GPT4All: An ecosystem of open-source assistants that run on local hardware.

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

We release two new models: GPT4All-J v1.3 Groovy an Apache-2 licensed chatbot, and GPT4All-13B-snoozy, a GPL licenced 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 previous GPT4All releases by training on a larger and clearner corpus, 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 support to our Chat UI with 4 bit-quantized versions of our models, allowing virtually anyone to run the model on CPU.

1 Data Collection and Curation

We collected roughly one million prompt-response pairs using the GPT-3.5-Turbo OpenAI API between March 20, 2023 and March 26th, 2023. To do this, we first gathered a diverse sample of questions/prompts by leveraging five publicly available datasets:

- The unified_chip2 subset of LAION OIG.
- Coding questions with a random sub-sample of Stackoverflow Questions
- Instruction-tuning with a sub-sample of Bigscience/P3
- Conversation Data from ShareGPT
- Instruction Following Data curated for Dolly Dolly (Conover et al.)

We additionally curated a creative-style dataset using GPT-3.5-Turbo to generate poems, short stories, and raps in the style of various artists.

Following our previous work in previous versions of GPT4All, we curated and cleaned the data using Atlas. We similarly filtered examples that contained phrases like "I'm sorry, as an AI language model" and responses where the model refused to answer the question. Adding ShareGPT (90k examples) and Dolly (15k examples), our total dataset size was approximately 900,000 datapoints including our previously published dataset which contained 806,199 examples.

After cleaning the data for exact duplicates, missing prompts and responses, and poorly formatted data, we realized that there existed many data points that were semantically similar. We tagged potential duplicates using Atlas and on inspection found that the clusters accurately contained semantic duplicates. Using these tags, we quickly removed approximately 8% of semantic duplicates.

Our final dataset, which GPT4All-J-v1.3 Groovy and GPT4All-13b-snoozy were trained on, contains 739,259 examples which is visualized in 1

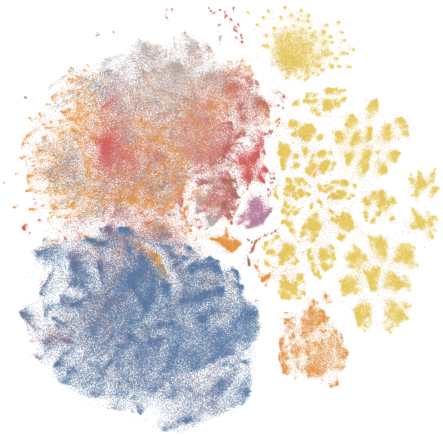

Figure 1: TSNE visualization of the training data including Stackoverflow, UnifiedChip, ShareGPT, Dolly, and data generated by Nomic). The yellow clusters are sections of creative prompts, such as asking for short stores or raps in the style of.

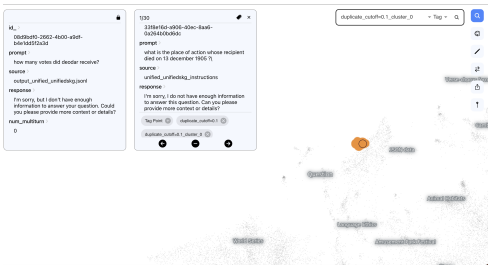

Figure 2: Cluster of Semantically Similar Examples Identified by Atlas Duplication Detection

Figure 3: TSNE visualization of the final GPT4All training data, colored by extracted topic.

You can interactively explore the dataset at the following links:

- Deduplicated Dataset Indexed on Prompt
- Deduplicated Dataset Indexed on Response

The final dataset has been added as a revision named ‘v1.3-groovy‘ to GPT4All-J dataset. Here you can also find previous versions of our dataset.

2 Model Training

We trained models finetuned from both LLaMA 13B (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 fine-tuned GPT-J was trained for one epoch. These models that we release are full fine-tunes. Detailed model hyper-parameters and training code can be found in the associated repository and model training log.

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 and GPT4All-13B-snoozy 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 while GPT4All-13B-snoozy can be trained in about 1 day for a total cost of $600. Using a government calculator, we estimate the model training to produce the equivalent of 0.18 and 0.54 metric tons of carbon dioxide for GPT4All-J and GPT4All-13B-snoozy, roughly equivalent to that produced by burning 20 gallons (75 liters) and 60 gallons (227 liters) of gasoline respectively.

3 Evaluation

We performed a preliminary evaluation of our original 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.,

Model	BoolQ	PIQA	HellaSwag	WinoGrande	ARC-e	ARC-c	OBQA	Avg.
GPT4All-J 6B v1.0	73.4	74.8	63.4	64.7	54.9	36.0	40.2	58.2
GPT4All-J v1.1-breezy	74.0	75.1	63.2	63.6	55.4	34.9	38.4	57.8
GPT4All-J v1.2-jazzy	74.8	74.9	63.6	63.8	56.6	35.3	41.0	58.6
GPT4All-J v1.3-groovy	73.6	74.3	63.8	63.5	57.7	35.0	38.8	58.1
GPT4All-J Lora 6B	68.6	75.8	66.2	63.5	56.4	35.7	40.2	58.1
GPT4All LLaMa Lora 7B	73.1	77.6	72.1	67.8	51.1	40.4	40.2	60.3
GPT4All 13B snoozy	**83.3**	79.2	75.0	**71.3**	60.9	44.2	43.4	**65.3**
Dolly 6B	68.8	77.3	67.6	63.9	62.9	38.7	41.2	60.1
Dolly 12B	56.7	75.4	71.0	62.2	**64.6**	38.5	40.4	58.4
Alpaca 7B	73.9	77.2	73.9	66.1	59.8	43.3	43.4	62.4
Alpaca Lora 7B	74.3	**79.3**	74.0	68.8	56.6	43.9	42.6	62.8
GPT-J 6.7B	65.4	76.2	66.2	64.1	62.2	36.6	38.2	58.4
LLama 7B	73.1	77.4	73.0	66.9	52.5	41.4	42.4	61.0
LLama 13B	68.5	79.1	**76.2**	70.1	60.0	**44.6**	42.2	63.0
Pythia 6.7B	63.5	76.3	64.0	61.1	61.3	35.2	37.2	57.0
Pythia 12B	67.7	76.6	67.3	63.8	63.9	34.8	38	58.9
Fastchat T5	81.5	64.6	46.3	61.8	49.3	33.3	39.4	53.7
Fastchat Vicuña 7B	76.6	77.2	70.7	67.3	53.5	41.2	40.8	61.0
Fastchat Vicuña 13B	81.5	76.8	73.3	66.7	57.4	42.7	43.6	63.1
StableVicuña RLHF	82.3	78.6	74.1	70.9	61.0	43.5	**44.4**	65.0
StableLM Tuned	62.5	71.2	53.6	54.8	52.4	31.1	33.4	51.3
StableLM Base	60.1	67.4	41.2	50.1	44.9	27.0	32.0	42.2
Koala 13B	76.5	77.9	72.6	68.8	54.3	41.0	42.8	62.0
Open Assistant Pythia 12B	67.9	78.0	68.1	65.0	64.2	40.4	43.2	61.0
text-davinci-003	88.1	83.8	83.4	75.8	83.9	63.9	51.0	75.7

Table 1: Zero-shot performance on Common Sense Reasoning tasks. The highest performing non-OpenAI model is bolded in each column. Note text-davinci-003 still beats every model on these tasks.

2018), PIQA (Bisk et al., 2020), and Winogrande (Sakaguchi et al., 2019). We evaluate several models: GPT-J (Wang and Komatsuzaki, 2021), Pythia (6B and 12B) (**?**), Dolly v1 and v2 (Conover et al.), Alpaca (Taori et al., 2023), LLama (Touvron et al., 2023), Pythia 6B and 12B (Biderman et al., 2023), FastChat T5 and Vicuña (Chiang et al., 2023), StableVicuña and StableLM base and tuned (Stability-AI), Koala (Geng et al., 2023), OpenAssistant 12B (Laion-Ai), text-davinci-003 (Ouyang et al., 2022) and GPT4All using lm-eval-harness (Gao et al., 2021). Similar to results in (Ouyang et al., 2022), instruction-tuning showed some performance regressions over the base model for. However, we notice in some tasks that the LoRA instruction finetuned models as well as GPT4All-13B-snoozy trained on a large cleaned dataset 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 and GPT4All-13B-snoozy 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.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. 2020. Piqa: Reasoning about physical commonsense in natural language. In *Thirty-Fourth AAAI Conference on Artificial Intelligence*.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.

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.

Xinyang Geng, Arnav Gudibande, Hao Liu, Eric Wallace, Pieter Abbeel, Sergey Levine, and Dawn Song. 2023. Koala: A dialogue model for academic research. Blog post.

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.

Laion-Ai. Laion-ai/open-assistant: Openassistant is a chat-based assistant that understands tasks, can interact with third-party systems, and retrieve information dynamically to do so.

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.

Stability-AI. Stability-ai/stablelm: Stablelm: Stability ai language models.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`.

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?