GPT4All-J: An Apache-2 Licensed Assistant-Style Chatbot

Yuvanesh Anand
yuvanesh@nomic.ai

Zach Nussbaum
zach@nomic.ai

Brandon Duderstadt
brandon@nomic.ai

Benjamin M. Schmidt
ben@nomic.ai

Adam Treat
treat.adam@gmail.com

Andriy Mulyar
andriy@nomic.ai

Abstract

GPT4All-J is an Apache-2 licensed chatbot trained over a massive curated corpus of assistant interactions including word problems, multi-turn dialogue, code, poems, songs, and stories. It builds on the March 2023 GPT4All release by training on a significantly larger corpus, by deriving its weights from the Apache-licensed GPT-J model rather than the GPL-licensed of LLaMA, and by demonstrating improved performance on creative tasks such as writing stories, poems, songs and plays. We openly release the training data, data curation procedure, training code, and final model weights to promote open research and reproducibility. Additionally, we release Python bindings and a Chat UI to a quantized 4-bit version of GPT4All-J allowing virtually anyone to run the model on CPU.

1 Data Collection and Curation

We gather a diverse sample of questions/prompts by leveraging several publicly available datasets and curating our own set of prompts:

- Several subsamples from subsets of LAION OIG including unified_chip2, unified_unifiedskg_instruction, unified_hc3_human, unified_multi_news and unified_abstract_infill
- Coding questions with a random sub-sample of Stackoverflow Questions
- Instruction-tuning with a sub-sample of Bigscience/P3
- Custom-generated creative questions.

We accompany this paper with the 800k point GPT4All-J dataset that is a superset of the original 400k points GPT4All dataset. We dedicated substantial attention to data preparation and curation.

Building on the GPT4All dataset, we curated the GPT4All-J dataset by augmenting the original 400k GPT4All examples with new samples encompassing additional multi-turn QA samples and creative writing such as poetry, rap, and short stories. We designed prompt templates to create different scenarios for creative writing. The creative prompt template was inspired by Mad Libs style variations of 'Write a [creative story type] about [NOUN] in the style of [PERSON]'. In earlier versions of GPT4All, we found that rather than writing actual creative content, the model would discuss how it would go about writing the content. Training on this new dataset allows GPT4All-J to write poems, songs, and plays with increased competence.

We used Atlas to inform our data cleaning and curation efforts. We started with a collection of approximately 1,000,000 points. Several data curation iterations produced our final GPT4All-J training set. Among other changes, we removed exact duplicate prompts and responses characterized by homogeneous clusters in the Atlas map. We also removed prompts that were less than 10 characters such as single words like 'The', 'And', as well as poorly formatted examples.

Interactively explore the cleaned dataset in Atlas:

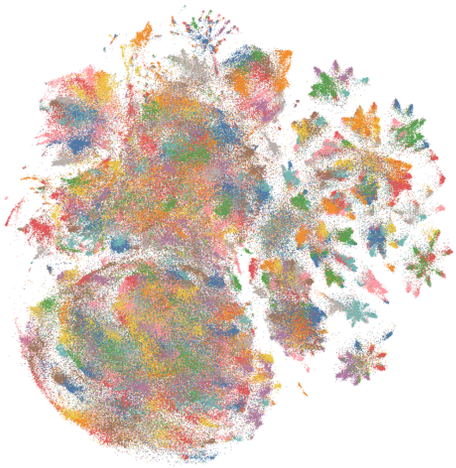
- GPT4All-J Curated Training Set Map

2 Model Training

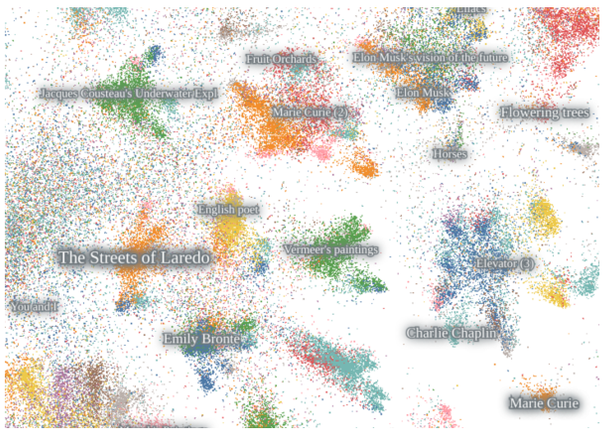
We trained several models finetuned from both LLaMA 7B (Touvron et al., 2023) and GPT-J (Wang and Komatsuzaki, 2021) checkpoints. The model associated with our initial public release is trained with LoRA (Hu et al., 2021) on the 437,605 post-processed examples for four epochs while the finetuned GPT-J was trained for one epoch. Detailed model hyper-parameters and training code can be found in the associated repos-

Model	BoolQ	PIQA	HellaSwag	WinoGrande	ARC-e	ARC-c	OBQA
GPT4All-J 6.7B	73.4	74.8	63.4	64.7	54.9	36.0	40.2
GPT4All-J Lora 6.7B	68.6	75.8	66.2	63.5	56.4	35.7	40.2
GPT4All LLaMa Lora 7B	73.1	77.6	72.1	67.8	51.1	40.4	40.2
Dolly 6B	68.8	77.3	67.6	63.9	62.9	38.7	41.2
Dolly 12B	56.7	75.4	71.0	62.2	**64.6**	38.5	40.4
Alpaca 7B	73.9	77.2	73.9	66.1	59.8	43.3	43.4
Alpaca Lora 7B	**74.3**	**79.3**	**74.0**	**68.8**	56.6	**43.9**	**42.6**
GPT-J 6.7B	65.4	76.2	66.2	64.1	62.2	36.6	38.2
LLaMa 7B	73.1	77.4	73.0	66.9	52.5	41.4	42.4
Pythia 6.7B	63.5	76.3	64.0	61.1	61.3	35.2	37.2
Pythia 12B	67.7	76.6	67.3	63.8	63.9	34.8	38

Table 1: Zero-shot performance on Common Sense Reasoning tasks



(a) TSNE visualization of the final GPT4All-J training data, ten-colored by extracted topic.



(b) Zoomed in view of Figure 1a. The region displayed contains generations related to personal health and wellness.

Figure 1: The final training data was curated to ensure a diverse distribution of prompt topics and model responses. View online

itory and model training log. We additionally release both GPT-J and GPT-J LoRa checkpoints. Updates to the training log were made to include the additional experiments run for GPT-J.

2.1 Reproducibility

We release all data, training code and logs for the community to learn, build and benefit from. Please check the Git repository for the most up-to-date data, training details and checkpoints.

2.2 Costs

Running all of our experiments cost about $5000 in GPU costs. We gratefully acknowledge our compute sponsor Paperspace for their generosity in making GPT4All-J training possible. Between GPT4All and GPT4All-J, we have spent about $800 in OpenAI API credits so far to generate the training samples that we openly release to the community. Our released model, GPT4All-J, can be trained in about eight hours on a Paperspace DGX A100 8x 80GB for a total cost of $200. Using a government calculator, we estimate the final model training to produce the equivalent of 0.18 metric tons of carbon dioxide, roughly equivalent to that produced by burning 20 gallons (75 liters) of gasoline.

3 Evaluation

We perform a preliminary evaluation of our model using the human evaluation data from the Self-Instruct paper (Wang et al., 2022). We report the ground truth perplexity of our model against what is, to our knowledge, the best openly available alpaca-lora model, provided by user chainyo on huggingface. We find that all models have very

large perplexities on a small number of tasks, and report perplexities clipped to a maximum of 100.

Models fine-tuned on this collected dataset exhibit much lower perplexity in the Self-Instruct evaluation compared to Alpaca. This evaluation is in no way exhaustive and further evaluation work remains. We welcome the reader to run the model locally on CPU (see Github for files).

3.1 Common Sense Reasoning

Following results from (Conover et al.), we evaluate on 7 standard common sense reasoning tasks: ARC easy and challenge (Clark et al., 2018), BoolQ (Clark et al., 2019), HellaSwag (Zellers et al., 2019), OpenBookQA (Mihaylov et al., 2018), and Winogrande (Sakaguchi et al., 2019). We evaluate several models: GPT-J (Wang and Komatsuzaki, 2021), Pythia (6B and 12B) (Biderman et al., 2023), Dolly v1 and v2 (Conover et al.), and GPT4All using lm-eval-harness (Gao et al., 2021). Similar to results in (Ouyang et al., 2022), instruction-tuning showed performance regressions over the base model. However, we notice in some tasks that the LoRA instruction fine-tuned models show some performance improvements.

4 Use Considerations

The authors release data and training details in hopes that it will accelerate open LLM research, particularly in the domains of fairness, alignment, interpretability, and transparency. GPT4All-J model weights and quantized versions are released under an Apache 2 license and are freely available for use and distribution. Please note that the less restrictive license does not apply to the original GPT4All model that is based on LLaMA, which has a non-commercial GPL license. The assistant data was gathered from OpenAI's GPT-3.5-Turbo, whose terms of use prohibit developing models that compete commercially with OpenAI.

References

Stella Biderman, Hailey Schoelkopf, Quentin Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar van der Wal. 2023. Pythia: A suite for analyzing large language models across training and scaling.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. Boolq: Exploring the surprising difficulty of natural yes/no questions.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge.

Mike Conover, Matt Hayes, Ankit Mathur, Xiangrui Meng, Jianwei Xie, Jun Wan, Sam Shah, Ali Ghodsi, Patrick Wendell, Matei Zaharia, and et al. Free dolly: Introducing the world's first truly open instruction-tuned llm.

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. 2021. A framework for few-shot language model evaluation.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *EMNLP*.

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, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. Winogrande: An adversarial winograd schema challenge at scale.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models.

Ben Wang and Aran Komatsuzaki. 2021. GPT-J-6B: A 6 Billion Parameter Autoregressive Language Model. https://github.com/kingoflolz/mesh-transformer-jax.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2022. Self-instruct: Aligning language model with self generated instructions.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence?