Pre-training Small Base LMs with Fewer Tokens

Sunny Sanyal   Sujay Sanghavi   Alexandros G. Dimakis

UT Austin

TEXAS

Paper   Tweet

We trained a 1.5B language model with just 1B available tokens using 1 GPU for less than half a day.

1. Game of Data and Compute

Small base LMs with few billion parameters (1B-2B) are pre-trained using billions/trillions of tokens and it's extremely compute intensive.

Models (# train tokens)	GPU Count	GPU Type	Time (# days)
MPT-1.3B (200B)	440	A100	half
Pythia-1.4B (300B)	64	A100	4.6
TinyLLaMA-1.1B (3T)	16	A100	90
OPT-1.3B (300B)	992	A100	–
Sheared LLaMA-1.3B (50B)	16	A100	–
OpenLLaMA-3B (1T)	256	TPU v4	10
Our-1.5B (1B)	1	A6000	~half

Table 1. Pre-training compute details of some publicly available small base LMs compared to our model developed using our proposed recipe **Inheritune**.

2. Inheritune for Low Data Regime

We assume the existence of a pre-trained large base language model, denoted as $\mathcal{M}_{\text{ref}}$. Only a small subset of its pre-training data, represented as $\hat{\mathcal{D}}_{\text{train}} \sim \mathcal{D}_{\text{train}}$, is available.

Step 1: Inherit the first n layers of $\mathcal{M}_{ref}$ to target $\mathcal{M}_{\text{tgt}}$. The prediction head and token embedding are also inherited.

Step 2: Train $\mathcal{M}_{\text{tgt}}$ with the available training data $\hat{\mathcal{D}}_{\text{train}}$ for multiple passes over the data.

3. Inheritune with Full Pre-train Data

We have a pre-trained large base language model, denoted as $\mathcal{M}_{\text{ref}}$ trained with $\mathcal{D}_{\text{train}}$ for T steps. Evaluate $\mathcal{M}_{\text{ref}}$ with $\mathcal{D}_{\text{val}}$ to obtain a benchmark val loss.

Step 1: Inherit the first n layers of $\mathcal{M}_{ref}$ to target $\mathcal{M}_{\text{tgt}}$. The prediction head and token embedding are also inherited.

Step 2: Train $\mathcal{M}_{\text{tgt}}$ with full training data $\mathcal{D}_{\text{train}}$ for T steps.

Step 3: Grow $\mathcal{M}_{\text{tgt}}$ and retrain (step 1) until it matches the benchmark val loss.

4. Main results of Inheritune with 1B tokens

Model	Commonsense Reasoning				
Name (# train tokens)	Winograd	PIQA	Boolq	WinoGrande	Logiqa
OpenLLaMA-3B (1T)	63.46	74.97	67.18	62.27	28.4
OPT-1.3B (300B)	38.46	71.82	57.83	59.51	27.04
Pythia-1.4B (300B)	36.54	70.89	63.12	56.99	27.65
MPT-1.3B (200B)	63.46	71.44	50.89	58.09	28.26
Sheared LLaMA-1.3B (50B)	36.54	73.45	62.02	58.17	27.34
Ours-1.5B (1B)	**50.96**	56.47	**61.68**	51.69	25.19

Model	Lang. Understanding & Inference				Factuality
Name (# train tokens)	MMLU(5)	WNLI	QNLI	MNLI	TruthfulQA
OpenLLaMA-3B (1T)	27.21	50.7	51.3	37.3	35
OPT-1.3B (300B)	24.96	42.25	51.29	35.82	38.67
Pythia-1.4B (300B)	25.56	53.52	49.48	32.76	38.66
MPT-1.3B (200B)	25.82	40.85	50.52	35.93	38.68
Sheared LLaMA-1.3B (50B)	25.71	49.3	50.98	37.94	37.14
Ours-1.5B (1B)	**25.67**	**43.66**	**49.41**	**34.42**	**48.61**

Table 2. Comparison of Our-1.5B small base LM derived using **Inheritune** with OpenLLaMA-3B as reference LM and other baseline models of similar size. Our model although trained with fewer tokens achieves comparable performance compared to the baseline models. We have highlighted all the scores in **bold** where Our-1.5B model achieves at least 90% of the score compared to its reference LM or it outperforms at least two of the publicly available baseline LMs. All the tasks are evaluated using 0 shot except MMLU which is 5-shot.

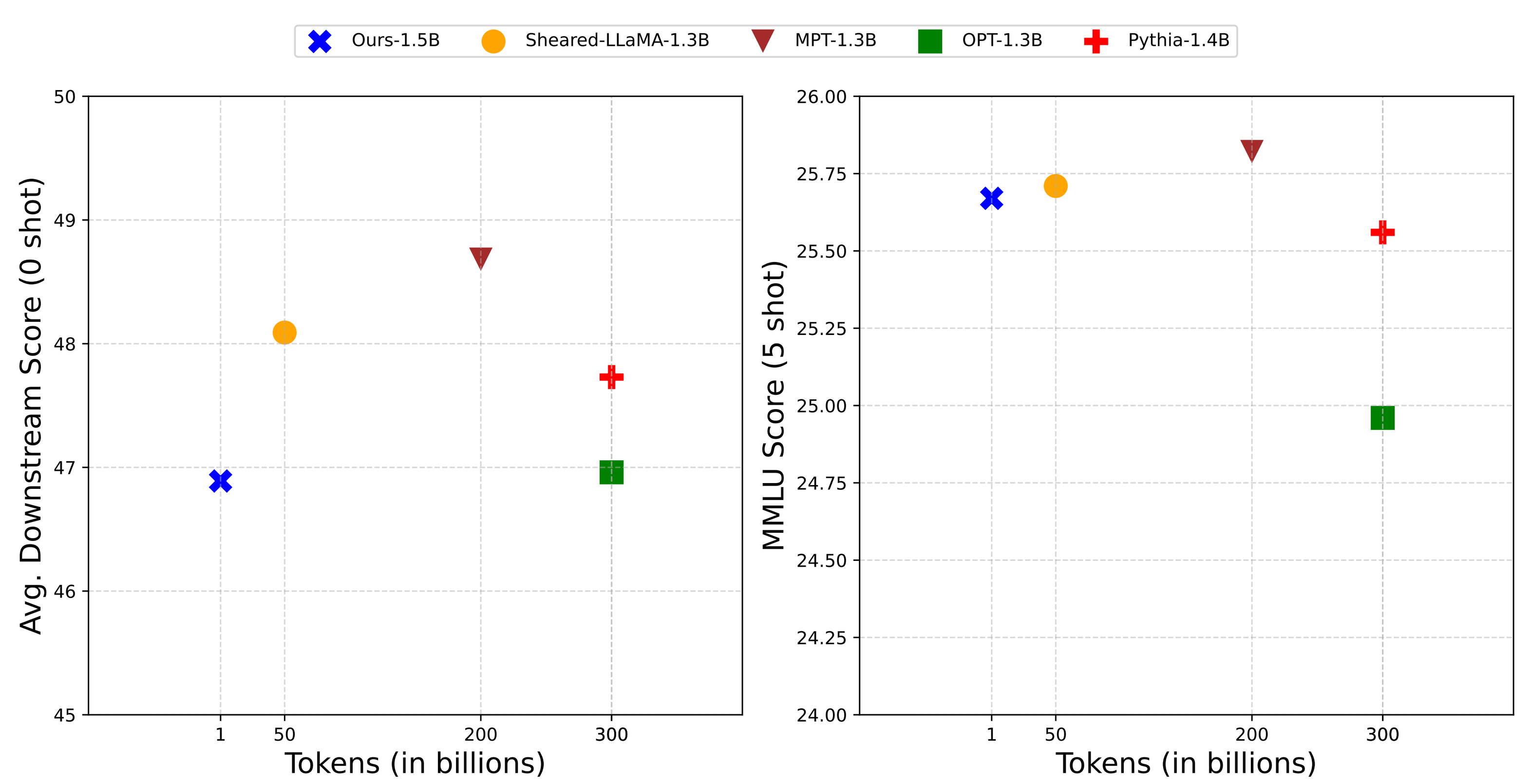

Figure 1. Performance of our 1.5B base LM derived using 1B tokens with **Inheritune** on an average of 9 different datasets (left) and MMLU benchmark (right).

5. Inheritune improves MMLU with more data

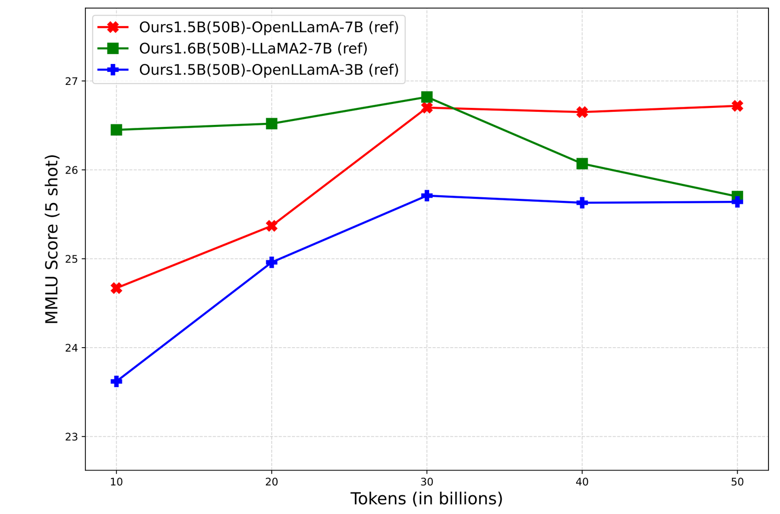

Figure 2. Performance of our small base LM derived using **Inheritune** using 50B tokens (without repetition) on MMLU benchmark. We present results with three different reference models OpenLLaMA-7B, LLaMA2-7B and OpenLLaMA-3B.

Model (# tokens)	Data type	MMLU (5-shot)
Ours-1.5B (1B)	10 epochs	24.95
Ours-1.5B (50B)	10B fresh	23.62
Ours-1.5B (1B)	20 epochs	25.46
Ours-1.5B (50B)	20B fresh	24.96

Table 3. MMLU (5-shot) performance of Our-1.5B small base LM derived using 1B data with multiple data repetition–10 epochs and 20 epochs compared to the same model trained without data repetition for 10B and 20B fresh tokens.

6. Main results of Inheritune with Full Pre-train Data

Models	Layers	Initialization	Steps	Pre-train	Downstream (0-shot)	
				Val loss (↓)	Wikitext (↓)	Lambada
GPT-2 Large	36	rand init	100K	2.85	34.84	34.14
	18	rand init	100K	2.97	37.63	30.97
	18	rand init	200K	2.84	–	–
	18	Ours	100K	2.80	35.38	34.64
GPT-2 Medium	24	rand init	100K	2.81	31.93	36.54
	16	rand init	100K	2.86	33.67	34.60
	16	rand init	200K	2.83	–	–
	12	Ours	100K	2.87	–	–
	14	Ours	100K	2.84	–	–
Final Model ⟶	16	Ours	100K	2.81	32.04	35.96

Table 4. Pre-training and downstream performance of GPT-2 medium and large LLMs evaluated using val loss, Wikitext, and Lambada downstream tasks. Small LMs derived using our method perform comparably to their full-sized counterparts for GPT-2 large and GPT-2 medium.

sanyal.sunny@utexas.edu

Sanghavi Lab, EER, 2501 Speedway, Austin, TX 7871