What powers Replit AI?

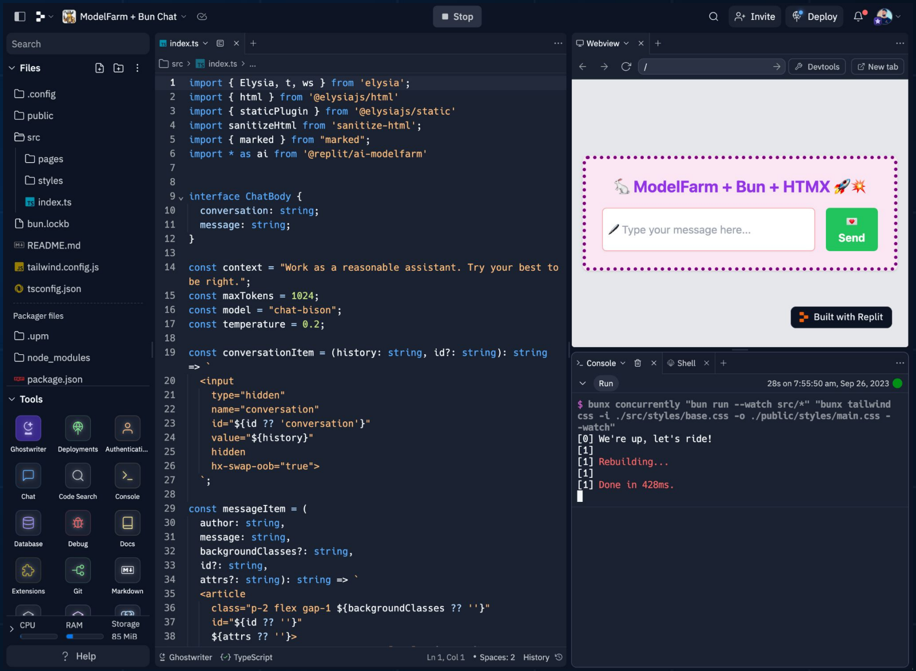
Michele Catasta

https://twitter.com/pirroh

https://pirroh.fyi

What is Replit?

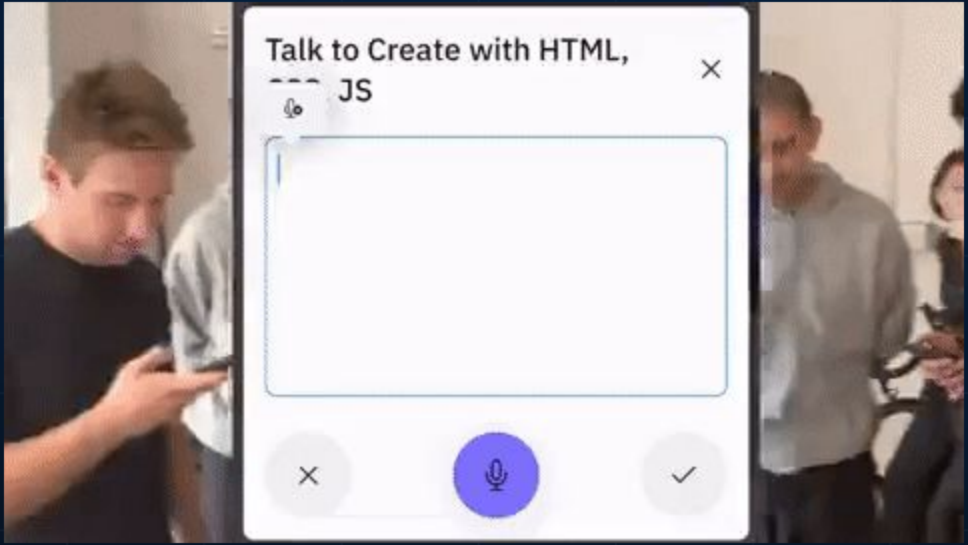
- Platform to build and collaborate on code in any language
- 22M+ community of creators and learners
- Users get their own cloud computer to develop, run, and deploy apps
- Company founded 2016 based on side project in 2011 to put coding in browser. Today: $1B VC-backed company



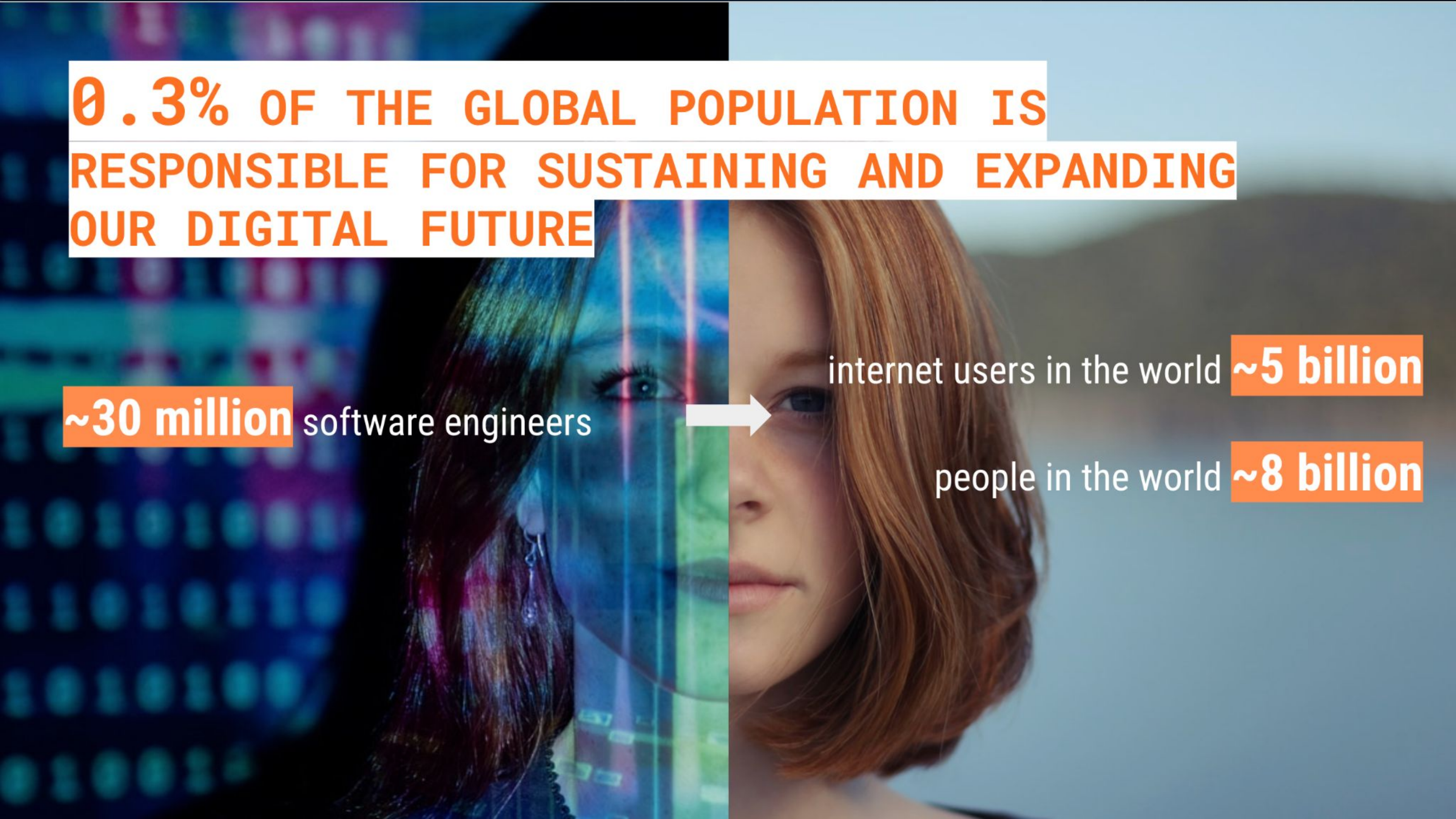
```typescript
import { Elysia, t, ws } from 'elysia';
import { html } from '@elysiajs/html'
import { staticPlugin } from '@elysiajs/static'
import sanitizeHtml from 'sanitize-html';
import { marked } from "marked";
import * as ai from '@replit/ai-modelfarm'


interface ChatBody {
  conversation: string;
  message: string;
}

const context = "Work as a reasonable assistant. Try your best to be right.";
const maxTokens = 1024;
const model = "chat-bison";
const temperature = 0.2;

const conversationItem = (history: string, id?: string): string => `
  <input
    type="hidden"
    name="conversation"
    id="${id ?? 'conversation'}"
    value="${history}"
    hidden
    hx-swap-oob="true">
  `;

const messageItem = (
  author: string,
  message: string,
  backgroundClasses?: string,
  id?: string,
  attrs?: string): string => `
  <article
    class="p-2 flex gap-1 ${backgroundClasses ?? ''}"
    id="${id ?? ''}"
    ${attrs ?? ''}>
```

```
$ bunx concurrently "bun run --watch src/*" "bunx tailwind
css -i ./src/styles/base.css -o ./public/styles/main.css -
-watch"
[0] We're up, let's ride!
[1]
[1] Rebuilding...
[1]
[1] Done in 428ms.
```

North star: Speak software into existence



0.3% OF THE GLOBAL POPULATION IS RESPONSIBLE FOR SUSTAINING AND EXPANDING OUR DIGITAL FUTURE



~30 million software engineers →

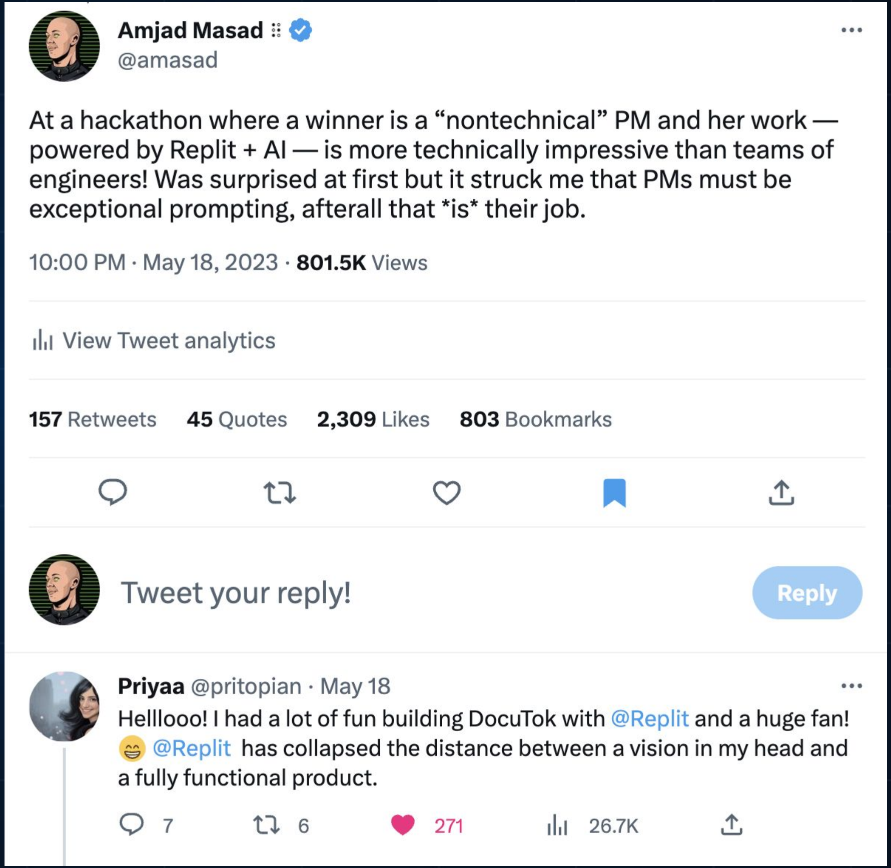
internet users in the world **~5 billion**

people in the world **~8 billion**

How do we empower the next billion software developers?

AI + Software Creation = 1B+ devs

- Like every other medium, software is getting easier to make
- Can go from an idea to software in mere minutes
- This means software is cheaper & faster to make
- Demand for software will go up
- Massive expansion of what it means to be a “developer”



Amjad Masad ✓
@amasad

At a hackathon where a winner is a "nontechnical" PM and her work — powered by Replit + AI — is more technically impressive than teams of engineers! Was surprised at first but it struck me that PMs must be exceptional prompting, afterall that *is* their job.

10:00 PM · May 18, 2023 · **801.5K** Views

View Tweet analytics

157 Retweets **45** Quotes **2,309** Likes **803** Bookmarks

Tweet your reply!

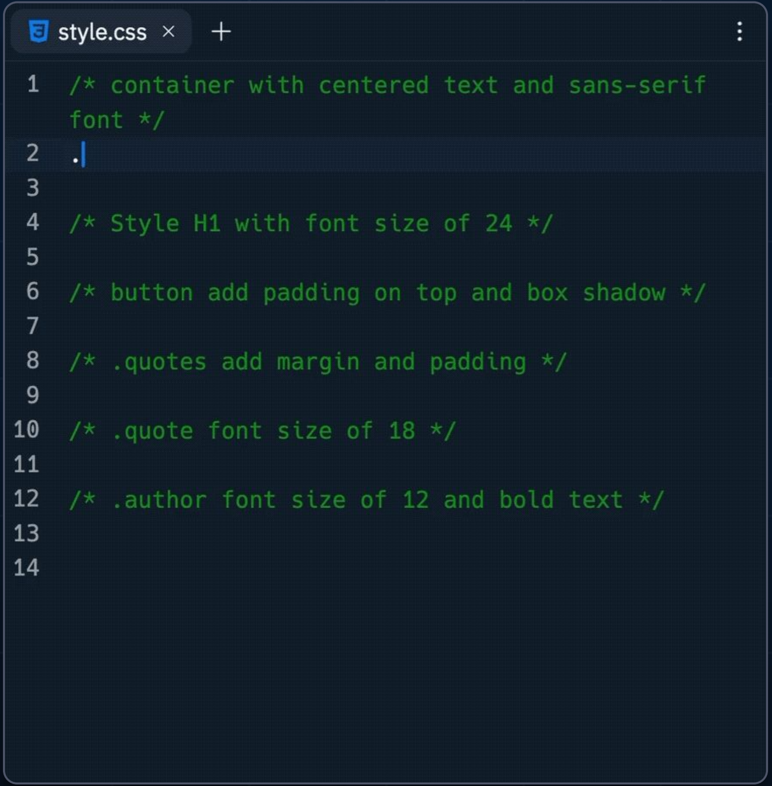
Reply

Priyaa @pritopian · May 18

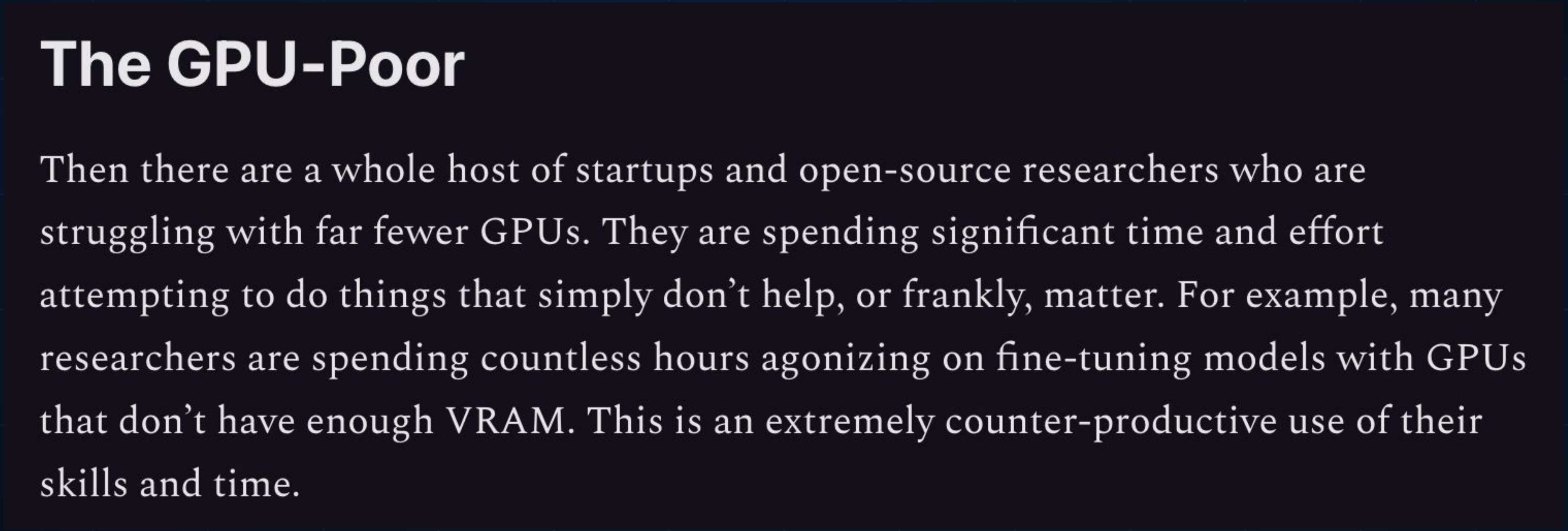
Helllooo! I had a lot of fun building DocuTok with @Replit and a huge fan! 😁 @Replit has collapsed the distance between a vision in my head and a fully functional product.

7 6 271 26.7K

Code Completion on Replit



```
/* container with centered text and sans-serif font */
.

/* Style H1 with font size of 24 */

/* button add padding on top and box shadow */

/* .quotes add margin and padding */

/* .quote font size of 18 */

/* .author font size of 12 and bold text */


```

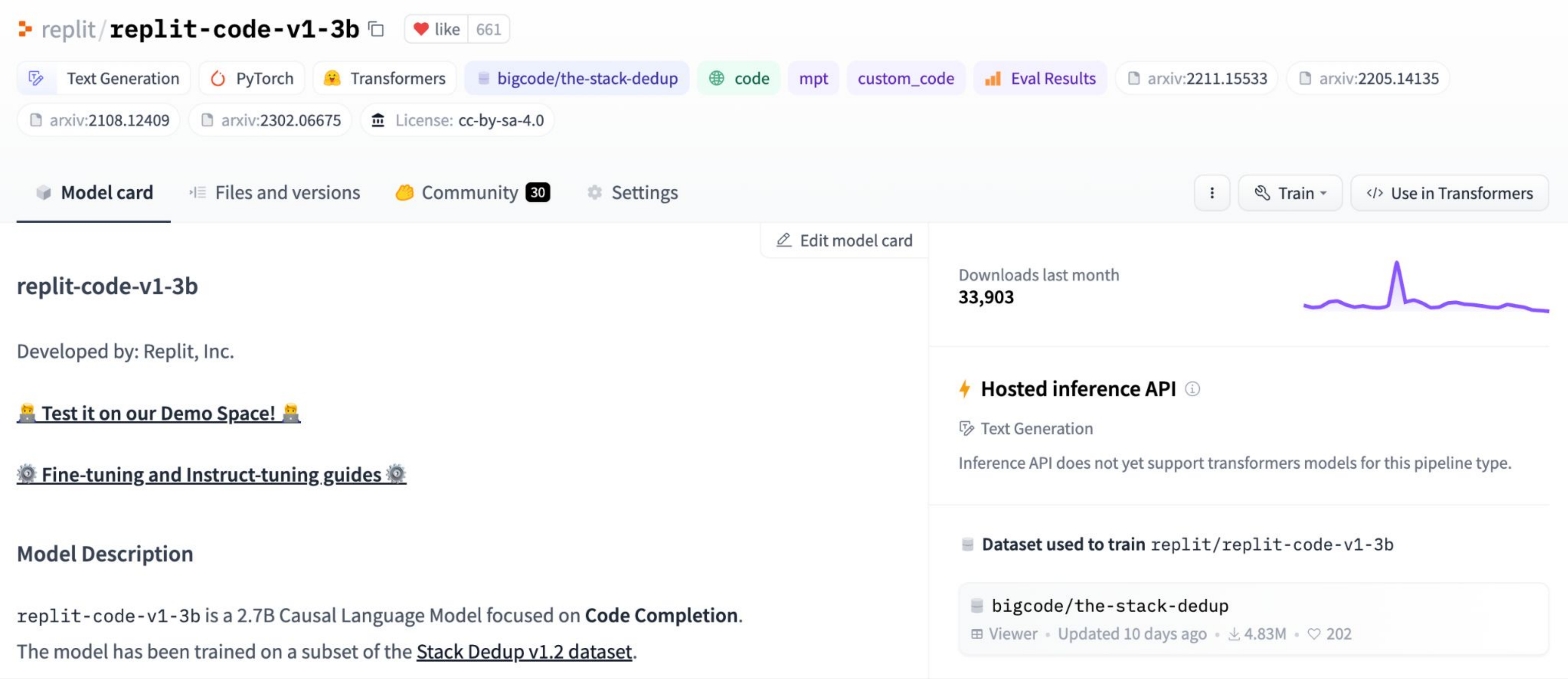


The GPU-Poor

Then there are a whole host of startups and open-source researchers who are struggling with far fewer GPUs. They are spending significant time and effort attempting to do things that simply don’t help, or frankly, matter. For example, many researchers are spending countless hours agonizing on fine-tuning models with GPUs that don’t have enough VRAM. This is an extremely counter-productive use of their skills and time.

https://www.semianalysis.com/p/google-gemini-eats-the-world-gemini

In early May 2023 we released replit-code-v1-3b, our bespoke Code Completion LLM serving a large number of Replit users



replit-code-v1-3b

Developed by: Replit, Inc.

Test it on our Demo Space!

Fine-tuning and Instruct-tuning guides

Model Description

`replit-code-v1-3b` is a 2.7B Causal Language Model focused on **Code Completion**. The model has been trained on a subset of the Stack Dedup v1.2 dataset.

Downloads last month
33,903

Hosted inference API

Text Generation

Inference API does not yet support transformers models for this pipeline type.

Dataset used to train `replit/replit-code-v1-3b`

`bigcode/the-stack-dedup`
Viewer • Updated 10 days ago • 4.83M • 202

replit-code-v1-3b / **Data**

First Llama-style LLM for code

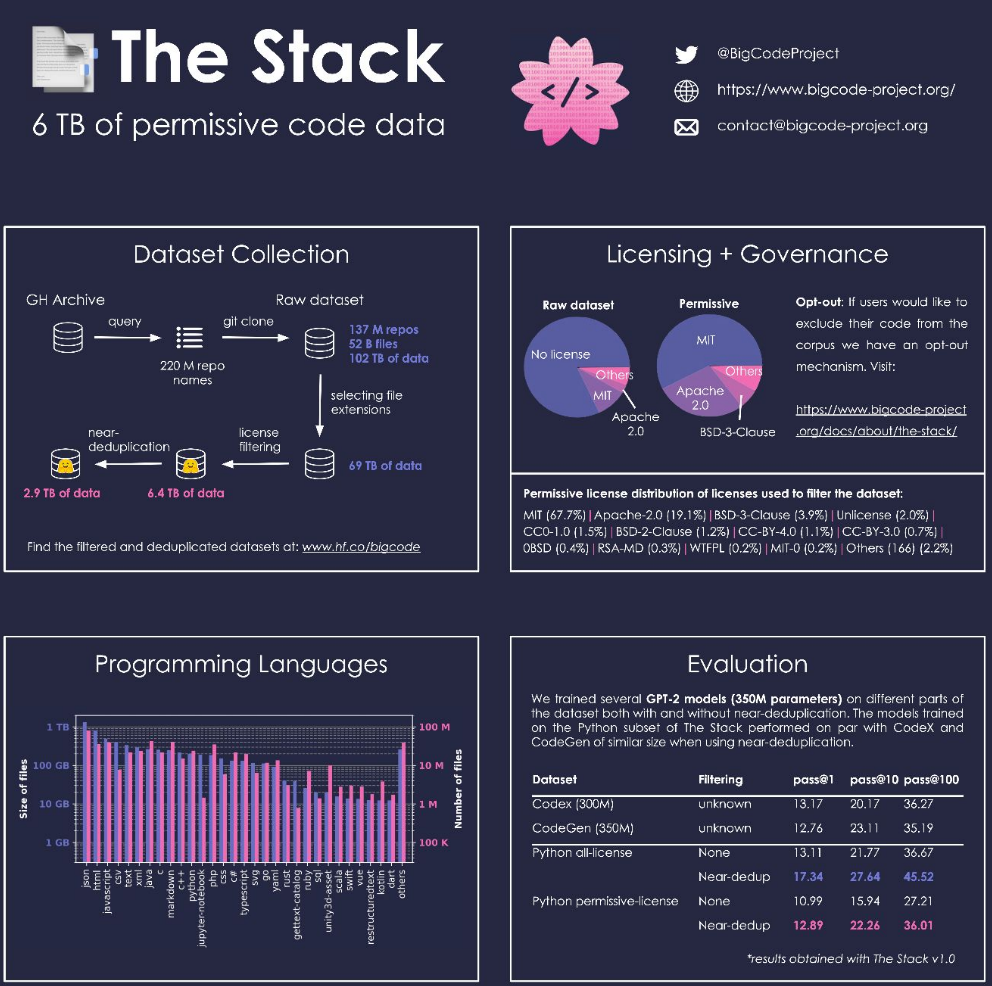
~195 tokens per parameter

Trained on 525B tokens of code

175B tokens over 3 epochs

20 languages

Markdown, Java, JavaScript, Python, TypeScript, PHP, SQL, JSX, reStructuredText, Rust, C, CSS, Go, C++, HTML, Vue, Ruby, Jupyter Notebook, R, Shell



The Stack

6 TB of permissive code data

@BigCodeProject

https://www.bigcode-project.org/

contact@bigcode-project.org

Dataset Collection

GH Archive
query
220 M repo names
git clone
Raw dataset
137 M repos
52 B files
102 TB of data
selecting file extensions
69 TB of data
license filtering
6.4 TB of data
near-deduplication
2.9 TB of data

Find the filtered and deduplicated datasets at: www.hf.co/bigcode

Licensing + Governance

Raw dataset
No license
Others
MIT
Apache 2.0
Permissive
MIT
Others
Apache 2.0
BSD-3-Clause

Opt-out: If users would like to exclude their code from the corpus we have an opt-out mechanism. Visit:

https://www.bigcode-project.org/docs/about/the-stack/

Permissive license distribution of licenses used to filter the dataset:

MIT (67.7%) | Apache-2.0 (19.1%) | BSD-3-Clause (3.9%) | Unlicense (2.0%) | CC0-1.0 (1.5%) | BSD-2-Clause (1.2%) | CC-BY-4.0 (1.1%) | CC-BY-3.0 (0.7%) | 0BSD (0.4%) | RSA-MD (0.3%) | WTFPL (0.2%) | MIT-0 (0.2%) | Others (166) (2.2%)

Programming Languages

Evaluation

We trained several **GPT-2 models (350M parameters)** on different parts of the dataset both with and without near-deduplication. The models trained on the Python subset of The Stack performed on par with CodeX and CodeGen of similar size when using near-deduplication.

Dataset	Filtering	pass@1	pass@10	pass@100
Codex (300M)	unknown	13.17	20.17	36.27
CodeGen (350M)	unknown	12.76	23.11	35.19
Python all-license	None	13.11	21.77	36.67
	Near-dedup	17.34	27.64	45.52
Python permissive-license	None	10.99	15.94	27.21
	Near-dedup	12.89	22.26	36.01

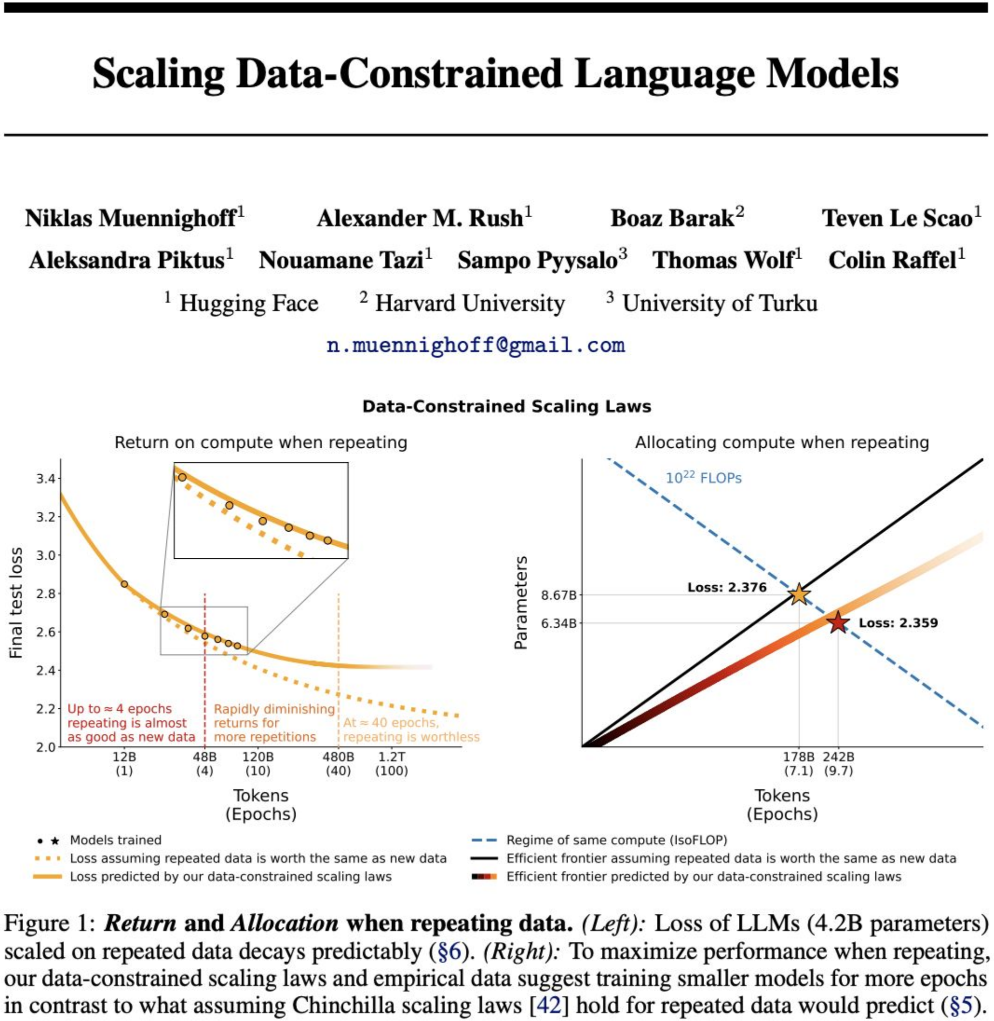
**results obtained with The Stack v1.0*

- Pretraining data mixture based on The Stack v1.2 (released in March 2023)

- Selected the top 20 languages used on Replit

- Large number of code quality heuristics to filter the dataset (e.g., Codex paper, stripping long content from HTML/CSS files, etc.)

- Data processing on Spark, vocabulary training with Google SentencePiece



Scaling Data-Constrained Language Models

Niklas Muennighoff[1] Alexander M. Rush[1] Boaz Barak[2] Teven Le Scao[1]
Aleksandra Piktus[1] Nouamane Tazi[1] Sampo Pyysalo[3] Thomas Wolf[1] Colin Raffel[1]
[1] Hugging Face [2] Harvard University [3] University of Turku
n.muennighoff@gmail.com

Data-Constrained Scaling Laws
Return on compute when repeating
Final test loss
3.4
3.2
3.0
2.8
2.6
2.4
2.2
2.0
Up to ≈ 4 epochs repeating is almost as good as new data
Rapidly diminishing returns for more repetitions
At ≈ 40 epochs, repeating is worthless
12B (1)
48B (4)
120B (10)
480B (40)
1.2T (100)
Tokens (Epochs)
Allocating compute when repeating
10^{22} FLOPs
Parameters
8.67B
6.34B
Loss: 2.376
Loss: 2.359
178B (7.1)
242B (9.7)
Tokens (Epochs)
• ★ Models trained
Loss assuming repeated data is worth the same as new data
Loss predicted by our data-constrained scaling laws
Regime of same compute (IsoFLOP)
Efficient frontier assuming repeated data is worth the same as new data
Efficient frontier predicted by our data-constrained scaling laws

Figure 1: ***Return*** **and** ***Allocation*** **when repeating data.** *(Left):* Loss of LLMs (4.2B parameters) scaled on repeated data decays predictably (§6). *(Right):* To maximize performance when repeating, our data-constrained scaling laws and empirical data suggest training smaller models for more epochs in contrast to what assuming Chinchilla scaling laws [42] hold for repeated data would predict (§5).

- Published coincidentally just a few weeks after we released our LLM
- Highly recommended paper, confirming our ablation studies on repeated data
- This intuition allowed us to train to completion using only permissively-licensed code, hence we could release our model under CC BY-SA-4.0

replit-code-v1-3b / Model Training

2.7B parameters

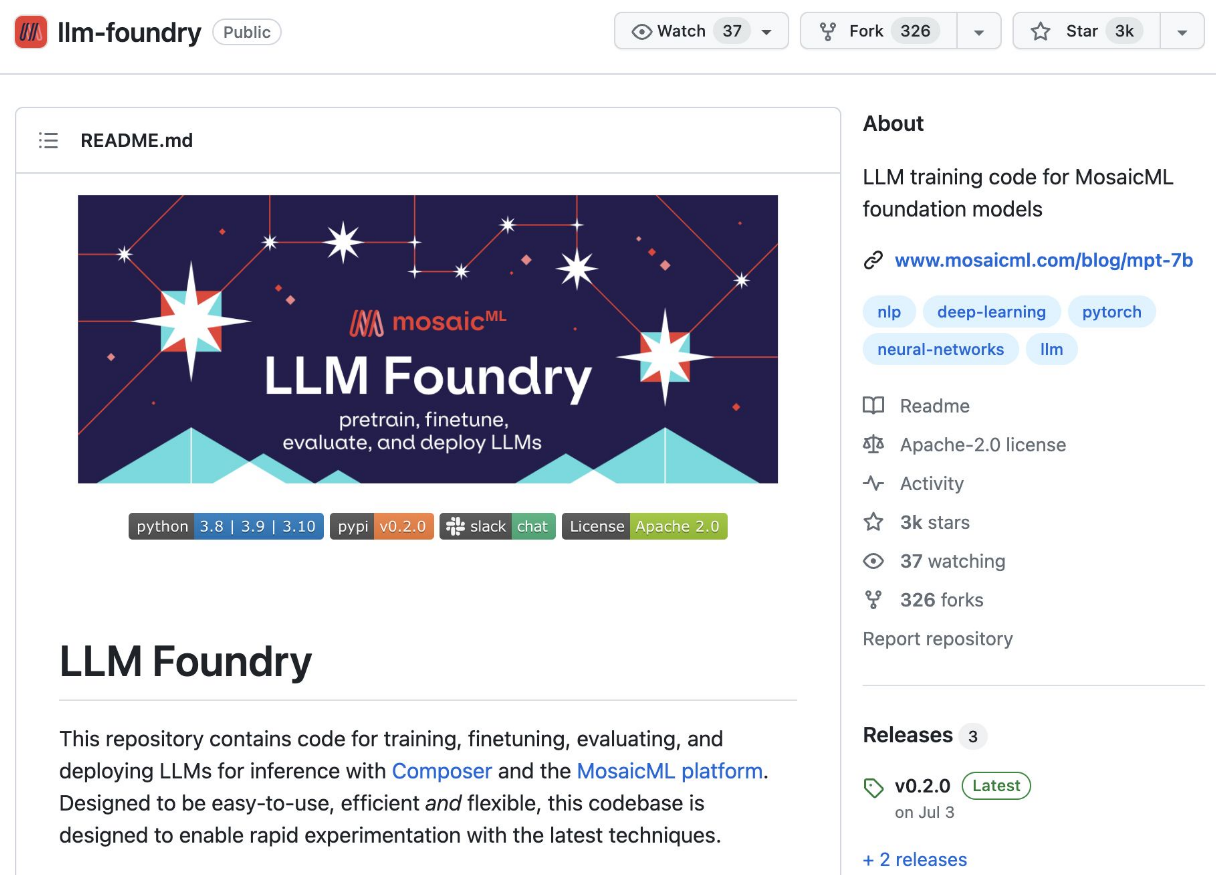
Custom 32k vocabulary focused on code

256 A100-40GB GPUs

For ~3 days on the MosaicML platform

LLM best practices

Flash Attention, AliBi positional embeddings, LionW optimizer, etc.



LLM Foundry

pretrain, finetune, evaluate, and deploy LLMs

LLM Foundry

This repository contains code for training, finetuning, evaluating, and deploying LLMs for inference with Composer and the MosaicML platform. Designed to be easy-to-use, efficient *and* flexible, this codebase is designed to enable rapid experimentation with the latest techniques.

About

LLM training code for MosaicML foundation models

www.mosaicml.com/blog/mpt-7b

nlp deep-learning pytorch neural-networks llm

- Readme
- Apache-2.0 license
- Activity
- 3k stars
- 37 watching
- 326 forks

Report repository

Releases 3

v0.2.0 Latest
on Jul 3

+ 2 releases

- All training runs based on an early release of LLM Foundry by MosaicML

- Same library used to train larger open-source models like MPT-7B and MPT-30B

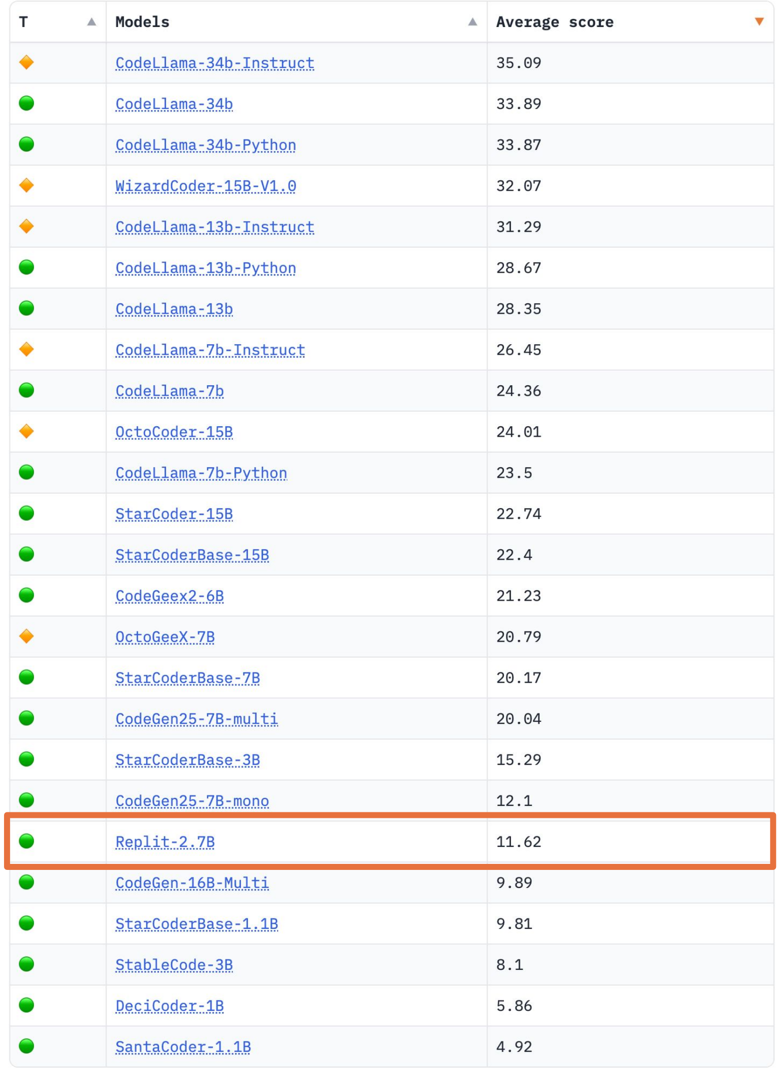
replit-code-v1-3b / Evaluation

	Score pass@1
Python (OpenAI HumanEval)	22.56%
Python (MultiPL-E)	20.49%
Java (MultiPL-E)	20.25%
JavaScript (MultiPL-E)	19.25%
C++ (MultiPL-E)	18.63%
Rust (MultiPL-E)	16.02%
PHP (MultiPL-E)	13.04%

- To navigate the latest Code LLM releases, BigCode (🤗) created Multilingual Code Models Evaluation

- Based on MultiPL-E, an extension of the original OpenAI HumanEval benchmark to 18 languages

- replit-code-v1-3b was trained only on **10 languages out of the 18** supported by MultiPL-E



T ▲	Models ▲	Average score ▼
🔶	CodeLlama-34b-Instruct	35.09
🟢	CodeLlama-34b	33.89
🟢	CodeLlama-34b-Python	33.87
🔶	WizardCoder-15B-V1.0	32.07
🔶	CodeLlama-13b-Instruct	31.29
🟢	CodeLlama-13b-Python	28.67
🟢	CodeLlama-13b	28.35
🔶	CodeLlama-7b-Instruct	26.45
🟢	CodeLlama-7b	24.36
🔶	OctoCoder-15B	24.01
🟢	CodeLlama-7b-Python	23.5
🟢	StarCoder-15B	22.74
🟢	StarCoderBase-15B	22.4
🟢	CodeGeex2-6B	21.23
🔶	OctoGeeX-7B	20.79
🟢	StarCoderBase-7B	20.17
🟢	CodeGen25-7B-multi	20.04
🟢	StarCoderBase-3B	15.29
🟢	CodeGen25-7B-mono	12.1
🟢	Replit-2.7B	11.62
🟢	CodeGen-16B-Multi	9.89
🟢	StarCoderBase-1.1B	9.81
🟢	StableCode-3B	8.1
🟢	DeciCoder-1B	5.86
🟢	SantaCoder-1.1B	4.92

replit-repltuned-v1-3b / Data & Training

Further pretraining on 111B tokens of code

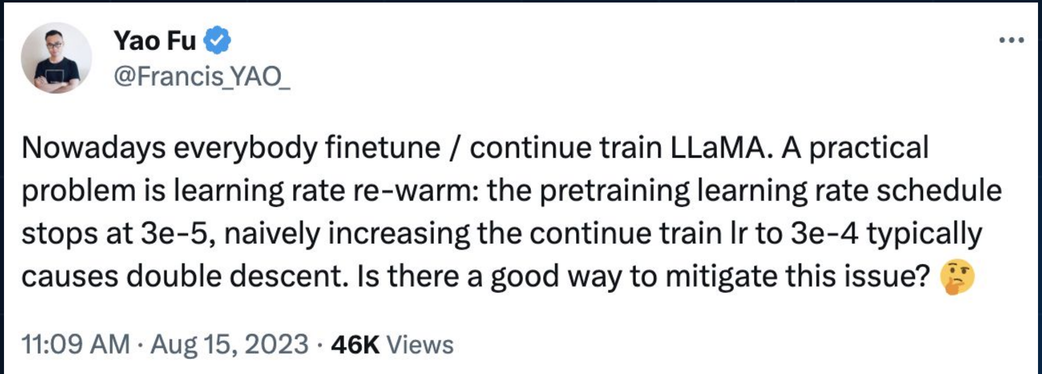
37B tokens over 3 epochs

Code authored by our users in public Repls

A lot of Python and Javascript

Same languages, same data filtering heuristics

The problem

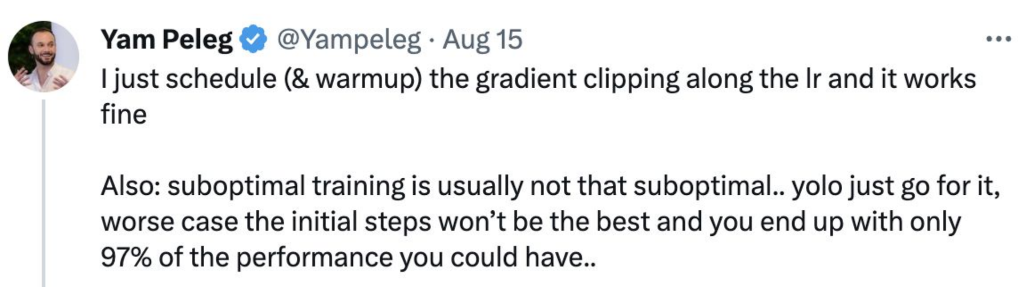


Yao Fu @Francis_YAO_

Nowadays everybody finetune / continue train LLaMA. A practical problem is learning rate re-warm: the pretraining learning rate schedule stops at 3e-5, naively increasing the continue train lr to 3e-4 typically causes double descent. Is there a good way to mitigate this issue? 🤔

11:09 AM · Aug 15, 2023 · **46K** Views

Our experience



Yam Peleg @Yampeleg · Aug 15

I just schedule (& warmup) the gradient clipping along the lr and it works fine

Also: suboptimal training is usually not that suboptimal.. yolo just go for it, worse case the initial steps won't be the best and you end up with only 97% of the performance you could have..

The solution?

- Continual Pre-Training of Large Language Models: How to (re)warm your model?
- A pragmatic hack explained by Shital Shah in this thread, inspired by the LR schedule from "Scaling Vision Transformers"

replit-repltuned-v1-3b / Evaluation

	Score pass@1	Base model
Python (OpenAI HumanEval)	30.48%	22.56%
Python (MultiPL-E)	29.81%	20.49%
Java (MultiPL-E)	19.62%	20.25%
JavaScript (MultiPL-E)	27.95%	19.25%
C++ (MultiPL-E)	26.08%	18.63%
Rust (MultiPL-E)	15.38%	16.02%
PHP (MultiPL-E)	23.60%	13.04%

replit-*-v1-3b / **Inference**

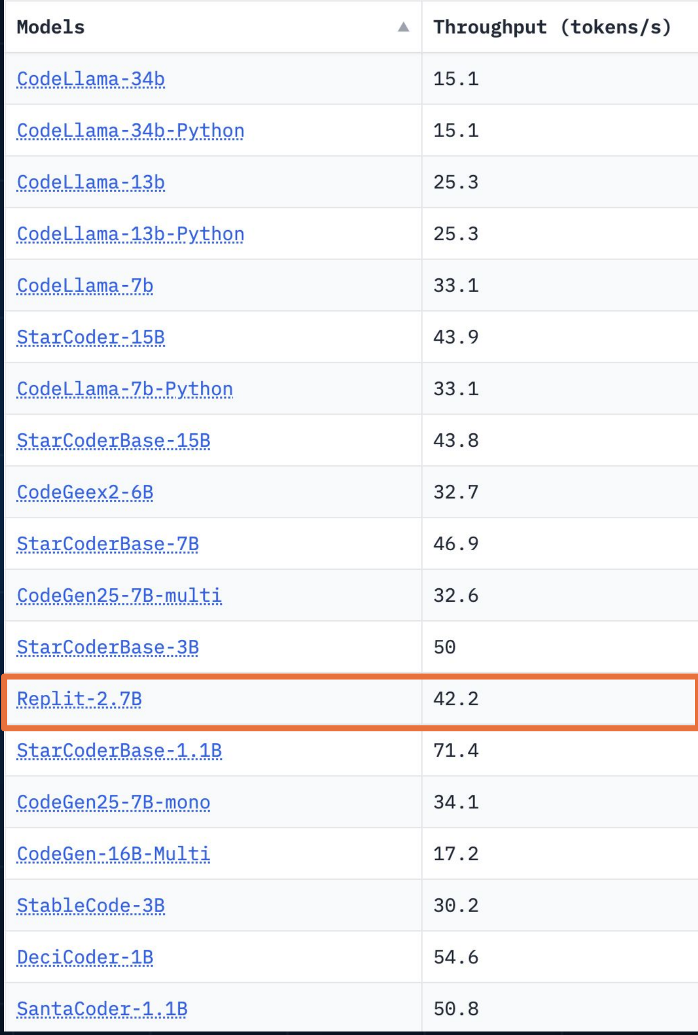
~ 200 tokens / s on a single A100-40G (no batching)

We made explicit architectural choices to support:

- https://github.com/NVIDIA/FasterTransformer
- https://github.com/triton-inference-server

for optimized inference on NVIDIA GPUs

Reliable inference evaluation across model architectures is still really **HARD**



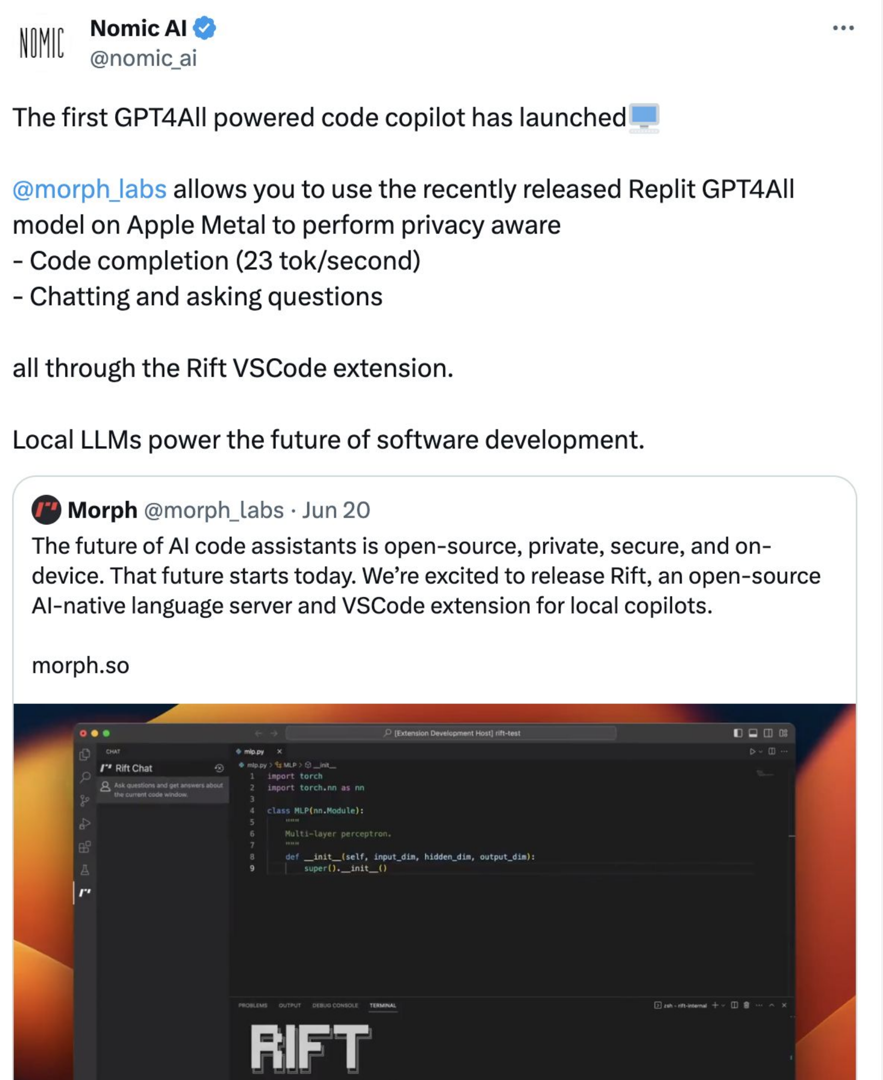
Models	Throughput (tokens/s)
CodeLlama-34b	15.1
CodeLlama-34b-Python	15.1
CodeLlama-13b	25.3
CodeLlama-13b-Python	25.3
CodeLlama-7b	33.1
StarCoder-15B	43.9
CodeLlama-7b-Python	33.1
StarCoderBase-15B	43.8
CodeGeex2-6B	32.7
StarCoderBase-7B	46.9
CodeGen25-7B-multi	32.6
StarCoderBase-3B	50
Replit-2.7B	42.2
StarCoderBase-1.1B	71.4
CodeGen25-7B-mono	34.1
CodeGen-16B-Multi	17.2
StableCode-3B	30.2
DeciCoder-1B	54.6
SantaCoder-1.1B	50.8

https://huggingface.co/spaces/bigcode/multilingual-code-evals

- Since the open-source release, a lot of interesting projects spun up from replit-code-v1-3b

- Instruct fine tuned on CodeAlpaca and GPTeacher Code-Instruct: https://huggingface.co/teknium/Replit-v2-CodeInstruct-3B

- Quantization + ggml support to boost local inference for VSCode plugins



Nomic AI
@nomic_ai

The first GPT4All powered code copilot has launched💻

@morph_labs allows you to use the recently released Replit GPT4All model on Apple Metal to perform privacy aware
- Code completion (23 tok/second)
- Chatting and asking questions

all through the Rift VSCode extension.

Local LLMs power the future of software development.

Morph @morph_labs · Jun 20
The future of AI code assistants is open-source, private, secure, and on-device. That future starts today. We're excited to release Rift, an open-source AI-native language server and VSCode extension for local copilots.

morph.so

[Extension Development Host] rift-test
CHAT
Rift Chat
Ask questions and get answers about the current code window.
mlp.py
mlp.py > MLP > __init__
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
RIFT

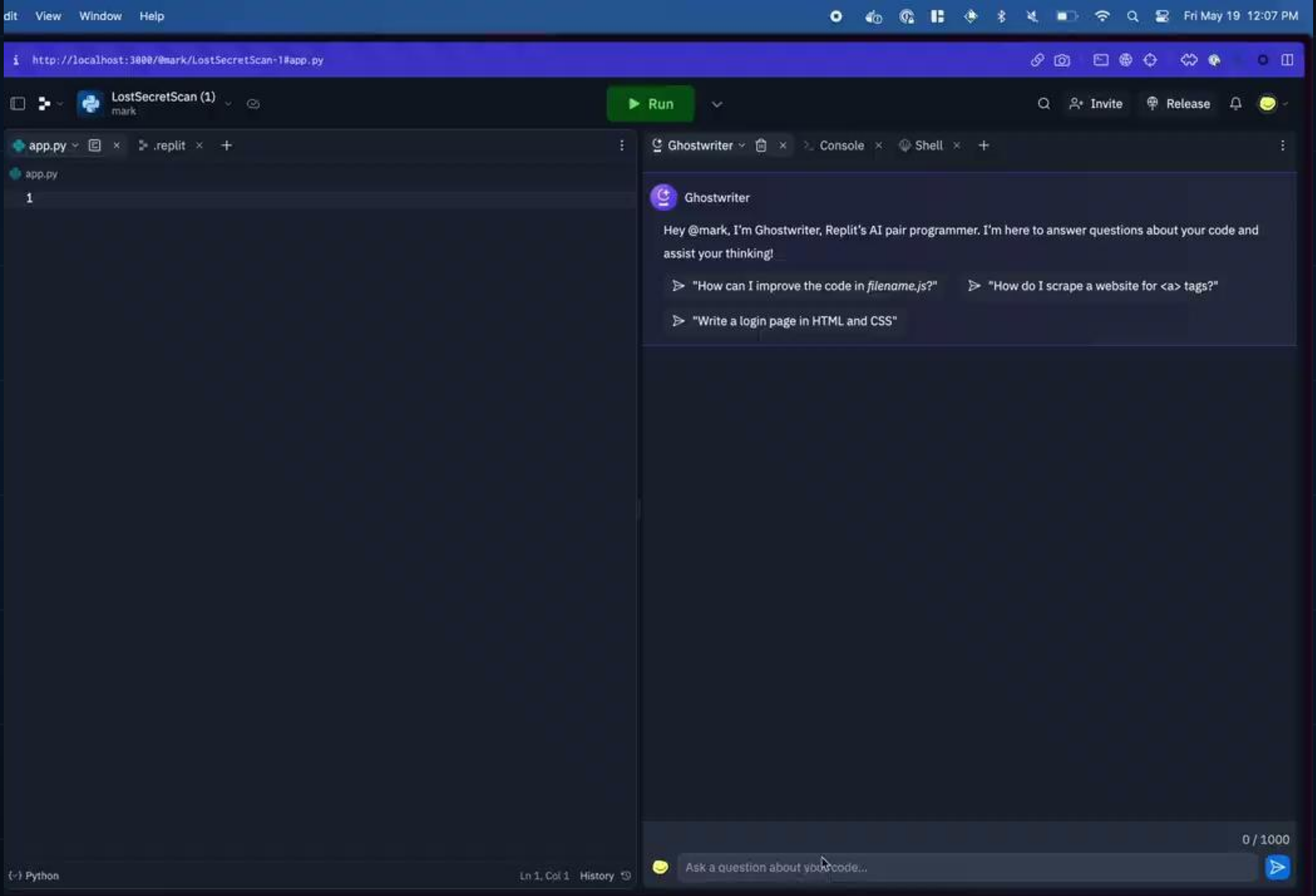
```
import torch
import torch.nn as nn

class MLP(nn.Module):
    """
    Multi-layer perceptron.
    """
    def __init__(self, input_dim, hidden_dim, output_dim):
        super().__init__()
```

Links

https://github.com/replit/ReplitLM
https://huggingface.co/replit/replit-code-v1-3b
https://blog.replit.com/llm-training

Acknowledgements

- Madhav Singhal, Juan Sigler Priego, Bradley Heilbrun, Samip Dahal, Giuseppe Burtini, Reza Shabani, Amjad Masad & the whole Replit team
- Jonathan Frankle, Hanling Tang, Abhinav Venigalla, Vitaliy Chiley, Alexander Trott, Daya Khudia, Scott Sovine, Barry Dauber, Naveen Rao & the whole MosaicML team



Ghostwriter

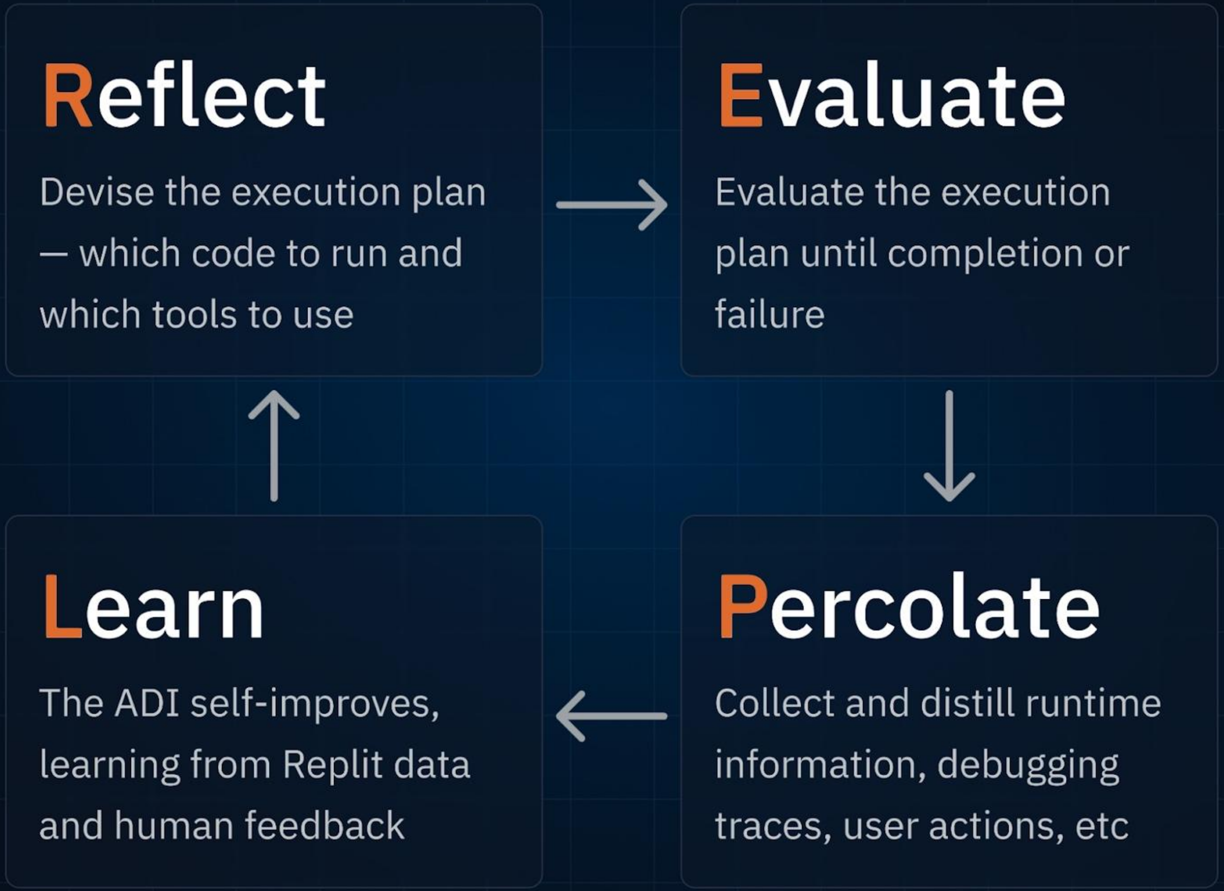
Hey @mark, I'm Ghostwriter, Replit's AI pair programmer. I'm here to answer questions about your code and assist your thinking!

"How can I improve the code in *filename.js*?"

"How do I scrape a website for <a> tags?"

"Write a login page in HTML and CSS"

Artificial Developer Intelligence



Reflect

Devise the execution plan — which code to run and which tools to use

Evaluate

Evaluate the execution plan until completion or failure

Percolate

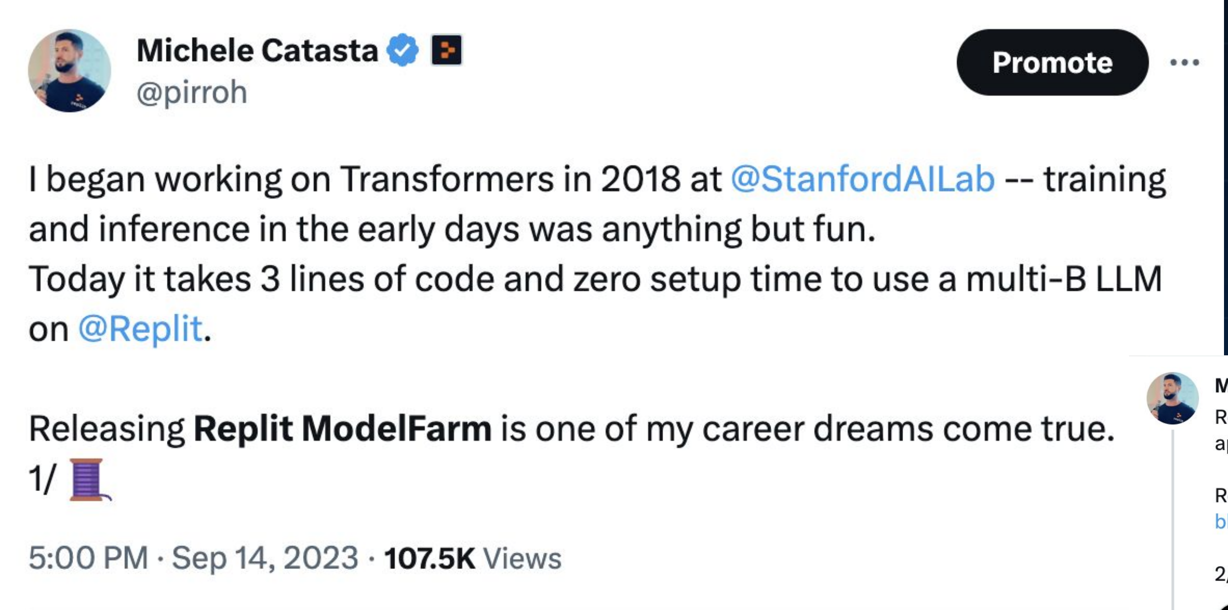
Collect and distill runtime information, debugging traces, user actions, etc

Learn

The ADI self-improves, learning from Replit data and human feedback

https://blog.replit.com/replit-ai-manifesto

Replit ModelFarm

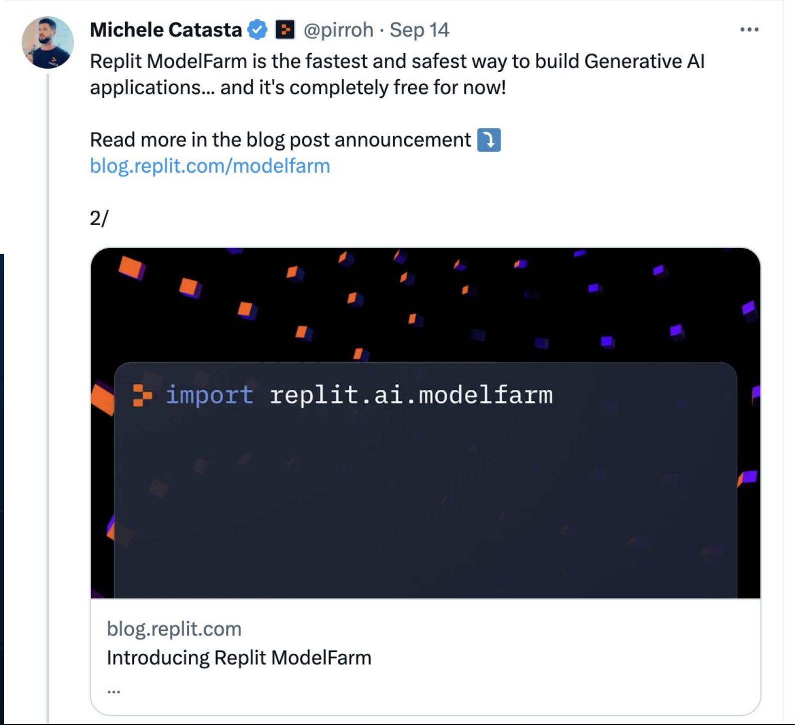


Michele Catasta
@pirroh

I began working on Transformers in 2018 at @StanfordAILab -- training and inference in the early days was anything but fun.
Today it takes 3 lines of code and zero setup time to use a multi-B LLM on @Replit.

Releasing **Replit ModelFarm** is one of my career dreams come true.
1/ 🧵

5:00 PM · Sep 14, 2023 · **107.5K** Views



Michele Catasta @pirroh · Sep 14

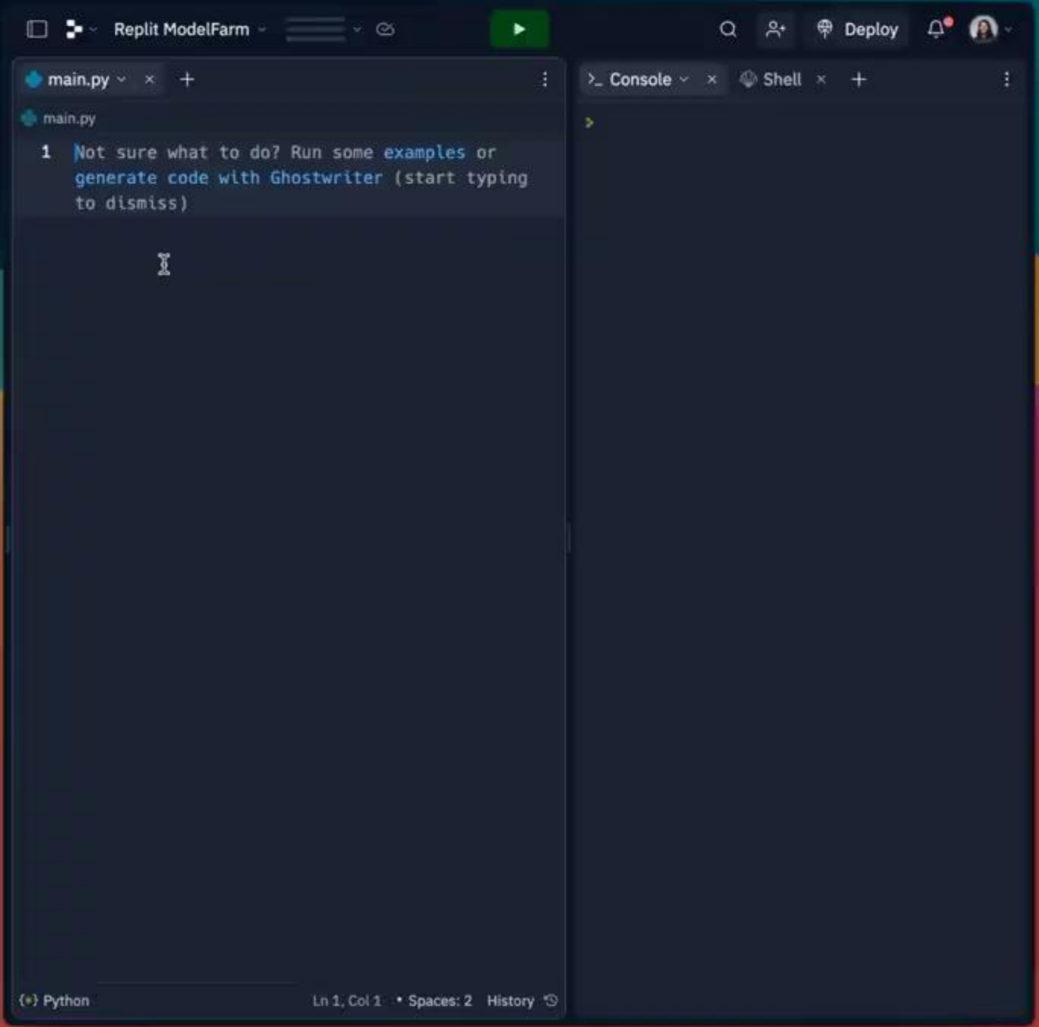
Replit ModelFarm is the fastest and safest way to build Generative AI applications... and it's completely free for now!

Read more in the blog post announcement ⤵️
blog.replit.com/modelfarm

2/

```
import replit.ai.modelfarm
```

blog.replit.com
Introducing Replit ModelFarm
...



Not sure what to do? Run some examples or generate code with Ghostwriter (start typing to dismiss)

Thank you!

Michele Catasta

https://twitter.com/pirroh

https://pirroh.fyi