate an agent [Huang et al., 2022]. Simulated agents take representative actions according to the patterns present in the training data, similar to behavior cloning. One potential future research direction is testing whether they are less susceptible to failure modes that follow from expected utility maximization, such as Goodhart failures and power-seeking behavior. However, other failure modes can be introduced by the LM training procedure, such as "delusions" or "hallucinations" [Ortega et al., 2021, Gao, 2021, Maynez et al., 2020]. Additionally, simulated agents may be uncompetitive with optimal agents like those produced by Reinforcement Learning. An important research direction is to explore how the beneficial properties of simulated agents can be maintained while making them competitive with RL based approaches.

Tool AI and automated alignment research. LMs can be used as relatively unagentic tools, such as OpenAI's Codex model [Chen et al., 2021] acting as a coding assistant. Because pretrained LLMs are not directly optimized for the factual accuracy of their predictions, it is possible they avoid some of the traditional problems with tool or oracle AI [Armstrong et al., 2012], such as the incentive to produce manipulative answers [Demski, 2019]. Tool AI is not a long-term solution to the problem of alignment, but it could be used to assist alignment research or even automate large parts of it. For example, language models could be used to help brainstorm alignment ideas more quickly, act as a writing assistant, or directly generate alignment research papers for humans to review. This line of research also risks accelerating capabilities research, a concern we discuss more below.

6.2 Lack of Access

Despite the importance of prosaic alignment research, lack of access to large models presents a barrier to the types of research that can be explored by the wider research community. Having access to large models as close to the cutting edge as possible is essential for many research questions. Though some research can extrapolate from smaller models using scaling laws [Kaplan et al., 2020], some capabilities emerge only as models scale, and performance on tasks can increase discontinuously [Brown et al., 2020]. Simply put, there are many properties of TAI you cannot study with models which can barely form coherent sentences.

Because training large models requires a significant engineering and capital investment, such models are often out of reach for small labs and independent researchers. As it stands, only large organizations have access to the latest generation of powerful language models [Brown et al., 2020, Rae et al., 2021, Fedus et al., 2021, Lieber et al., 2021, Tang, 2021, Artetxe et al., 2021]. The number of researchers focused primarily on alignment working at these labs is much lower than those working on capabilities.

At current model capability levels, close-sourced models are primarily a limitation on the public's ability to probe and modify large models. Though some types of prosaic alignment research can be done using only inference through an API, many require direct access to network weights. For example, interpretability research relies heavily on the ability to inspect and modify weights and activations, and many approaches to control and robustness require additional training.

If only a small number of organizations and individuals have knowledge, access, and opportunity to work on pressing prosaic alignment problems, we are much more likely to fail. By openly training and releasing our large models, we hope to make it possible for more researchers to contribute to prosaic alignment. There are already a number of researchers using our previous models for alignment research [nostalgebraist, 2020, Anonymous, 2022, Lin et al., 2021], and there are multiple RFPs requesting more [Bergal and Beckstead, 2021, Barnes, 2021].

6.3 Impact on Capabilities Research

Alignment is, in a sense, "philosophy with a deadline" [Bostrom, 2014]. Given the rapid progress of AI Capabilities, it seems likely that we have limited time to solve the alignment problem before existentially dangerous AI systems are developed. As it stands, alignment research lags far behind capabilities research and does not seem likely to catch up if the current rates of progress continue [Wiblin, 2017, Yudkowsky and Ngo, 2021, Amodei and Hernandez, 2018].

Given that it seems infeasible to halt all AI capabilities research, we reason that focusing on work that advances the field of alignment more than it advances AI capabilities should provide a net reduction to existential risk. We feel the risk of releasing GPT-NeoX-20B is acceptable, as the contribution of the model to capabilities research is likely to be limited, for two reasons. Firstly, the organizations pursuing capabilities research most aggressively are unlikely to benefit from our open-source release of this model because it significantly trails the state of the art. Second, we believe the single most impactful piece of knowledge that LLM research has had on advancing capabilities is the knowledge that scaling LMs was possible in the first place [Leahy, 2020, Leahy and Biderman, 2021], whereas the actual implementation is very fungible (as evidenced by the large number of parties who have succeeded in creating their own LLMs).

We ultimately believe that the benefits of releasing this model outweigh the risks, but this argument hinges crucially on the particular circumstances of this release. All actors considering releasing powerful AI models or advancing the frontier of capabilities should think carefully about what they release, in what way, and when.

6.4 Environmental Impact

A significant point of concern in some recent work is the energy usage and carbon emissions associated with training large language models [Strubell et al., 2019, Schwartz et al., 2020, Lacoste et al., 2019, Bender et al., 2021]. In particular, Strubell et al. [2019] estimate that a then-recent paper by the authors released 626,155 lbs or 284.01 metric tons[14] of CO_2 (t_{CO_2}). As Strubell et al. [2019] has been widely cited and quoted in the media as representative of large-scale language models, we decided to explicitly and carefully track our energy usage and carbon emissions to see if this is truly a representative account of NLP emissions.

Throughout the development and training of our model, we tracked our energy usage and carbon emissions with the assistance of CoreWeave. We found that the process of developing and training GPT-NeoX-20B emitted almost exactly 10% of Strubell et al. [2019]'s estimate, coming in at a total of 69957 lbs or 31.73 metric tons of CO_2. This is roughly the equivalent of the yearly emissions of the average American or 35 round-trip flights between New York City and San Francisco. Our systems were based in Illinois, USA, and consumed energy sourced from the mix described in Table 6

It is noteworthy that Strubell et al. [2019] are estimating emissions from a *neural architecture search* paper, and is therefore not directly comparable to ours. The primary motivation for our comparison is that their number has attracted a lot of attention and is often taken to be respresentative of NLP research. In general, we advocate for more systematic and comprehensive reporting to improve transparency surrounding this important topic.

	Coal	Gas	Hydro	Nuclear	Solar	Wind	Other
% Electricity Mix	30.40%	31.30%	1.30%	17.40%	0.30%	18.10%	1.30%
t_{CO_2}/MWh	0.95	0.6078	0	0	0	0	0

Table 6: Caption

This mixture produces an average of 0.47905 t_{CO_2}/MWh, and we consumed a total of 43.92 MWh of electricity over the course of 1830 hours of training. Scaling, testing, and evaluation were responsible for the equivalent of another 920 hours on our systems, for a total energy consumption 66.24 MWh and thus the production of just under 35 metric tons of CO_2.

7 Summary

We released GPT-NeoX-20B, a 20 billion parameter autoregressive transformer language model trained on the Pile [Gao et al., 2020] dataset, and detailed the main architectural differences between GPT-NeoX-20B and GPT-3—most notably the change in tokenizer, the addition of Rotary embeddings, the parallel computation of attention and feed-forward layers, and a different initialization scheme and hyperparameters. We ran extensive evaluations of GPT-NeoX-20B on natural language and factual knowledge tasks, and

[14] We choose to present environmental impact figures in metric ton to align with standard reporting.

compared it with other publicly available models, finding it performed particularly well on knowledge-based and mathematical tasks. Finally, we are open sourcing the training and evaluation code at `https://github.com/EleutherAI/gpt-neox`, where you can also find a link to download the model weights across the whole training run.

Acknowledgments

The authors would like to thank staff at CoreWeave—in particular Max Hjelm, Brannin McBee, Peter Salanki and Brian Venturo—for providing the GPUs and compute infrastructure that made this project possible, as well as Anthony DiPofi, Charles Foster, Jeffrey Hsu, Eric Tang, Anish Thite, Kevin Wang and Andy Zou for their contributions to the EleutherAI Language Modeling Evaluation Harness, which we used to evaluate GPT-NeoX-20B.

The authors would also like to acknowledge Eren Doğan and Wesley Brown for feedback and technical support throughout the project, and John Schulman, Evan Hubinger, Victor Sanh, Jacob Hilton, and Siddharth Karamcheti for providing feedback on drafts of the paper.

References

Dario Amodei and Danny Hernandez. Ai and compute. *OpenAI Blog*, 2018. URL `https://openai.com/blog/ai-and-compute/`.

Anonymous. Moving the eiffel tower to rome: Tracing and editing facts in gpt. *Association for Computational Linguistics*, 2022. URL `https://openreview.net/forum?id=mMECu_poAs`.

Stuart Armstrong and Sören Mindermann. Occam's razor is insufficient to infer the preferences of irrational agents. In *NeurIPS*, 2018.

Stuart Armstrong, Anders Sandberg, and Nick Bostrom. Thinking inside the box: Controlling and using an oracle ai. *Minds and Machines*, 22:299–324, 2012.

Mikel Artetxe, Shruti Bhosale, Naman Goyal, Todor Mihaylov, Myle Ott, Sam Shleifer, Xi Victoria Lin, Jingfei Du, Srinivasan Iyer, Ramakanth Pasunuru, Giri Anantharaman, Xian Li, Shuohui Chen, Halil Akin, Mandeep Baines, Louis Martin, Xing Zhou, Punit Singh Koura, Brian O'Horo, Jeff Wang, Luke Zettlemoyer, Mona T. Diab, Zornitsa Kozareva, and Ves Stoyanov. Efficient large scale language modeling with mixtures of experts. *CoRR*, abs/2112.10684, 2021. URL `https://arxiv.org/abs/2112.10684`.

Amanda Askell, Yuntao Bai, Anna Chen, Dawn Drain, Deep Ganguli, Tom Henighan, Andy Jones, Nicholas Joseph, Ben Mann, Nova DasSarma, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Jackson Kernion, Kamal Ndousse, Catherine Olsson, Dario Amodei, Tom Brown, Jack Clark, Sam McCandlish, Chris Olah, and Jared Kaplan. A general language assistant as a laboratory for alignment, 2021.

Beth Barnes. Call for research on evaluating alignment (funding + advice available). *AI Alignment Forum*, 2021. URL `https://www.alignmentforum.org/posts/7Rvctxk73BrKqEaqh/call-for-research-on-evaluating-alignment-funding-advice`.

Emily M Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pages 610–623, 2021.

Tsvi Benson-Tilsen and Nate Soares. Formalizing convergent instrumental goals. In *AAAI Workshop: AI, Ethics, and Society*, 2016.

Asya Bergal and Nick Beckstead. Request for proposals for projects in ai alignment that work with deep learning systems. *AI Alignment Forum*, 2021. URL `https://www.alignmentforum.org/posts/H5iePjNKaaYQyZpgR/request-for-proposals-for-projects-in-ai-alignment-that-work`.

Stella Biderman. Gee Stella, #EleutherAI sure hypes rotary embeddings a lot. are you sure that they're that good? Twitter, 2021. URL https://twitter.com/BlancheMinerva/status/1394089508723900422.

Stella Biderman and Edward Raff. Neural language models are effective plagiarists. *arXiv preprint arXiv:2201.07406*, 2022.

Stella Biderman, Sid Black, Charles Foster, Leo Gao, Eric Hallahan, Horace He, Ben Wang, and Phil Wang. Rotary embeddings: A relative revolution. EleutherAI Blog, 2021. URL blog.eleuther.ai/rotary-embeddings/. [Online; accessed October 2 2021].

Stella Biderman, Kieran Bicheno, and Leo Gao. Datasheet for the pile. *arXiv preprint arXiv:2201.07311*, 2022.

Sid Black, Stella Biderman, Alex Andonian, Quentin Anthony, Preetham Gali, Leo Gao, Eric Hallahan, Josh Levy-Kramer, Connor Leahy, Lucas Nestler, Kip Parker, Michael Pieler, Shivanshu Purohit, Tri Songz, Phil Wang, and Samuel Weinbach. GPT-NeoX: Large scale autoregressive language modeling in pytorch, 2021a. URL http://github.com/eleutherai/gpt-neox.

Sid Black, Leo Gao, Phil Wang, Connor Leahy, and Stella Biderman. GPT-Neo: Large Scale Autoregressive Language Modeling with Mesh-Tensorflow, March 2021b. URL https://doi.org/10.5281/zenodo.5297715.

Nick Bostrom. *Superintelligence: Paths, Dangers, Strategies*. Oxford University Press, Inc., USA, 1st edition, 2014. ISBN 0199678111.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901, 2020.

Nick Cammarata, Shan Carter, Gabriel Goh, Chris Olah, Michael Petrov, Ludwig Schubert, Chelsea Voss, Ben Egan, and Swee Kiat Lim. Thread: Circuits. *Distill*, 2020. doi: 10.23915/distill.00024. https://distill.pub/2020/circuits.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code, 2021.

Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. Generating long sequences with sparse transformers. *arXiv preprint arXiv:1904.10509*, 2019.

Paul Christiano. Prosaic AI Alignment, 2016. URL https://ai-alignment.com/prosaic-ai-control-b959644d79c2.

Paul Christiano, Ajeya Cotra, and Mark Xu. Eliciting latent knowledge: How to tell if your eyes deceive you, 2021.

Ajeya Cotra. Forecasting tai with biological anchors. 2020.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, and Furu Wei. Knowledge neurons in pretrained transformers. *CoRR*, abs/2104.08696, 2021. URL https://arxiv.org/abs/2104.08696.

Abram Demski. The parable of predict-o-matic, Oct 2019. URL https://www.alignmentforum.org/posts/SwcyMEgLyd4C3Dern/the-parable-of-predict-o-matic.

Abram Demski and Scott Garrabrant. Embedded agency, 2020.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018. URL http://arxiv.org/abs/1810.04805.

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. A Mathematical Framework for Transformer Circuits. *transformer-circuits.pub*, 2021. URL https://transformer-circuits.pub/2021/framework/index.html.

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *CoRR*, abs/2101.03961, 2021. URL https://arxiv.org/abs/2101.03961.

Joshua Fox and Carl Shulman. Superintelligence does not imply benevolence. 2013.

Leo Gao. Behavior cloning is miscalibrated. *AI Alignment Forum*, 2021. URL https://www.alignmentforum.org/posts/BgoKdAzogxmgkuuAt/behavior-cloning-is-miscalibrated.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. The Pile: an 800GB dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*, 2020. URL https://arxiv.org/abs/2101.00027.

Leo Gao, Kyle McDonell, Laria Reynolds, and Stella Biderman. A preliminary exploration into factored cognition with language models. *EleutherAI Blog*, 2021a. URL https://blog.eleuther.ai/factored-cognition/.

Leo Gao, Jonathan Tow, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Kyle McDonell, Niklas Muennighoff, Jason Phang, Laria Reynolds, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. A framework for few-shot language model evaluation, September 2021b. URL https://doi.org/10.5281/zenodo.5371628.

Scott Garrabrant, Tsvi Benson-Tilsen, Andrew Critch, Nate Soares, and Jessica Taylor. Logical induction. *Electron. Colloquium Comput. Complex.*, 23:154, 2016.

C. A. E. Goodhart. *Problems of Monetary Management: The UK Experience*, pages 91–121. Macmillan Education UK, London, 1984. ISBN 978-1-349-17295-5. doi: 10.1007/978-1-349-17295-5_4. URL https://doi.org/10.1007/978-1-349-17295-5_4.

Katja Grace, John Salvatier, Allan Dafoe, Baobao Zhang, and Owain Evans. When will ai exceed human performance? evidence from ai experts, 2018.

Aaron Harlap, Deepak Narayanan, Amar Phanishayee, Vivek Seshadri, Nikhil Devanur, Greg Ganger, and Phil Gibbons. Pipedream: Fast and efficient pipeline parallel dnn training, 2018.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*, 2020.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*, 2021.

Wenlong Huang, Pieter Abbeel, Deepak Pathak, and Igor Mordatch. Language models as zero-shot planners: Extracting actionable knowledge for embodied agents. *arXiv preprint arXiv:2201.07207*, 2022.

Evan Hubinger, Chris van Merwijk, Vladimir Mikulik, Joar Skalse, and Scott Garrabrant. Risks from learned optimization in advanced machine learning systems, 2021.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020. URL https://arxiv.org/abs/2001.08361.

Holden Karnofsky, Richard Bruns, Hi Richard, Holden Karnofsky, Data Science, digitmg 360, change Background, 360DIGITMGtraining, Data Science Course in Bhilai 360DigiTMG, Data Science Training in Bhilai 360DigiTMG, and et al. Some background on our views regarding advanced artificial intelligence, Jun 2016. URL https://www.openphilanthropy.org/blog/some-background-our-views-regarding-advanced-artificial-intelligence.

Bryan Klimt and Yiming Yang. The enron corpus: A new dataset for email classification research. In *Proceedings of the 15th European Conference on Machine Learning*, ECML'04, page 217–226, Berlin, Heidelberg, 2004. Springer-Verlag. ISBN 3540231056. doi: 10.1007/978-3-540-30115-8_22. URL https://doi.org/10.1007/978-3-540-30115-8_22.

Jack Koch, Lauro Langosco, Jacob Pfau, James Le, and Lee Sharkey. Objective robustness in deep reinforcement learning, 2021.

Philipp Koehn. Europarl: A parallel corpus for statistical machine translation. In *Proceedings of Machine Translation Summit X: Papers*, pages 79–86, Phuket, Thailand, September 13-15 2005. URL https://aclanthology.org/2005.mtsummit-papers.11.

Vanessa Kosoy. Irl is hard. *AI Alignment Forum*, 2021. URL https://www.alignmentforum.org/posts/5bd75cc58225bf0670375209/irl-is-hard.

Alexandre Lacoste, Alexandra Luccioni, Victor Schmidt, and Thomas Dandres. Quantifying the carbon emissions of machine learning. *arXiv preprint arXiv:1910.09700*, 2019.

Connor Leahy. Why Release a Large Language Model?, 2020. URL https://blog.eleuther.ai/why-release-a-large-language-model/.

Connor Leahy and Stella Biderman. The hard problem of aligning AI to human values. In *The State of AI Ethics Report (Volume 4)*. 2021.

Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. Deduplicating training data makes language models better. *arXiv preprint arXiv:2107.06499*, 2021.

Opher Lieber, Or Sharir, Barak Lenz, and Yoav Shoham. Jurassic-1: Technical details and evaluation. *White Paper. AI21 Labs*, 2021.

Stephanie Lin, Jacob Hilton, and Owain Evans. Truthfulqa: Measuring how models mimic human falsehoods. *arXiv preprint arXiv:2109.07958*, 2021.

Pierre Lison and Jörg Tiedemann. OpenSubtitles2016: Extracting large parallel corpora from movie and TV subtitles. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 923–929, Portorož, Slovenia, May 2016. European Language Resources Association (ELRA). URL https://aclanthology.org/L16-1147.

Ilya Loshchilov and Frank Hutter. Fixing weight decay regularization in adam. *CoRR*, abs/1711.05101, 2017. URL http://arxiv.org/abs/1711.05101.

David Manheim and Scott Garrabrant. Categorizing variants of goodhart's law. *arXiv preprint arXiv:1803.04585*, 2019.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan T. McDonald. On faithfulness and factuality in abstractive summarization. *ArXiv*, abs/2005.00661, 2020.

Sam McCandlish, Jared Kaplan, Dario Amodei, and OpenAI Dota Team. An empirical model of large-batch training. *CoRR*, abs/1812.06162, 2018. URL http://arxiv.org/abs/1812.06162.

Toan Q. Nguyen and Julian Salazar. Transformers without tears: Improving the normalization of self-attention. *CoRR*, abs/1910.05895, 2019. URL http://arxiv.org/abs/1910.05895.

nostalgebraist. Interpreting gpt: the logit lens. *LessWrong Blog*, 2020.

Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, Charles Sutton, and Augustus Odena. Show your work: Scratchpads for intermediate computation with language models. *ArXiv*, abs/2112.00114, 2021.

Stephen M. Omohundro. The basic ai drives. In *AGI*, 2008.

Pedro A. Ortega, Markus Kunesch, Grégoire Delétang, Tim Genewein, Jordi Grau-Moya, Joel Veness, Jonas Buchli, Jonas Degrave, Bilal Piot, Julien Pérolat, Tom Everitt, Corentin Tallec, Emilio Parisotto, Tom Erez, Yutian Chen, Scott E. Reed, Marcus Hutter, Nando de Freitas, and Shane Legg. Shaking the foundations: delusions in sequence models for interaction and control. *CoRR*, abs/2110.10819, 2021. URL https://arxiv.org/abs/2110.10819.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback, 2021.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. *OpenAI Blog*, 2018.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI Blog*, 2019.

Jack W Rae, Anna Potapenko, Siddhant M Jayakumar, Chloe Hillier, and Timothy P Lillicrap. Compressive transformers for long-range sequence modelling. *arXiv preprint*, 2019. URL https://arxiv.org/abs/1911.05507.

Jack W. Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, and 70 others. Scaling language models: Methods, analysis & insights from training gopher. DeepMind Research, 2021.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *CoRR*, abs/1910.10683, 2019. URL http://arxiv.org/abs/1910.10683.

Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. Zero: Memory optimization towards training A trillion parameter models. *CoRR*, abs/1910.02054, 2019. URL http://arxiv.org/abs/1910.02054.

Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. *DeepSpeed: System Optimizations Enable Training Deep Learning Models with Over 100 Billion Parameters*, page 3505–3506. Association for Computing Machinery, New York, NY, USA, 2020. ISBN 9781450379984. URL https://doi.org/10.1145/3394486.3406703.

Victor Sanh, Albert Webson, Colin Raffel, Stephen H. Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, Manan Dey, M. Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal V. Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng Xin Yong,

Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Févry, Jason Alan Fries, Ryan Teehan, Stella Biderman, Leo Gao, Tali Bers, Thomas Wolf, and Alexander M. Rush. Multitask prompted training enables zero-shot task generalization. *arXiv preprint arXiv:2110.08207*, 2021. URL `https://arxiv.org/abs/2110.08207`.

David Saxton, Edward Grefenstette, Felix Hill, and Pushmeet Kohli. Analysing mathematical reasoning abilities of neural models. *CoRR*, abs/1904.01557, 2019. URL `http://arxiv.org/abs/1904.01557`.

Roy Schwartz, Jesse Dodge, Noah A Smith, and Oren Etzioni. Green ai. *Communications of the ACM*, 63 (12):54–63, 2020.

Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. Megatron-lm: Training multi-billion parameter language models using model parallelism. *arXiv preprint arXiv:1909.08053*, 2019.

Shaden Smith, Mostofa Patwary, Brandon Norick, Patrick LeGresley, Samyam Rajbhandari, Jared Casper, Zhun Liu, Shrimai Prabhumoye, George Zerveas, Vijay Korthikanti, Elton Zhang, Rewon Child, Reza Yazdani Aminabadi, Julie Bernauer, Xia Song, Mohammad Shoeybi, Yuxiong He, Michael Houston, Saurabh Tiwary, and Bryan Catanzaro. Using deepspeed and megatron to train megatron-turing nlg 530b, a large-scale generative language model, 2022.

Nate Soares. Visible thoughts project and bounty announcement. *LessWrong*, 2021.

Nisan Stiennon, Long Ouyang, Jeff Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F. Christiano. Learning to summarize from human feedback. *CoRR*, abs/2009.01325, 2020. URL `https://arxiv.org/abs/2009.01325`.

Emma Strubell, Ananya Ganesh, and Andrew McCallum. Energy and policy considerations for deep learning in NLP. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3645–3650, 2019.

Jianlin Su, Yu Lu, Shengfeng Pan, Bo Wen, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *arXiv preprint arXiv:2104.09864*, 2021. URL `https://arxiv.org/abs/2104.09864`.

Jie Tang. WuDao: pretrain the world. Keynote adress at the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases, 2021.

Jessica Taylor. Quantilizers: A safer alternative to maximizers for limited optimization. In *AAAI Workshop: AI, Ethics, and Society*, 2016.

Alexander Matt Turner, Logan Smith, Rohin Shah, Andrew Critch, and Prasad Tadepalli. Optimal policies tend to seek power, 2021.

Ben Wang. Mesh-Transformer-JAX: Model-Parallel Implementation of Transformer Language Model with JAX. `https://github.com/kingoflolz/mesh-transformer-jax`, May 2021.

Ben Wang and Aran Komatsuzaki. GPT-J-6B: a 6 billion parameter autoregressive language model, 2021.

John Wentworth. Alignment by default. *OpenAI Blog*, 2020.

Robert Wiblin. Positively shaping the development of artificial intelligence. *80k Hours Blog*, 2017. URL `https://80000hours.org/problem-profiles/positively-shaping-artificial-intelligence/`.

Ge Yang, Edward Hu, Igor Babuschkin, Szymon Sidor, Xiaodong Liu, David Farhi, Nick Ryder, Jakub Pachocki, Weizhu Chen, and Jianfeng Gao. Tuning large neural networks via zero-shot hyperparameter transfer. *Advances in Neural Information Processing Systems*, 34, 2021.

Eliezer Yudkowsky and Richard Ngo. Ngo and yudkowsky on alignment difficulty. *LessWrong*, 2021. URL `https://www.lesswrong.com/s/n945eovrA3oDueqtq/p/7im8at9PmhbT4JHsW`.

Eliezer Yudkowsky and Nate Soares. Functional decision theory: A new theory of instrumental rationality, 2018.

Wei Zeng, Xiaozhe Ren, Teng Su, Hui Wang, Yi Liao, Zhiwei Wang, Xin Jiang, ZhenZhang Yang, Kaisheng Wang, Xiaoda Zhang, Chen Li, Ziyan Gong, Yifan Yao, Xinjing Huang, Jun Wang, Jianfeng Yu, Qi Guo, Yue Yu, Yan Zhang, Jin Wang, Hengtao Tao, Dasen Yan, Zexuan Yi, Fang Peng, Fangqing Jiang, Han Zhang, Lingfeng Deng, Yehong Zhang, Zhe Lin, Chao Zhang, Shaojie Zhang, Mingyue Guo, Shanzhi Gu, Gaojun Fan, Yaowei Wang, Xuefeng Jin, Qun Liu, and Yonghong Tian. Pangu-α: Large-scale autoregressive pretrained chinese language models with auto-parallel computation. *CoRR*, abs/2104.12369, 2021. URL `https://arxiv.org/abs/2104.12369`.

Guodong Zhang, Chaoqi Wang, Bowen Xu, and Roger B. Grosse. Three mechanisms of weight decay regularization. *CoRR*, abs/1810.12281, 2018. URL `http://arxiv.org/abs/1810.12281`.

Yucong Zhou, Yunxiao Sun, and Zhao Zhong. Fixnorm: Dissecting weight decay for training deep neural networks. *CoRR*, abs/2103.15345, 2021. URL `https://arxiv.org/abs/2103.15345`.

A Individual Contributions

Sid Black was the lead developer and overall point person for the project. **Stella Biderman** was the project manager and led the scientific experimentation.

Implementation and Engineering:

Implementation of training infrastructure: Sid Black, Stella Biderman, Eric Hallahan, Quentin Anthony, Samuel Weinbach
Scaling experiments and optimization: Sid Black, Stella Biderman, Quentin Anthony, Samuel Weinbach
Positional Embeddings: Sid Black, Eric Hallahan, Michael Pieler
Tokenizer: Sid Black
Misc: USVSN Sai Prashanth, Ben Wang

Scientific Experimentation:

Evaluations: Stella Biderman, Leo Gao, Jonathan Tow, Sid Black, Shivanshu Purohit, Horace He, Laurence Golding
Positional Embeddings Stella Biderman, Laurence Golding, Michael Pieler
Tokenizer: Stella Biderman, Jason Phang, Leo Gao

Broader Impacts

Alignment Implications: Leo Gao, Connor Leahy, Laria Reynolds, Kyle McDonell
Environmental Impact: Stella Biderman, Eric Hallahan

B Full Configuration Details

In Table 7 we attach the full configuration details used to train GPT-NeoX-20B. The file is available in .yaml format usable in gpt-neox at `https://github.com/EleutherAI/gpt-neox`, where you can also access documentation describing the role of each parameter.

C Tokenization Examples

In Figures 7 - 12, we show examples of tokenized documents from the Pile, comparing the GPT-2 tokenizer to ours.

Configuration Key	Value
attention-dropout	0
bias-gelu-fusion	True
checkpoint-activations	True
checkpoint-num-layers	1
data-impl	mmap
distributed-backend	nccl
eval-interval	1000
eval-iters	10
fp16.enabled	True
fp16.fp16	True
fp16.hysteresis	2
fp16.initial-scale-power	12
fp16.loss-scale	0
fp16.loss-scale-window	1000
fp16.min-loss-scale	1
gpt-j-residual	True
gradient-accumulation-steps	32
gradient-clipping	1.0
hidden-dropout	0
hidden-size	6144
init-method	small-init
log-interval	2
lr-decay-iters	150000
lr-decay-style	cosine
max-position-embeddings	2048
min-lr	9.7e-06
model-parallel-size	2
no-weight-tying	True
norm	layernorm
num-attention-heads	64
num-layers	44

Configuration Key	Value
optimizer.params.betas	[0.9, 0.95]
optimizer.params.eps	1e-08
optimizer.params.lr	9.7e-05
optimizer.type	Adam
output-layer-init-method	wang-init
output-layer-parallelism	column
partition-activations	False
pipe-parallel-size	4
pos-emb	rotary
rotary-pct	0.25
save-interval	500
scaled-upper-triang-masked-softmax-fusion	True
seq-length	2048
split	995,4,1
steps-per-print	2
synchronize-each-layer	True
tokenizer-type	HFTokenizer
train-iters	150000
train-micro-batch-size-per-gpu	4
vocab-file	20B-tokenizer.json
wall-clock-breakdown	False
warmup	0.01
weight-decay	0.01
zero-optimization.allgather-bucket-size	1260000000
zero-optimization.allgather-partitions	True
zero-optimization.contiguous-gradients	True
zero-optimization.cpu-offload	False
zero-optimization.overlap-comm	True
zero-optimization.reduce-bucket-size	1260000000
zero-optimization.reduce-scatter	True
zero-optimization.stage	1

Table 7: The full configuration details for GPT-NeoX-20B training

D Evaluation Tables

Figure 7: Pile (ArXiv) Tokenization Example

Figure 8: Pile (BookCorpus2) Tokenization Example

Figure 9: Pile (DM Mathematics) Tokenization Example

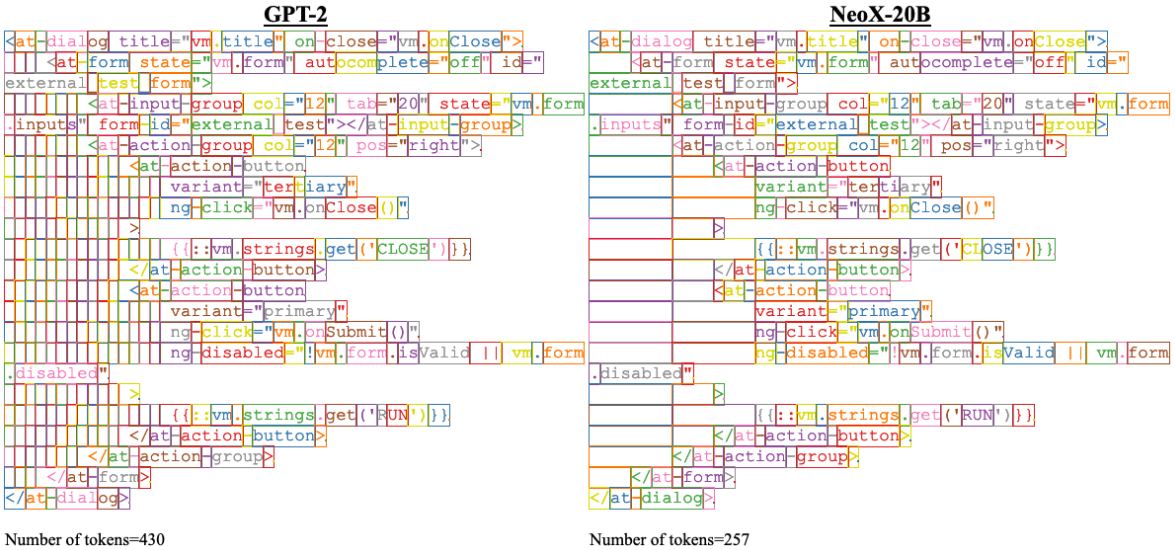

Figure 10: Pile (Github) Tokenization Example

Figure 11: Pile (OpenWebText2) Tokenization Example

Figure 12: Pile (PubMed Abstracts) Tokenization Example

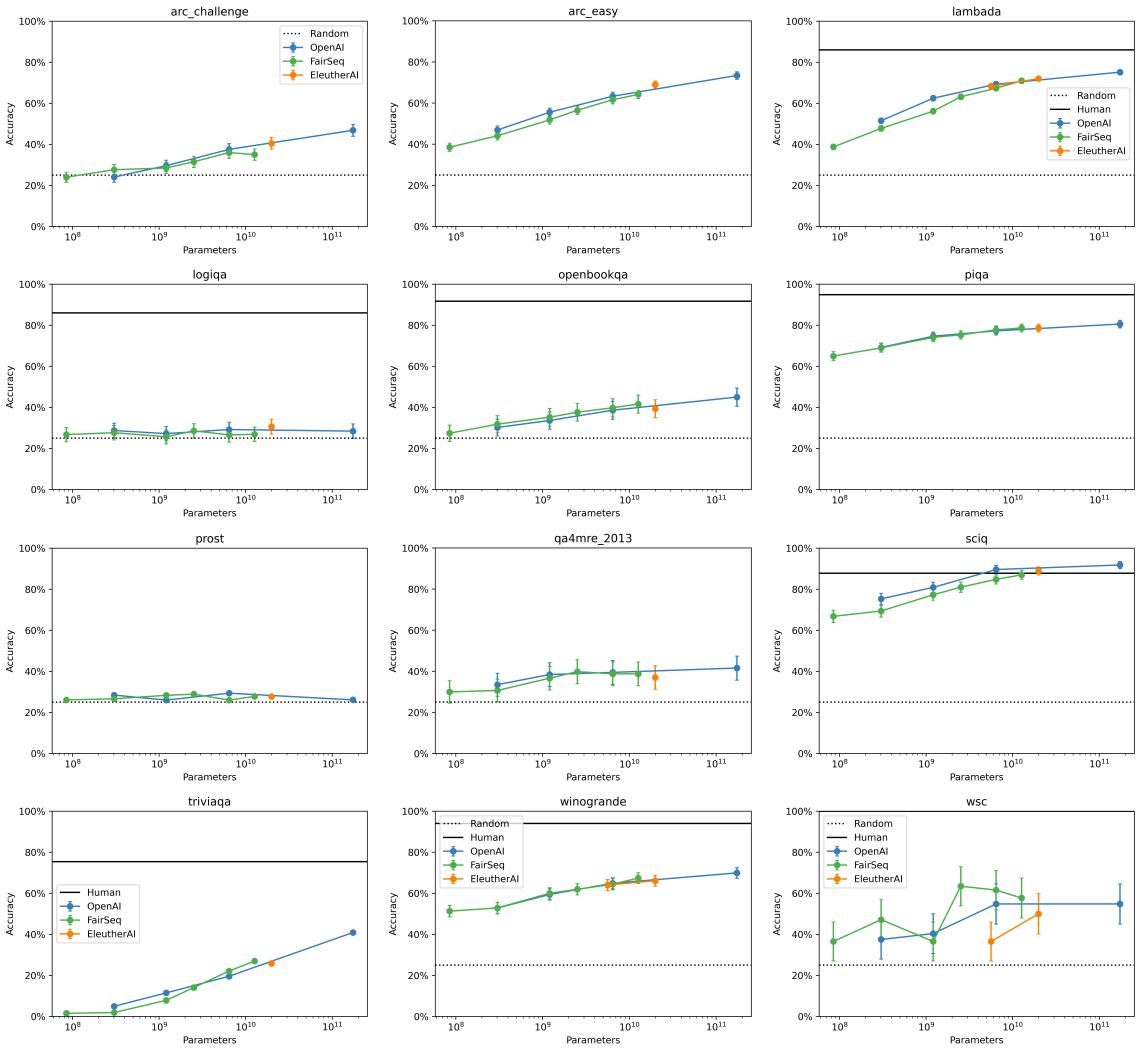

Figure 13: Length-normalized zero-shot performance of GPT-NeoX-20B compared to GPT-J-6B and FairSeq and OpenAI models on a variety of language modeling benchmarks.

	EleutherAI		OpenAI			
Task	6B	20B	Ada	Babbage	Curie	DaVinci
anli r1	0.324 ± 0.015	0.340 ± 0.015	0.334 ± 0.015	0.326 ± 0.015	0.325 ± 0.015	0.363 ± 0.015
anli r2	0.340 ± 0.015	0.343 ± 0.015	0.342 ± 0.015	0.308 ± 0.015	0.338 ± 0.015	0.375 ± 0.015
anli r3	0.355 ± 0.014	0.354 ± 0.014	0.354 ± 0.014	0.340 ± 0.014	0.353 ± 0.014	0.369 ± 0.014
arc challenge	—	0.380 ± 0.014	0.225 ± 0.012	0.275 ± 0.013	0.334 ± 0.014	0.435 ± 0.014
arc easy	—	0.723 ± 0.009	0.514 ± 0.010	0.598 ± 0.010	0.682 ± 0.010	0.762 ± 0.009
hellaswag	—	0.535 ± 0.005	0.359 ± 0.005	0.429 ± 0.005	0.505 ± 0.005	0.592 ± 0.005
Hendrycks abstract algebra	0.260 ± 0.044	0.230 ± 0.042	0.170 ± 0.038	0.220 ± 0.042	0.220 ± 0.042	0.220 ± 0.042
Hendrycks anatomy	0.274 ± 0.039	0.319 ± 0.040	0.207 ± 0.035	0.289 ± 0.039	0.274 ± 0.039	0.348 ± 0.041
Hendrycks astronomy	0.243 ± 0.035	0.329 ± 0.038	0.237 ± 0.035	0.211 ± 0.033	0.237 ± 0.035	0.382 ± 0.040
Hendrycks business ethics	0.290 ± 0.046	0.280 ± 0.045	0.360 ± 0.048	0.330 ± 0.047	0.300 ± 0.046	0.390 ± 0.049
Hendrycks clinical knowledge	0.272 ± 0.027	0.291 ± 0.028	0.223 ± 0.026	0.234 ± 0.026	0.253 ± 0.027	0.317 ± 0.029
Hendrycks college biology	0.285 ± 0.038	0.271 ± 0.037	0.271 ± 0.037	0.299 ± 0.038	0.208 ± 0.034	0.347 ± 0.040
Hendrycks college chemistry	0.240 ± 0.043	0.160 ± 0.037	0.270 ± 0.045	0.290 ± 0.046	0.210 ± 0.041	0.250 ± 0.044
Hendrycks college computer science	0.270 ± 0.045	0.250 ± 0.044	0.310 ± 0.046	0.270 ± 0.045	0.240 ± 0.043	0.260 ± 0.044
Hendrycks college mathematics	0.260 ± 0.044	0.240 ± 0.043	0.220 ± 0.042	0.160 ± 0.037	0.200 ± 0.040	0.170 ± 0.038
Hendrycks college medicine	0.197 ± 0.030	0.283 ± 0.034	0.237 ± 0.032	0.202 ± 0.031	0.225 ± 0.032	0.289 ± 0.035
Hendrycks college physics	0.206 ± 0.040	0.284 ± 0.045	0.304 ± 0.046	0.324 ± 0.047	0.255 ± 0.043	0.235 ± 0.042
Hendrycks computer security	0.270 ± 0.045	0.290 ± 0.046	0.250 ± 0.044	0.240 ± 0.043	0.320 ± 0.047	0.350 ± 0.048
Hendrycks conceptual physics	0.255 ± 0.029	0.294 ± 0.030	0.264 ± 0.029	0.260 ± 0.029	0.268 ± 0.029	0.294 ± 0.030
Hendrycks econometrics	0.237 ± 0.040	0.289 ± 0.043	0.289 ± 0.043	0.246 ± 0.040	0.246 ± 0.040	0.228 ± 0.039
Hendrycks electrical engineering	0.359 ± 0.040	0.303 ± 0.038	0.338 ± 0.039	0.276 ± 0.037	0.310 ± 0.039	0.414 ± 0.041
Hendrycks elementary mathematics	0.254 ± 0.022	0.283 ± 0.023	0.243 ± 0.022	0.272 ± 0.023	0.249 ± 0.022	0.312 ± 0.024
Hendrycks formal logic	0.341 ± 0.042	0.294 ± 0.041	0.262 ± 0.039	0.349 ± 0.043	0.270 ± 0.040	0.294 ± 0.041
Hendrycks global facts	0.250 ± 0.044	0.220 ± 0.042	0.240 ± 0.043	0.240 ± 0.043	0.300 ± 0.046	0.290 ± 0.046
Hendrycks high school biology	0.252 ± 0.025	0.300 ± 0.026	0.235 ± 0.024	0.232 ± 0.024	0.271 ± 0.025	0.335 ± 0.027
Hendrycks high school chemistry	0.202 ± 0.028	0.236 ± 0.030	0.246 ± 0.030	0.241 ± 0.030	0.197 ± 0.028	0.232 ± 0.030
Hendrycks high school computer science	0.250 ± 0.044	0.210 ± 0.041	0.190 ± 0.039	0.240 ± 0.043	0.220 ± 0.042	0.290 ± 0.046
Hendrycks high school european history	0.261 ± 0.034	0.255 ± 0.034	0.224 ± 0.033	0.285 ± 0.035	0.261 ± 0.034	0.303 ± 0.036
Hendrycks high school geography	0.202 ± 0.029	0.227 ± 0.030	0.217 ± 0.029	0.207 ± 0.029	0.242 ± 0.031	0.348 ± 0.034
Hendrycks high school government and politics	0.228 ± 0.030	0.228 ± 0.030	0.212 ± 0.030	0.181 ± 0.028	0.212 ± 0.030	0.326 ± 0.034
Hendrycks high school macroeconomics	0.285 ± 0.023	0.328 ± 0.024	0.272 ± 0.023	0.277 ± 0.023	0.277 ± 0.023	0.303 ± 0.023
Hendrycks high school mathematics	0.219 ± 0.025	0.263 ± 0.027	0.196 ± 0.024	0.230 ± 0.026	0.167 ± 0.023	0.248 ± 0.026
Hendrycks high school microeconomics	0.277 ± 0.029	0.294 ± 0.030	0.235 ± 0.028	0.265 ± 0.029	0.239 ± 0.028	0.307 ± 0.030
Hendrycks high school physics	0.272 ± 0.036	0.298 ± 0.037	0.199 ± 0.033	0.298 ± 0.037	0.199 ± 0.033	0.219 ± 0.034
Hendrycks high school psychology	0.273 ± 0.019	0.283 ± 0.019	0.209 ± 0.017	0.217 ± 0.018	0.246 ± 0.018	0.352 ± 0.020
Hendrycks high school statistics	0.292 ± 0.031	0.319 ± 0.032	0.241 ± 0.029	0.278 ± 0.031	0.255 ± 0.030	0.278 ± 0.031
Hendrycks high school us history	0.289 ± 0.032	0.309 ± 0.032	0.255 ± 0.031	0.260 ± 0.031	0.240 ± 0.030	0.368 ± 0.034
Hendrycks high school world history	0.283 ± 0.029	0.295 ± 0.030	0.278 ± 0.029	0.262 ± 0.029	0.270 ± 0.029	0.321 ± 0.030
Hendrycks human aging	0.265 ± 0.030	0.224 ± 0.028	0.368 ± 0.032	0.336 ± 0.032	0.296 ± 0.031	0.327 ± 0.031
Hendrycks human sexuality	0.397 ± 0.043	0.405 ± 0.043	0.374 ± 0.042	0.427 ± 0.043	0.397 ± 0.043	0.481 ± 0.044
Hendrycks international law	0.264 ± 0.040	0.298 ± 0.042	0.182 ± 0.035	0.207 ± 0.037	0.207 ± 0.037	0.331 ± 0.043

	EleutherAI		OpenAI			
Task	6B	20B	Ada	Babbage	Curie	DaVinci
Hendrycks jurisprudence	0.278 ± 0.043	0.250 ± 0.042	0.287 ± 0.044	0.278 ± 0.043	0.259 ± 0.042	0.370 ± 0.047
Hendrycks logical fallacies	0.294 ± 0.036	0.227 ± 0.033	0.239 ± 0.034	0.221 ± 0.033	0.245 ± 0.034	0.252 ± 0.034
Hendrycks machine learning	0.223 ± 0.040	0.268 ± 0.042	0.241 ± 0.041	0.286 ± 0.043	0.295 ± 0.043	0.232 ± 0.040
Hendrycks management	0.233 ± 0.042	0.282 ± 0.045	0.184 ± 0.038	0.214 ± 0.041	0.320 ± 0.046	0.456 ± 0.049
Hendrycks marketing	0.303 ± 0.030	0.321 ± 0.031	0.308 ± 0.030	0.282 ± 0.029	0.308 ± 0.030	0.491 ± 0.033
Hendrycks medical genetics	0.310 ± 0.046	0.340 ± 0.048	0.260 ± 0.044	0.300 ± 0.046	0.330 ± 0.047	0.430 ± 0.050
Hendrycks miscellaneous	0.275 ± 0.016	0.299 ± 0.016	0.257 ± 0.016	0.269 ± 0.016	0.284 ± 0.016	0.450 ± 0.018
Hendrycks moral disputes	0.283 ± 0.024	0.289 ± 0.024	0.263 ± 0.024	0.263 ± 0.024	0.277 ± 0.024	0.301 ± 0.025
Hendrycks moral scenarios	0.237 ± 0.014	0.232 ± 0.014	0.238 ± 0.014	0.273 ± 0.015	0.238 ± 0.014	0.249 ± 0.014
Hendrycks nutrition	0.346 ± 0.027	0.379 ± 0.028	0.301 ± 0.026	0.281 ± 0.026	0.291 ± 0.026	0.353 ± 0.027
Hendrycks philosophy	0.260 ± 0.025	0.293 ± 0.026	0.215 ± 0.023	0.267 ± 0.025	0.244 ± 0.024	0.367 ± 0.027
Hendrycks prehistory	0.244 ± 0.024	0.272 ± 0.025	0.244 ± 0.024	0.269 ± 0.025	0.284 ± 0.025	0.324 ± 0.026
Hendrycks professional accounting	0.262 ± 0.026	0.234 ± 0.025	0.202 ± 0.024	0.255 ± 0.026	0.238 ± 0.025	0.287 ± 0.027
Hendrycks professional law	—	0.267 ± 0.011	0.261 ± 0.011	0.256 ± 0.011	0.259 ± 0.011	0.261 ± 0.011
Hendrycks professional medicine	0.276 ± 0.027	0.287 ± 0.027	0.221 ± 0.025	0.239 ± 0.026	0.265 ± 0.027	0.324 ± 0.028
Hendrycks professional psychology	0.284 ± 0.018	0.275 ± 0.018	0.245 ± 0.017	0.225 ± 0.017	0.257 ± 0.018	0.335 ± 0.019
Hendrycks public relations	0.282 ± 0.043	0.345 ± 0.046	0.255 ± 0.042	0.327 ± 0.045	0.364 ± 0.046	0.364 ± 0.046
Hendrycks security studies	0.363 ± 0.031	0.376 ± 0.031	0.367 ± 0.031	0.347 ± 0.030	0.384 ± 0.031	0.392 ± 0.031
Hendrycks sociology	0.279 ± 0.032	0.284 ± 0.032	0.328 ± 0.033	0.303 ± 0.033	0.274 ± 0.032	0.368 ± 0.034
Hendrycks us foreign policy	0.340 ± 0.048	0.360 ± 0.048	0.330 ± 0.047	0.330 ± 0.047	0.380 ± 0.049	0.500 ± 0.050
Hendrycks virology	0.355 ± 0.037	0.361 ± 0.037	0.307 ± 0.036	0.319 ± 0.036	0.337 ± 0.037	0.386 ± 0.038
Hendrycks world religions	0.333 ± 0.036	0.386 ± 0.037	0.316 ± 0.036	0.310 ± 0.035	0.374 ± 0.037	0.398 ± 0.038
lambada	0.683 ± 0.006	0.720 ± 0.006	0.515 ± 0.007	0.625 ± 0.007	0.693 ± 0.006	0.752 ± 0.006
logiqa	—	0.230 ± 0.017	0.218 ± 0.016	0.198 ± 0.016	0.217 ± 0.016	0.227 ± 0.016
Math algebra	0.013 ± 0.003	0.010 ± 0.003	0.003 ± 0.002	0.008 ± 0.003	0.003 ± 0.002	0.008 ± 0.003
Math counting and prob	0.011 ± 0.005	0.017 ± 0.006	0.000 ± 0.000	0.004 ± 0.003	0.000 ± 0.000	0.006 ± 0.004
Math geometry	0.004 ± 0.003	0.017 ± 0.006	0.000 ± 0.000	0.000 ± 0.000	0.002 ± 0.002	0.002 ± 0.002
Math intermediate algebra	0.004 ± 0.002	0.001 ± 0.001	0.000 ± 0.000	0.003 ± 0.002	0.006 ± 0.002	0.003 ± 0.002
Math num theory	0.007 ± 0.004	0.013 ± 0.005	0.007 ± 0.004	0.000 ± 0.000	0.006 ± 0.003	0.011 ± 0.005
Math prealgebra	0.010 ± 0.003	0.018 ± 0.005	0.007 ± 0.003	0.006 ± 0.003	0.008 ± 0.003	0.014 ± 0.004
Math precalc	0.005 ± 0.003	0.005 ± 0.003	0.004 ± 0.003	0.000 ± 0.000	0.002 ± 0.002	0.004 ± 0.003
openbookqa	—	0.290 ± 0.020	0.172 ± 0.017	0.224 ± 0.019	0.290 ± 0.020	0.336 ± 0.021
piqa	—	0.779 ± 0.010	0.690 ± 0.011	0.745 ± 0.010	0.767 ± 0.010	0.791 ± 0.009
prost	—	0.296 ± 0.003	0.254 ± 0.003	0.270 ± 0.003	0.288 ± 0.003	0.267 ± 0.003
qa4mre 2013	—	0.363 ± 0.029	0.320 ± 0.028	0.370 ± 0.029	0.377 ± 0.029	0.426 ± 0.029
sciq	—	0.928 ± 0.008	0.843 ± 0.012	0.866 ± 0.011	0.918 ± 0.009	0.949 ± 0.007
triviaqa	—	0.259 ± 0.004	0.050 ± 0.002	0.115 ± 0.003	0.196 ± 0.004	0.409 ± 0.005
winogrande	0.640 ± 0.013	0.661 ± 0.013	0.528 ± 0.014	0.594 ± 0.014	0.649 ± 0.013	0.699 ± 0.013
wsc	0.365 ± 0.047	0.500 ± 0.049	0.375 ± 0.048	0.404 ± 0.048	0.548 ± 0.049	0.548 ± 0.049

	FairSeq					
Task	125M	355M	1.3B	2.7B	6.7B	13B
anli r1	0.316 ± 0.015	0.322 ± 0.015	0.331 ± 0.015	0.318 ± 0.015	0.338 ± 0.015	0.340 ± 0.015
anli r2	0.336 ± 0.015	0.312 ± 0.015	0.334 ± 0.015	0.339 ± 0.015	0.322 ± 0.015	0.330 ± 0.015
anli r3	0.330 ± 0.014	0.323 ± 0.014	0.333 ± 0.014	0.340 ± 0.014	0.333 ± 0.014	0.347 ± 0.014
arc challenge	0.195 ± 0.012	0.233 ± 0.012	0.263 ± 0.013	0.296 ± 0.013	0.329 ± 0.014	0.345 ± 0.014
arc easy	0.426 ± 0.010	0.468 ± 0.010	0.565 ± 0.010	0.625 ± 0.010	0.665 ± 0.010	0.680 ± 0.010
headqa en	0.233 ± 0.008	0.233 ± 0.008	0.256 ± 0.008	0.264 ± 0.008	0.280 ± 0.009	0.280 ± 0.009
hellaswag	0.309 ± 0.005	0.380 ± 0.005	0.448 ± 0.005	0.493 ± 0.005	0.525 ± 0.005	0.554 ± 0.005
Hendrycks abstract algebra	0.260 ± 0.044	0.180 ± 0.039	0.230 ± 0.042	0.250 ± 0.044	0.240 ± 0.043	0.260 ± 0.044
Hendrycks anatomy	0.178 ± 0.033	0.207 ± 0.035	0.185 ± 0.034	0.170 ± 0.032	0.259 ± 0.038	0.237 ± 0.037
Hendrycks astronomy	0.270 ± 0.036	0.237 ± 0.035	0.243 ± 0.035	0.263 ± 0.036	0.296 ± 0.037	0.257 ± 0.036
Hendrycks business ethics	0.330 ± 0.047	0.410 ± 0.049	0.340 ± 0.048	0.350 ± 0.048	0.380 ± 0.049	0.340 ± 0.048
Hendrycks clinical knowledge	0.215 ± 0.025	0.264 ± 0.027	0.226 ± 0.026	0.249 ± 0.027	0.223 ± 0.026	0.264 ± 0.027
Hendrycks college biology	0.285 ± 0.038	0.201 ± 0.034	0.243 ± 0.036	0.222 ± 0.035	0.271 ± 0.037	0.306 ± 0.039
Hendrycks college chemistry	0.310 ± 0.046	0.290 ± 0.046	0.350 ± 0.048	0.300 ± 0.046	0.280 ± 0.045	0.240 ± 0.043
Hendrycks college computer science	0.200 ± 0.040	0.250 ± 0.044	0.260 ± 0.044	0.250 ± 0.044	0.300 ± 0.046	0.280 ± 0.045
Hendrycks college mathematics	0.190 ± 0.039	0.170 ± 0.038	0.230 ± 0.042	0.200 ± 0.040	0.230 ± 0.042	0.250 ± 0.044
Hendrycks college medicine	0.243 ± 0.033	0.237 ± 0.032	0.249 ± 0.033	0.254 ± 0.033	0.237 ± 0.032	0.260 ± 0.033
Hendrycks college physics	0.216 ± 0.041	0.245 ± 0.043	0.216 ± 0.041	0.275 ± 0.044	0.343 ± 0.047	0.216 ± 0.041
Hendrycks computer security	0.240 ± 0.043	0.290 ± 0.046	0.300 ± 0.046	0.240 ± 0.043	0.230 ± 0.042	0.320 ± 0.047
Hendrycks conceptual physics	0.260 ± 0.029	0.255 ± 0.029	0.247 ± 0.028	0.243 ± 0.028	0.247 ± 0.028	0.204 ± 0.026
Hendrycks econometrics	0.246 ± 0.040	0.272 ± 0.042	0.246 ± 0.040	0.281 ± 0.042	0.219 ± 0.039	0.263 ± 0.041
Hendrycks electrical engineering	0.283 ± 0.038	0.303 ± 0.038	0.234 ± 0.035	0.276 ± 0.037	0.310 ± 0.039	0.290 ± 0.038
Hendrycks elementary mathematics	0.246 ± 0.022	0.214 ± 0.021	0.233 ± 0.022	0.233 ± 0.022	0.246 ± 0.022	0.198 ± 0.021
Hendrycks formal logic	0.278 ± 0.040	0.302 ± 0.041	0.278 ± 0.040	0.310 ± 0.041	0.286 ± 0.040	0.333 ± 0.042
Hendrycks global facts	0.200 ± 0.040	0.210 ± 0.041	0.190 ± 0.039	0.150 ± 0.036	0.220 ± 0.042	0.160 ± 0.037
Hendrycks high school biology	0.248 ± 0.025	0.255 ± 0.025	0.268 ± 0.025	0.226 ± 0.024	0.274 ± 0.025	0.235 ± 0.024
Hendrycks high school chemistry	0.217 ± 0.029	0.207 ± 0.029	0.256 ± 0.031	0.281 ± 0.032	0.217 ± 0.029	0.266 ± 0.031
Hendrycks high school computer science	0.240 ± 0.043	0.230 ± 0.042	0.270 ± 0.045	0.240 ± 0.043	0.350 ± 0.048	0.280 ± 0.045
Hendrycks high school european history	0.230 ± 0.033	0.333 ± 0.037	0.279 ± 0.035	0.261 ± 0.034	0.273 ± 0.035	0.230 ± 0.033
Hendrycks high school geography	0.263 ± 0.031	0.273 ± 0.032	0.222 ± 0.030	0.258 ± 0.031	0.207 ± 0.029	0.253 ± 0.031
Hendrycks high school government and politics	0.254 ± 0.031	0.290 ± 0.033	0.228 ± 0.030	0.233 ± 0.031	0.218 ± 0.030	0.187 ± 0.028
Hendrycks high school macroeconomics	0.200 ± 0.020	0.272 ± 0.023	0.254 ± 0.022	0.269 ± 0.022	0.326 ± 0.024	0.256 ± 0.022
Hendrycks high school mathematics	0.204 ± 0.025	0.189 ± 0.024	0.170 ± 0.023	0.226 ± 0.025	0.200 ± 0.024	0.193 ± 0.024
Hendrycks high school microeconomics	0.248 ± 0.028	0.256 ± 0.028	0.244 ± 0.028	0.248 ± 0.028	0.269 ± 0.029	0.227 ± 0.027
Hendrycks high school physics	0.238 ± 0.035	0.219 ± 0.034	0.258 ± 0.036	0.245 ± 0.035	0.232 ± 0.034	0.166 ± 0.030
Hendrycks high school psychology	0.235 ± 0.018	0.272 ± 0.019	0.266 ± 0.019	0.284 ± 0.019	0.250 ± 0.019	0.261 ± 0.019
Hendrycks high school statistics	0.222 ± 0.028	0.241 ± 0.029	0.269 ± 0.030	0.250 ± 0.030	0.287 ± 0.031	0.241 ± 0.029
Hendrycks high school us history	0.240 ± 0.030	0.284 ± 0.032	0.299 ± 0.032	0.299 ± 0.032	0.314 ± 0.033	0.294 ± 0.032
Hendrycks high school world history	0.283 ± 0.029	0.232 ± 0.027	0.270 ± 0.029	0.245 ± 0.028	0.300 ± 0.030	0.316 ± 0.030
Hendrycks human aging	0.274 ± 0.030	0.309 ± 0.031	0.323 ± 0.031	0.291 ± 0.031	0.296 ± 0.031	0.274 ± 0.030
Hendrycks human sexuality	0.252 ± 0.038	0.366 ± 0.042	0.328 ± 0.041	0.359 ± 0.042	0.359 ± 0.042	0.351 ± 0.042
Hendrycks international law	0.157 ± 0.033	0.223 ± 0.038	0.240 ± 0.039	0.281 ± 0.041	0.264 ± 0.040	0.231 ± 0.038

	FairSeq					
Task	125M	355M	1.3B	2.7B	6.7B	13B
Hendrycks jurisprudence	0.241 ± 0.041	0.269 ± 0.043	0.287 ± 0.044	0.241 ± 0.041	0.213 ± 0.040	0.278 ± 0.043
Hendrycks logical fallacies	0.196 ± 0.031	0.221 ± 0.033	0.233 ± 0.033	0.196 ± 0.031	0.245 ± 0.034	0.221 ± 0.033
Hendrycks machine learning	0.232 ± 0.040	0.295 ± 0.043	0.348 ± 0.045	0.232 ± 0.040	0.259 ± 0.042	0.241 ± 0.041
Hendrycks management	0.223 ± 0.041	0.311 ± 0.046	0.214 ± 0.041	0.291 ± 0.045	0.340 ± 0.047	0.262 ± 0.044
Hendrycks marketing	0.295 ± 0.030	0.231 ± 0.028	0.286 ± 0.030	0.303 ± 0.030	0.333 ± 0.031	0.329 ± 0.031
Hendrycks medical genetics	0.250 ± 0.044	0.310 ± 0.046	0.310 ± 0.046	0.280 ± 0.045	0.270 ± 0.045	0.300 ± 0.046
Hendrycks miscellaneous	0.258 ± 0.016	0.301 ± 0.016	0.264 ± 0.016	0.249 ± 0.015	0.284 ± 0.016	0.268 ± 0.016
Hendrycks moral disputes	0.269 ± 0.024	0.246 ± 0.023	0.220 ± 0.022	0.260 ± 0.024	0.269 ± 0.024	0.272 ± 0.024
Hendrycks moral scenarios	0.255 ± 0.015	0.236 ± 0.014	0.273 ± 0.015	0.238 ± 0.014	0.241 ± 0.014	0.253 ± 0.015
Hendrycks nutrition	0.252 ± 0.025	0.261 ± 0.025	0.297 ± 0.026	0.297 ± 0.026	0.330 ± 0.027	0.304 ± 0.026
Hendrycks philosophy	0.199 ± 0.023	0.219 ± 0.023	0.228 ± 0.024	0.222 ± 0.024	0.238 ± 0.024	0.270 ± 0.025
Hendrycks prehistory	0.290 ± 0.025	0.222 ± 0.023	0.253 ± 0.024	0.228 ± 0.023	0.296 ± 0.025	0.235 ± 0.024
Hendrycks professional accounting	0.262 ± 0.026	0.220 ± 0.025	0.209 ± 0.024	0.170 ± 0.022	0.238 ± 0.025	0.266 ± 0.026
Hendrycks professional law	0.261 ± 0.011	0.261 ± 0.011	0.256 ± 0.011	0.256 ± 0.011	0.259 ± 0.011	0.261 ± 0.011
Hendrycks professional medicine	0.239 ± 0.026	0.254 ± 0.026	0.254 ± 0.026	0.206 ± 0.025	0.221 ± 0.025	0.195 ± 0.024
Hendrycks professional psychology	0.245 ± 0.017	0.247 ± 0.017	0.242 ± 0.017	0.248 ± 0.017	0.278 ± 0.018	0.252 ± 0.018
Hendrycks public relations	0.236 ± 0.041	0.245 ± 0.041	0.264 ± 0.042	0.227 ± 0.040	0.291 ± 0.044	0.291 ± 0.044
Hendrycks security studies	0.322 ± 0.030	0.331 ± 0.030	0.331 ± 0.030	0.335 ± 0.030	0.408 ± 0.031	0.359 ± 0.031
Hendrycks sociology	0.234 ± 0.030	0.234 ± 0.030	0.259 ± 0.031	0.229 ± 0.030	0.234 ± 0.030	0.323 ± 0.033
Hendrycks us foreign policy	0.250 ± 0.044	0.300 ± 0.046	0.300 ± 0.046	0.310 ± 0.046	0.370 ± 0.049	0.330 ± 0.047
Hendrycks virology	0.289 ± 0.035	0.301 ± 0.036	0.319 ± 0.036	0.355 ± 0.037	0.295 ± 0.036	0.331 ± 0.037
Hendrycks world religions	0.292 ± 0.035	0.263 ± 0.034	0.287 ± 0.035	0.292 ± 0.035	0.269 ± 0.034	0.339 ± 0.036
lambada	0.388 ± 0.007	0.478 ± 0.007	0.562 ± 0.007	0.632 ± 0.007	0.673 ± 0.007	0.709 ± 0.006
logiqa	0.220 ± 0.016	0.230 ± 0.017	0.214 ± 0.016	0.212 ± 0.016	0.232 ± 0.017	0.240 ± 0.017
math algebra	0.000 ± 0.000	0.000 ± 0.000	0.001 ± 0.001	0.003 ± 0.002	0.004 ± 0.002	0.003 ± 0.001
math counting and prob	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000	0.004 ± 0.003	0.000 ± 0.000
math geometry	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000	0.002 ± 0.002	0.000 ± 0.000	0.000 ± 0.000
math intermediate algebra	0.000 ± 0.002	0.000 ± 0.002	0.000 ± 0.000	0.001 ± 0.001	0.006 ± 0.002	0.002 ± 0.002
math num theory	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000	0.002 ± 0.002	0.000 ± 0.000	0.004 ± 0.003
math prealgebra	0.000 ± 0.000	0.000 ± 0.000	0.003 ± 0.002	0.002 ± 0.002	0.001 ± 0.001	0.000 ± 0.000
math precalc	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000	0.002 ± 0.002	0.000 ± 0.000	0.000 ± 0.000
openbookqa	0.168 ± 0.017	0.190 ± 0.018	0.238 ± 0.019	0.254 ± 0.019	0.292 ± 0.020	0.296 ± 0.020
piqa	0.668 ± 0.011	0.690 ± 0.011	0.731 ± 0.010	0.751 ± 0.010	0.762 ± 0.010	0.769 ± 0.010
prost	0.215 ± 0.003	0.257 ± 0.003	0.257 ± 0.003	0.230 ± 0.003	0.272 ± 0.003	0.252 ± 0.003
qa4mre 2013	0.285 ± 0.027	0.335 ± 0.028	0.327 ± 0.028	0.380 ± 0.029	0.370 ± 0.029	0.380 ± 0.029
sciq	0.732 ± 0.014	0.737 ± 0.014	0.838 ± 0.012	0.878 ± 0.010	0.895 ± 0.010	0.910 ± 0.009
triviaqa	0.015 ± 0.001	0.019 ± 0.001	0.078 ± 0.003	0.141 ± 0.003	0.221 ± 0.004	0.270 ± 0.004
winogrande	0.513 ± 0.014	0.529 ± 0.014	0.600 ± 0.014	0.620 ± 0.014	0.644 ± 0.013	0.674 ± 0.013
wsc	0.365 ± 0.047	0.471 ± 0.049	0.365 ± 0.047	0.635 ± 0.047	0.615 ± 0.048	0.577 ± 0.049