



**The Open-Source
Meeting in Sofia**



Friendly reminder:

The choice of Math, ML, and AI topics we can discuss is endless.


We have one evening and will start with only the ultra hot open-source topics.

Why all the hype?



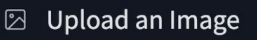
Microsoft Edge File Edit View History Favorites Tools Profiles Tab Window Help 100% mitko Tue Mar 5 13:12

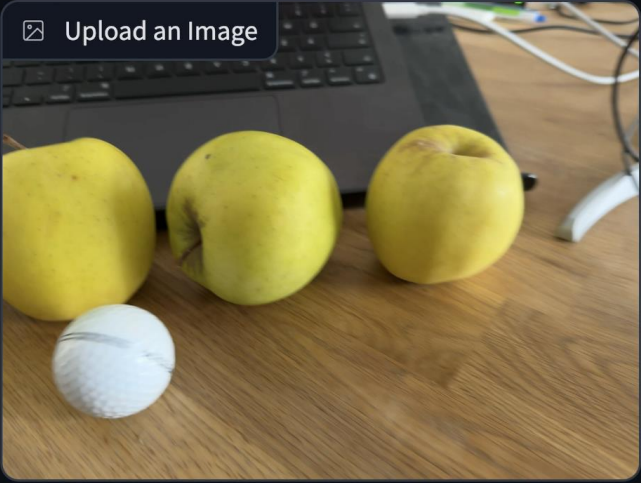
localhost:7860

 **moondream**

Prompt

How many apples are there?

 Upload an Image



3

Draft Agenda



- ❑ Practical open-source AI resources - datasets, tools, models
- ❑ How to start on your PC today
- ❑ Open AI platform architectures - from on-device to hybrid local/remote
- ❑ From PoC to pilot to production - Edge to Cloud AI platforms
- ❑ End-2-End performance optimization
- ❑ Security for AI platforms
- ❑ Beyond the wrappers, RAG, and prompt engineering - advanced AI systems engineering
- ❑ Practical use cases

AI/ML did not happen overnight



Prehistoric

- 1950s Machine Translation

Stone Age

- 1980s Knowledge-Based Systems

Bronze Age

- 1993 – 2012 - Statistical Era

Iron Age

- 2013 – 2017 - Special Purpose, Deep Learning ML

Modern Age

- 2018 – Present – Generative AI, Foundation Models, LLMs

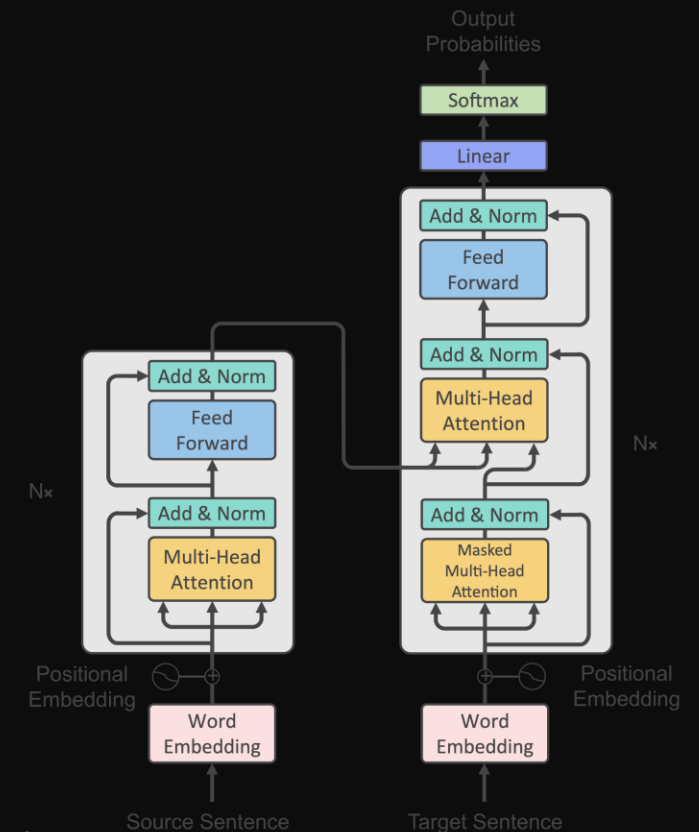
Transformers Era



2017 – Present

“Attention is All You Need paper” in 2017

- ❑ Type of Deep Neural Network
- ❑ Leverages Attention/Self-Attention, including multi-head attention
- ❑ **Expressive:** Feed-forward;
- ❑ **Optimizable:** Backpropagation, Gradient Descent;
- ❑ **Efficient:** High Parallelism compute graph
- ❑ Examples: Mixtral-8x22B, LLaMA-2, phi-2, GPT-4, Claude-3
- ❑ Learn all from Karpathy https://www.youtube.com/watch?v=zjkBMFhNj_g



Brief New Age Tech Glossary



- ❑ **Transformers:** A type of general-purpose neural network architecture that facilitates the modeling of sequences without the need for recurrent connections, prominently used in language processing tasks
- ❑ **Foundational Model:** A large-scale model that is trained on vast amounts of data and can be fine-tuned for a variety of downstream tasks, serving as a base for further specialized models
- ❑ **Large Language Model:** A substantial neural network model trained on extensive textual data to understand and generate human-like text across many languages and contexts. **Small Language Model:** A more compact version of a language model designed for efficiency and lower resource consumption while performing natural language processing tasks
- ❑ **Visual Language Model:** A model that combines language and vision processing to understand and generate content related to both text and images
- ❑ **Multimodal Models:** AI models that can process and understand information from different types of data, such as text, images, and audio, simultaneously
- ❑ **RWKV (RwaKuv):** A variant of a recurrent neural network, which stands for "Reduced Weight KneeV", designed for efficiency and performance in sequence modeling tasks
- ❑ **Mamba/Jamba, Hawk/Griffin, DPO, DORPO, Flash Attention...**

Where does open-source AI live



<https://HuggingFace.co> – models, data, research papers, AI social network, compute...

<https://arXiv.org> - Research Papers

<https://Github.com> – All the source code in one Place

<https://github.com/ggerganov/llama.cpp> - local AI on your CPU, GPU „Хайде наште!“

<https://Discord.com> – almost all projects have a channel

<https://x.com> – social network for emerging AI/ML devs, researchers, companies

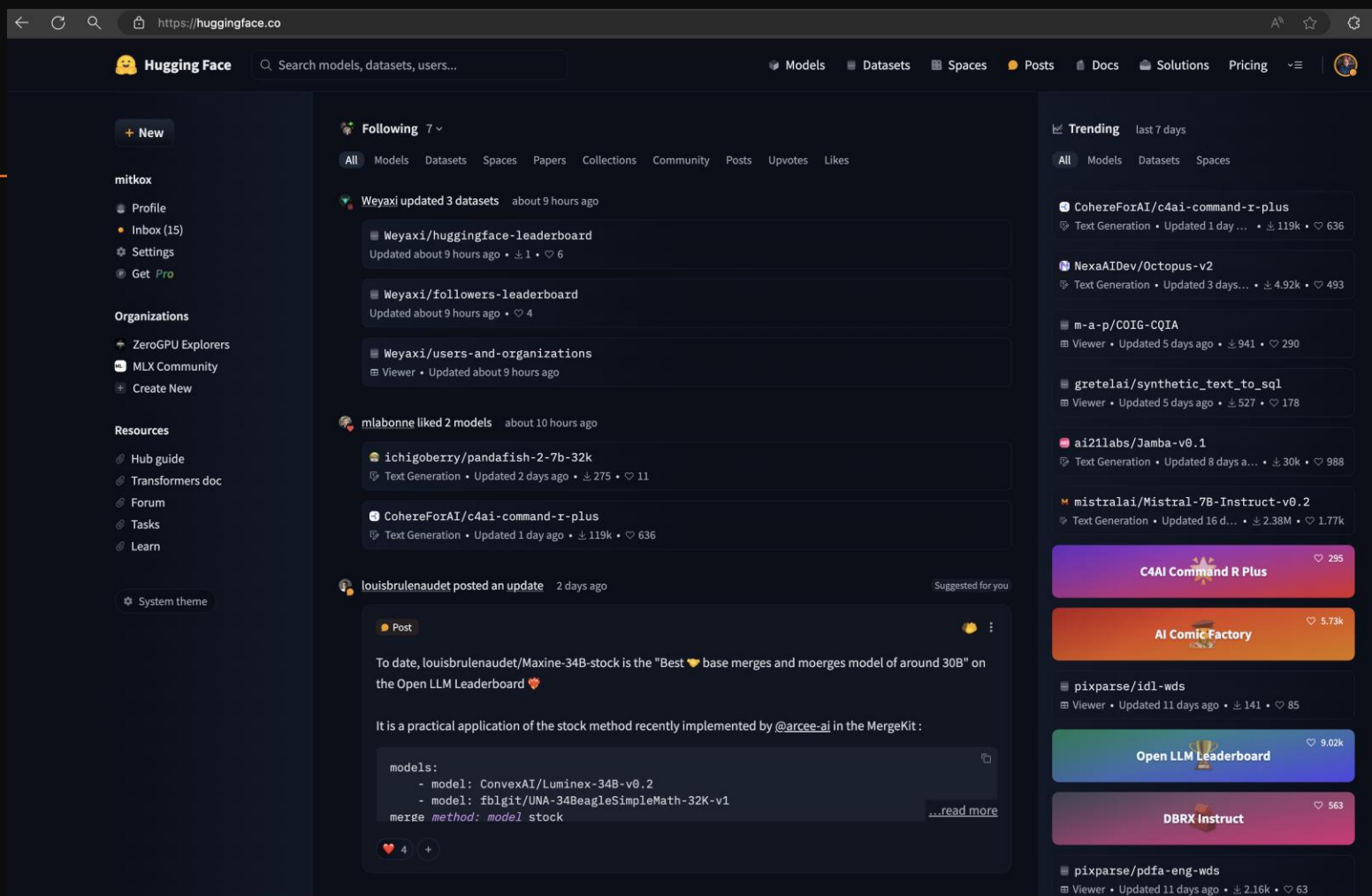
<https://colab.research.google.com/> - ‘Free’ compute and managed Jupyter notebooks



WTF is Hugging Face?



Hugging Face: The home of open AI/ML



Founded In
2016

170
Employees

300K+ stars on Github
600K+ open source models

130K+

public data sets

1M+

daily downloads

700K+

daily visitors

30+

Libraries

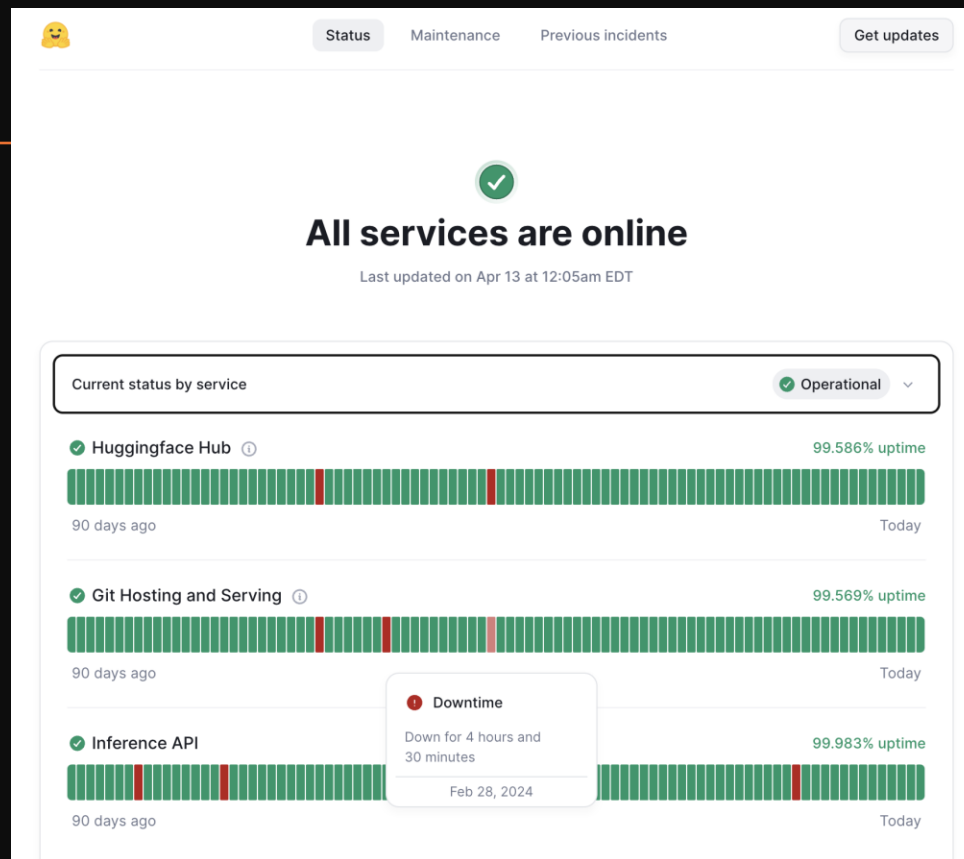
When the AI World Stopped



Founded In
2016

170
Employees

300K+ stars on Github
600K+ open source models



130K+

public data sets

1M+

daily downloads

700K+

daily visitors

30+

Libraries

Used everywhere in the AI world



15,000+ startups and enterprises



Open-source contributors



Cloud partners



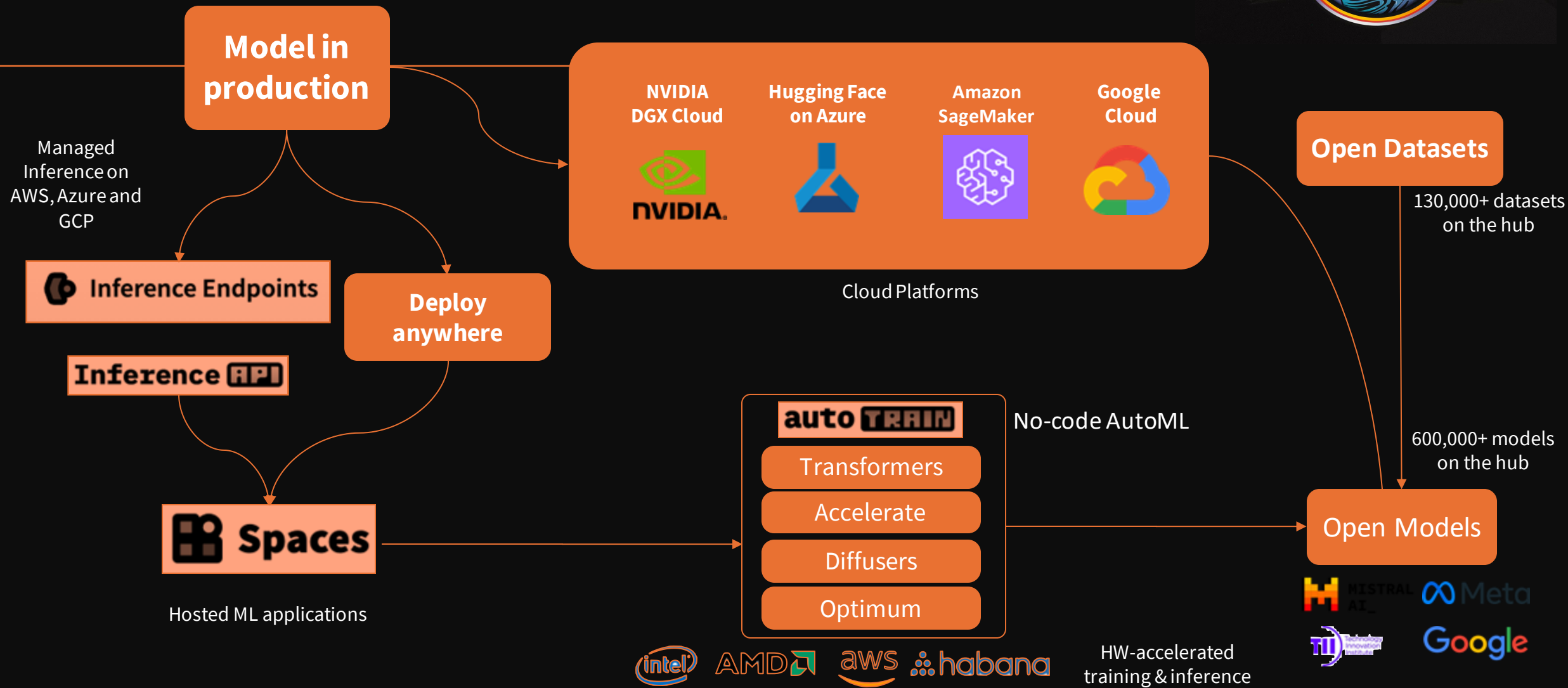
Hardware partners



On-prem partners



The Global AI/ML Ecosystem of 🤖



Open-Source Ecosystem



- **Transformers**

State-of-the-art ML for Pytorch, TensorFlow, and JAX.

- **Datasets**

Access and share datasets for computer vision, audio, and NLP tasks.

- **Gradio**

Build machine learning demos and other web apps, in just a few lines of Python.

- **Safetensors**

Simple, safe way to store and distribute neural networks weights safely and quickly.

- **Transformers.js**

Community library to run pretrained models from Transformers in your browser.

- **Hub Python Library**

Client library for the HF Hub: manage repositories from your Python runtime.

- **Diffusers**

State-of-the-art diffusion models for image and audio generation in PyTorch.

- **Accelerate**

Easily train and use PyTorch models with multi-GPU, TPU, mixed-precision.

- **TRL**

Train transformer language models with reinforcement learning.

- **timm**

State-of-the-art computer vision models, layers, optimizers, training/evaluation, and utilities.

- **PEFT**

Parameter efficient finetuning methods for large models

Back to OSS license!









- **Text Generation Inference**

Toolkit to serve Large Language Models.

Open vs Closed Models



Open and closed models have different benefits and should be considered for each use-case

	Open-Source	Closed/Proprietary
Security	Models can be self-hosted , data stays in your environment	Models cannot be self-hosted. Data is sent outside your environment to vendor(s)
Control	The lifecycle is controlled by you	Updates and changes to performance are happening without notice
Customization	Open Weights and sometimes open code access to customize the model for your needs	Limited ability to customize for your needs
Transparency	Inspect code and data provides better auditability and understandability	No ability to audit or understand performance
Cost	Typical lower long term cost due to smaller model size	larger model size and proprietary premium often balanced by decreased cost from server-side optimization
Latency	Lower latency due to on premise and smaller model sizes	Often greater latency due to larger model sizes + API latency
Quality	No single approach is best. Each use case will vary. Proprietary is typically closer to the frontier of performance .	
Examples	 OpenAI  Meta  salesforce  FLAN-T5  MISTRAL AI  Microsoft	 OpenAI  ANTHROPIC

Energy/carbon footprint and LLMs



Start by test existing models on your domain and task(s) of interest

Most of the time the answer is “no” =>

Focus on **efficient fine-tuning and inference**

Do you **need** to pretrain an LLM?

Yes =>

Focus on **efficient pretraining** while taking a holistic view of **model life-cycle**

Energy budget will likely be dominated by inference costs.

Select a **compute efficient** model:

- smallest size
- quantized
- classification models > generative

Deploy it in an on-prem setup/cloud provider in a region with a **good energy mix**

Train-compute-optimal models (Chinchilla law) are not efficient for inference

Train **smaller models** for longer if you plan to deploy it at large scale

Train in a local cluster/provider with a **good energy mix**

Share the model so people can **re-use**/leverage the compute spent – *It's like recycling AI models*

Ressources:

- Power Hungry Processing: Watts Driving the Cost of AI Deployment? <https://arxiv.org/abs/2311.16863>
- Language models scale reliably with over-training and on downstream tasks <https://arxiv.org/abs/2403.08540>
- Region energy mix (e.g. solar, nuclear, coal, gas) can have a x500 impact on model carbon footprint: <https://app.electricitymaps.com/>



How to start on your PC today

llama.cpp (Made in Sofia)



A screenshot of the llama.cpp GitHub repository page. The page shows the repository name "llama.cpp" with 505 watches, 7.8k forks, and 55.1k stars. It lists 212 branches and 1,686 tags. The file list includes folders like .devops, .github, ci, cmake, common, docs, examples, ggml-cuda, gguf-py, grammars, kompute, kompute-shaders, media, models, pocs, prompts, requirements, scripts, spm-headers, and tests. The right sidebar shows the "About" section with "LLM inference in C/C++", "Releases" (1,661), "Packages" (1), "Contributors" (664), and "Languages" (C++ 73.6%, C 13.2%, Python 3.8%, Metal 2.0%, Objective-C 1.5%).

LLaMA^{C++}

Started In
2023

664

Contributors

55.1K+ 180+

stars on Github

Active PRs

50+

Other project Integrations

40+

Examples

7.8K+

Forks

MLX (Apple owned Open Source)



Started In
2023

95
Contributors

13.8K+

stars on Github

10+

Active PRs

A screenshot of the MLX GitHub repository page. The page shows the repository name "mlx" under the organization "ml-explore". It displays statistics such as 126 watchers, 780 forks, and 13.8k stars. The main content area shows a list of files and folders, including ".circleci", ".github", "benchmarks", "cmake", "docs", "examples", "mix", "python", "tests", ".clang-format", ".gitignore", ".pre-commit-config.yaml", "ACKNOWLEDGMENTS.md", "CMakeLists.txt", "CODE_OF_CONDUCT.md", "CONTRIBUTING.md", "LICENSE", "MANIFEST.in", "README.md", "mlx.pc.in", and "pyproject.toml". The right sidebar shows the "About" section with the description "MLX: An array framework for Apple silicon" and various repository statistics.

20+

Other project Integrations

15+

Examples

770+

Forks

One Click Tools



https://gpt4all.io/index.html

NOMIC

GPT4All

A free-to-use, locally running, privacy-aware chatbot. **No GPU or internet required.**

GPT4All Model (gpt4all-j)

Write a poem about a large language model that runs on my laptop.

In a world where technology is king,
A large language model was a thing.

Send a message

Real-time inference latency on an M1 Mac

Download Desktop Chat Client

Windows Installer | OSX Installer | Ubuntu Installer

https://lmstudio.ai

We're hiring TypeScript and C++ Engineers

LM Studio

Discover, download, and run local LLMs

Run any LLaMa, Falcon, MPT, Gemma, Repit, GPT-Neo-X, gguf models from Hugging Face

Technology Preview: LM Studio 0.2.19 with AMD ROCm

Download LM Studio for M1/M2/M3 0.2.19

Download LM Studio for Windows 0.2.19

Download LM Studio for Linux (Beta) 0.2.19

LM Studio is provided under the terms of use.

Sign up for new version email updates

Twitter | GitHub | Discord | Email

With LM Studio, you can ...

- Run LLMs on your laptop, entirely offline
- Use models through the in-app Chat UI or an OpenAI compatible local server
- Download any compatible model files from HuggingFace repositories
- Discover new & noteworthy LLMs in the app's home page

LM Studio supports any gguf, Llama, MPT, and StarCoder model on Hugging Face (Llama 2, Orca, Vicuna, Nous Hermes, WizardCoder, MPT, etc.)

GPT4All's Capabilities

Explore what GPT4All can do. On your own ha

https://pinokio.computer

pinokio

Install, Run & Control Bots on Your Computer with 1 Click.

Pinokio is a browser that lets you install, run, and programmatically control ANY application, automatically.

Download | Explore | Learn

VIRTUAL COMPUTER

FILE SYSTEM CPU MEMORY

Watch Later | Share



Open AI Platform Architecture

Open AI Platform



Your Own Data



Data Lakehouse
Delta Lake, Spark,
Trino

Embeddings Model
E5 Mistral

Embeddings DB
Quant

IaC

DevSecOps

Model Factory
Prompt
Programming

Playground
DSPy

Guardrails
DSPy

Orchestration
Routing
DSPy

APIs/Plugins
Open
CodeInterpreter

Vault
OpenBao

Query/
API Call

Hybrid Identity Service
Keycloak

Output/
API Call

Local Orchestration/Router DSPy, AICI	In-mem, Databases Memmap, pgvector
Base Model/Function calling Hermes-2-pro	Lifecycle /Control Plane Agent Rust

App/API/Inference on CPU and GPU
Local Platform on Computer/Edge GW/Phone

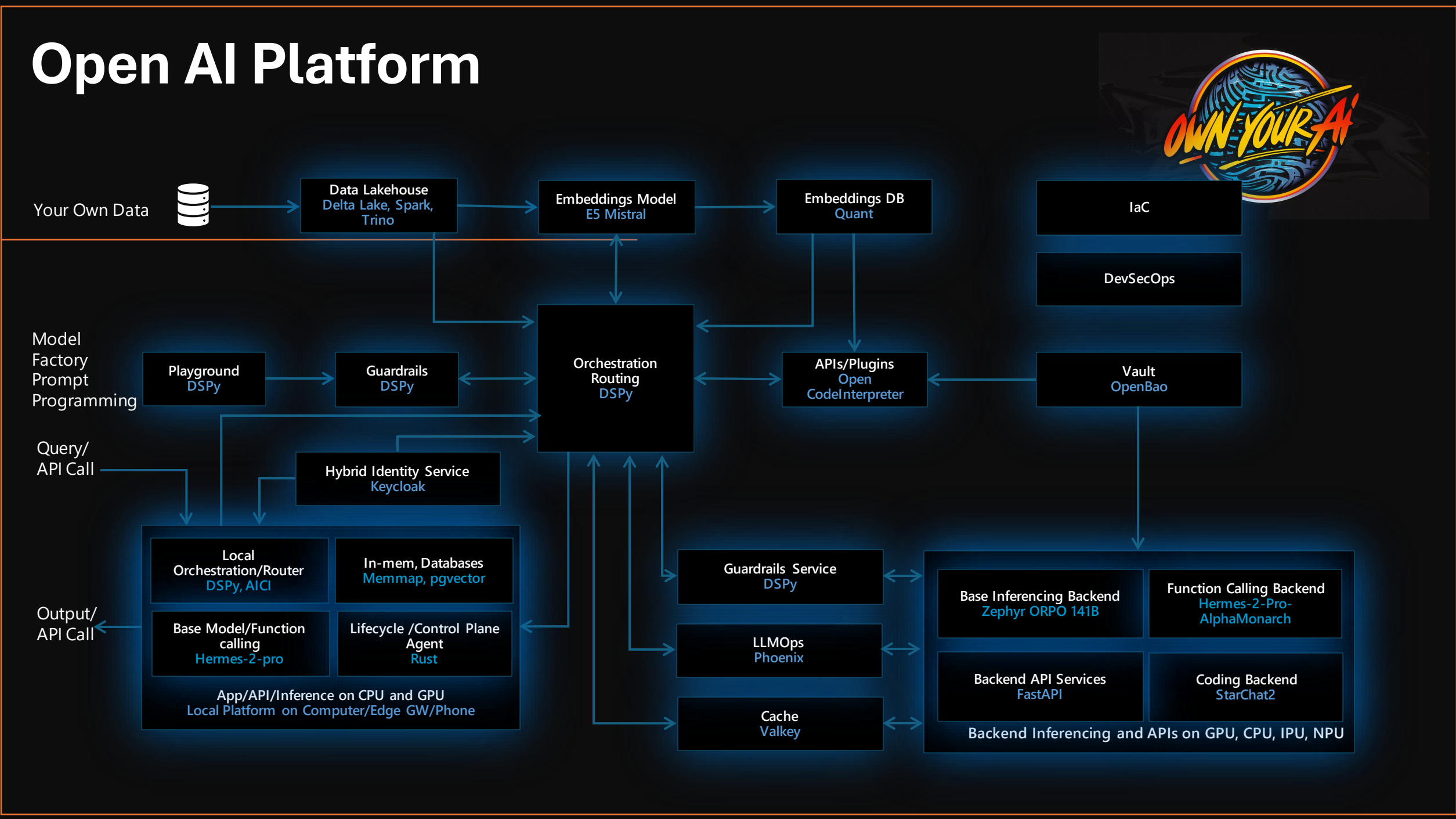
Guardrails Service
DSPy

LLMOps
Phoenix

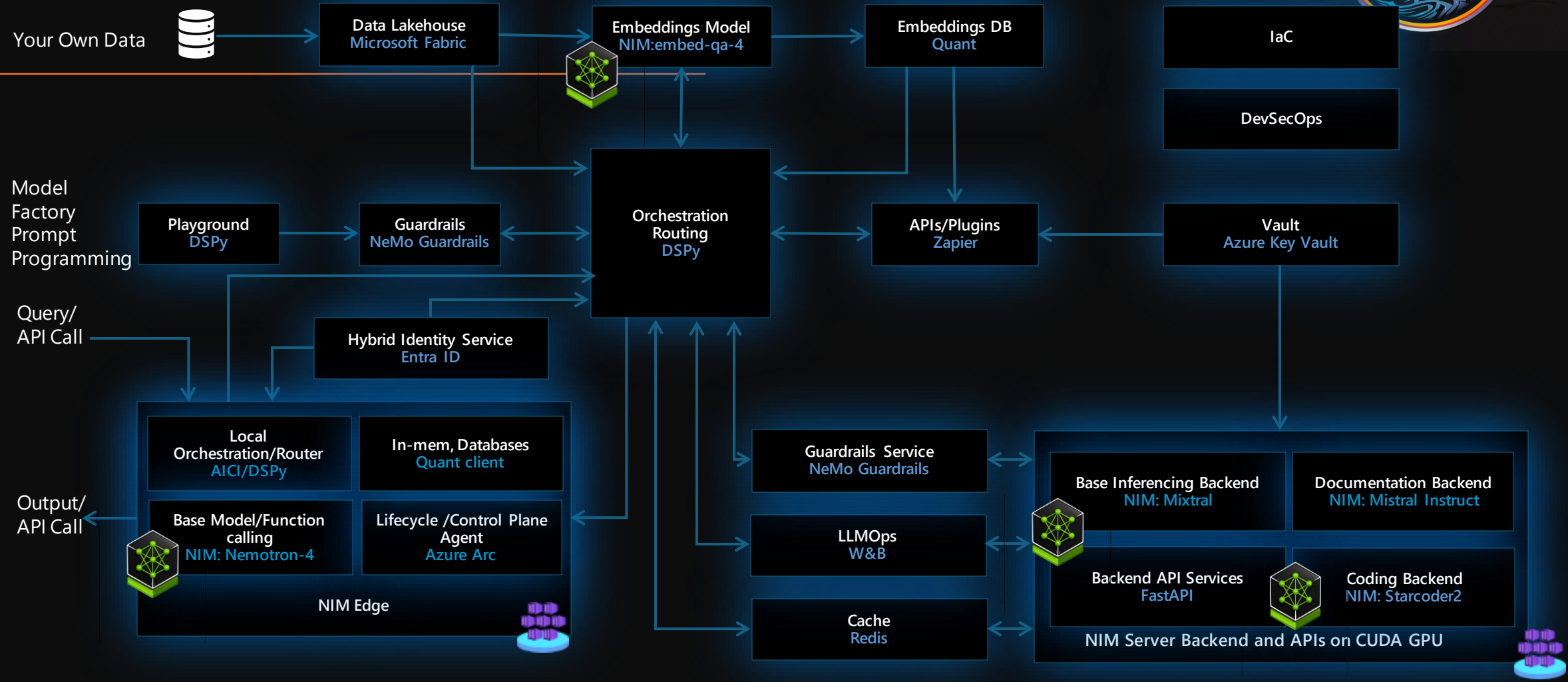
Cache
Valkey

Base Inferencing Backend Zephyr ORPO 141B	Function Calling Backend Hermes-2-Pro- AlphaMonarch
Backend API Services FastAPI	Coding Backend StarChat2

Backend Inferencing and APIs on GPU, CPU, IPU, NPU



Production Ready with Vendors Lock-in





From Proof of Concept to Pilot to Production

Lessons Learned



- ❑ Set **expectations**
- ❑ Minimize risks
- ❑ Always experiment and build with the North Star to take it to **production**
- ❑ Work 3x faster from product start to launch to happen in **6 months**

Set Expectations



Building cool demos with GenAI is **easy**

Building an industrial or enterprise product with GenAI is **hard**

- ❑ If you want cool demos to show everyone externally that you're ahead of the curve, just do it!
- ❑ If you want your team to experiment and build out AI muscles for production, just do it!
- ❑ If you want a product, build data, get compute and train talents to build it, and just do it!

There are a lot of things GenAI can do



Q: But can these things meaningfully transform your customers' business?

A: Unclear

There are a lot of things Generative AI can't do NOW



Q: But would GenAI still not be able to do those in the future?

A: Unclear

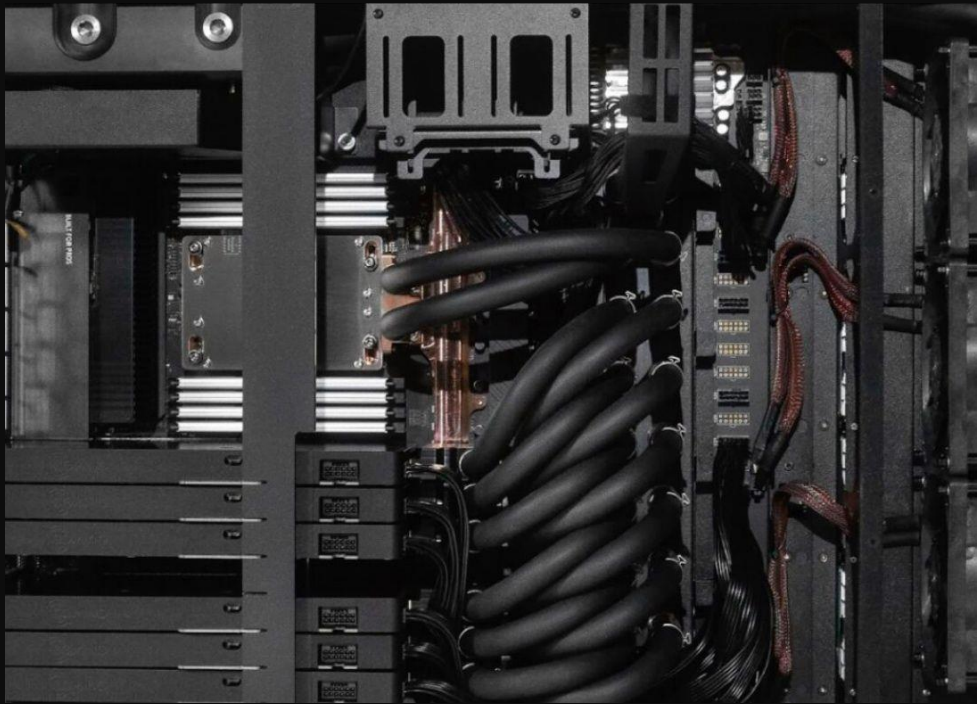
“When a distinguished but elderly scientist states that something is possible, he is almost certainly right. When he states that something is impossible, he is very probably wrong.”

- Arthur Clarke



End-2-End Performance Optimization

Local AI Platform - Which Way?

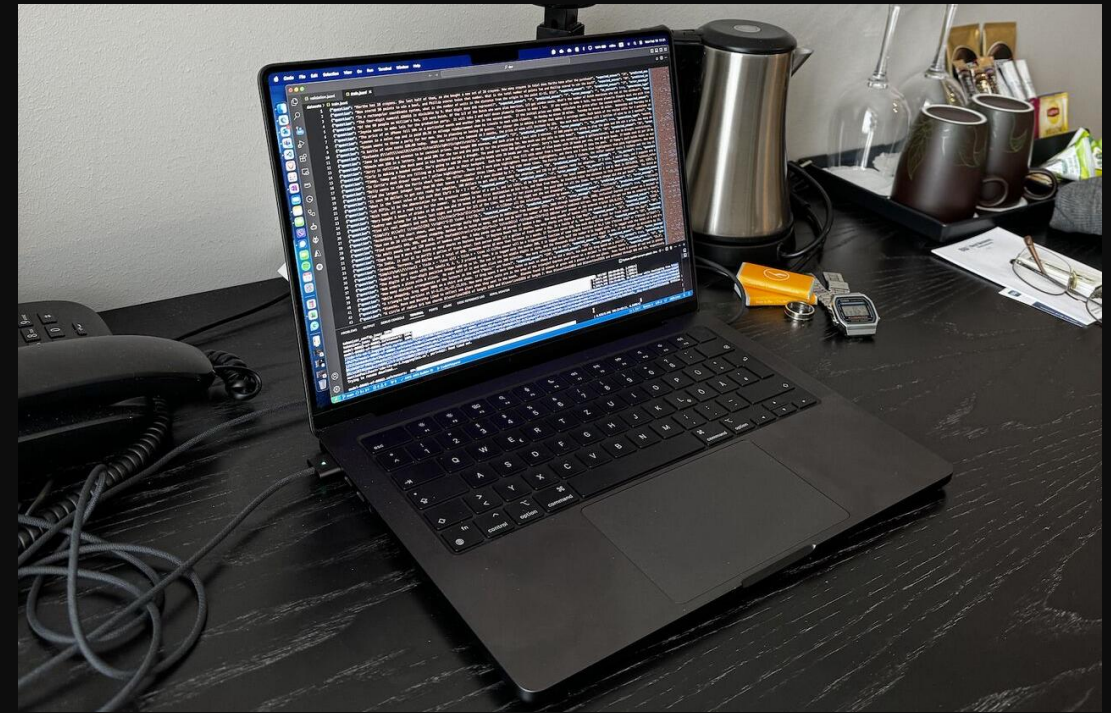


Recommended Enthusiasts Hardware:

Ryzen CPU
64GB RAM
3090-RTX
1TB SSD

Recommended Pros Hardware:

Ryzen CPU
256GB RAM
6x4090-RTX with P2P Kernel
4TB SSD



Recommended Enthusiasts Hardware:

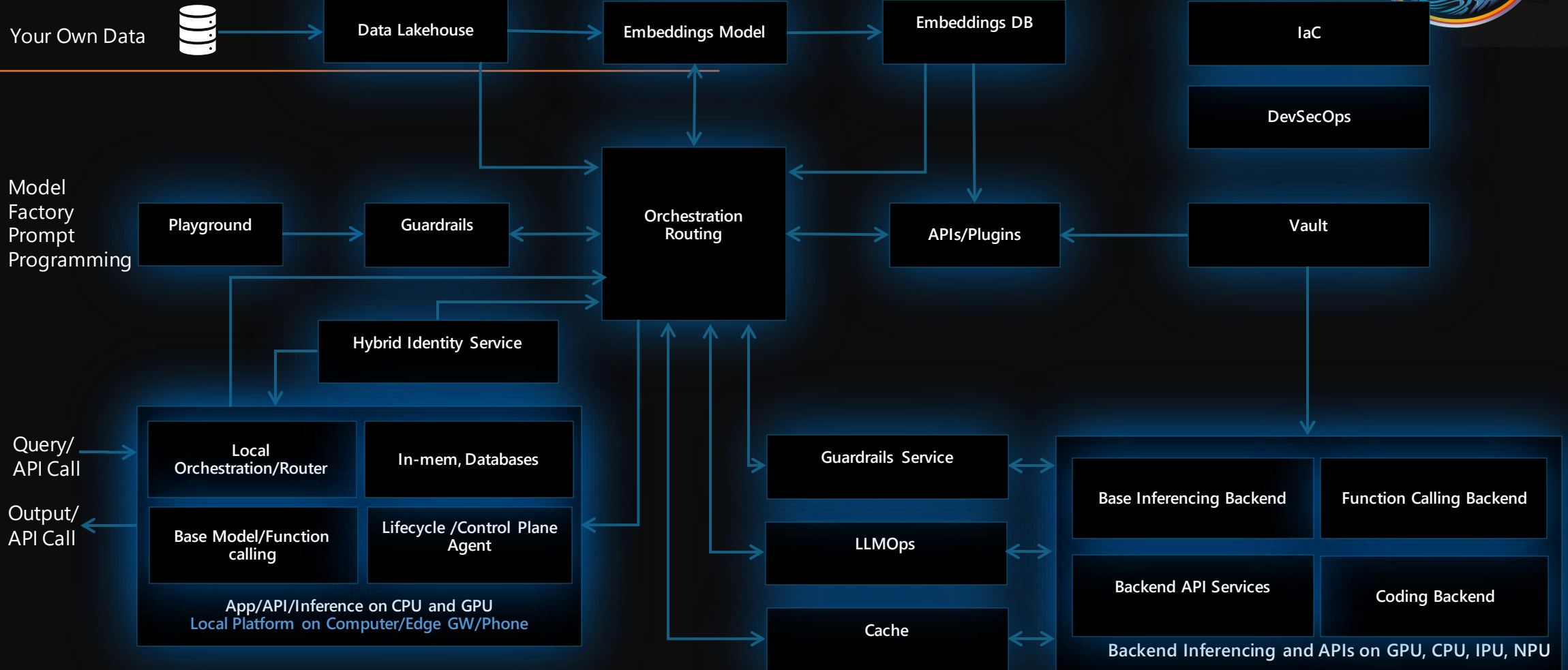
MacBook M2/M3
16GB RAM
1TB SSD

Recommended Pros Hardware:

MacBook M3 Max
128GB RAM
4TB SSD

When Local AI Platform is not Enough

Build Your Own Multi-user AI Platform



Lessons Learned



1. Choose the Best Models for your Use Cases
2. Balance fine-tuning and Data Pipelines
3. Activate only when you need
4. Inference Quantized Models
5. Use all available Hardware – Edge to Core
6. Make everything OSS Plug&Play
7. Shorten the Lifecycle of PoC-Pilot-Production


Compute Everywhere when Learning



https://deepinfra.com/mistralai/Mixtral-8x22B-v0.1

deepinfra Models Docs Pricing Chat Compare Blog [email] [chat] Log In

Models > Text Generation > mistralai/Mixtral-8x22B-v0.1

featured  **mistralai/Mixtral-8x22B-v0.1** Copy

Mixtral-8x22B is the latest and largest mixture of expert large language model (LLM) from Mistral AI. This is state of the art machine learning model using a mixture 8 of experts (MoE) 22b models. During inference 2 experts are selected. This architecture allows large models to be fast and cheap at inference. This model is not instruction tuned.

Public **\$0.65 / Mtoken** 64k

DEMO API VERSIONS

Input

Input *
I have this dream
text to generate from

Output
I have this dream about the day I got a job at a tech company. I just woke up on a plane. I sat down on the floor and started getting work done. After getting up around 6 p.m., I looked around and



Open AI Platform Security

All Cybersecurity Best Practices Plus... Meta Prompts, Grounding, ASCII, DSPy red teaming...



The screenshot shows the ImHex hex editor interface. The main window displays a hex dump of memory data with corresponding ASCII characters. The Data Inspector on the right shows a list of detected metadata fields, including binary data, integers, floats, and strings. The Disassembler on the far right shows the assembly code for the selected memory location.

Name	Value
Binary (8 bit)	0b00000000
uint8_t	0
int8_t	0
uint16_t	0
int16_t	0
uint32_t	8388608
int32_t	-8388608
uint64_t	8388608
int64_t	-8388608
half float (16 bit)	0
float (32 bit)	1.17549E-38
double (64 bit)	1.19511E-310
long double (128 bit)	8.81747E-4938
Signed LEB128	0
Unsigned LEB128	0
bool	false
ASCII Character	'NULL'
Wide Character	'Invalid'
UTF-8 code point	'NULL' (U+0000)
String	"\x00"
Wide String	L''
time_t	Sun, 07. 04. 768497 18:28:08
DOS Date	0/0/1980
DOS Time	00:00:00
GUID	{00000000-1600-0000-0000-000000000000}
RGBA Color	
RGB565 Color	

Name	Color	Start	End	Size	Type	Value
string	Green	0x0000020	0x0000033	0x0014	String	"general.architecture"
type	Yellow	0x0000034	0x0000037	0x0004	enum gguf_metadata_value_type	gguf_metadata_value_type::GGUF
len	Yellow	0x0000038	0x000003F	0x0008	u64	5 (0x0000000000000005)
string	Red	0x0000040	0x0000044	0x0005	String	"llama"
len	Red	0x0000045	0x000004C	0x0008	u64	12 (0x000000000000000C)
string	Red	0x000004D	0x0000050	0x0005	String	"general.name"
type	Yellow	0x0000051	0x000005C	0x0004	enum gguf_metadata_value_type	gguf_metadata_value_type::GGUF
len	Yellow	0x000005D	0x0000064	0x0008	u64	5 (0x0000000000000005)
string	Red	0x0000065	0x0000069	0x0005	String	"jaffa"
len	Red	0x000006A	0x0000071	0x0008	u64	20 (0x0000000000000014)
string	Red	0x0000072	0x0000085	0x0014	String	"llama.context_length"
type	Yellow	0x0000086	0x0000089	0x0004	enum gguf_metadata_value_type	gguf_metadata_value_type::GGUF
value	Yellow	0x000008A	0x000008D	0x0004	enum gguf_metadata_value_type	gguf_metadata_value_type::GGUF
len	Yellow	0x000008E	0x0000095	0x0008	u64	32768 (0x00000000)
string	Red	0x0000096	0x00000AB	0x0016	String	"llama.embedding_length"



Beyond the wrappers, RAG and Prompt Engineering - Advanced AI Systems Engineering

Lifecycle of an AI Model



❑ **Training:**

Data preparation

Efficient training techniques

Evaluation

❑ **Fine-tuning:**

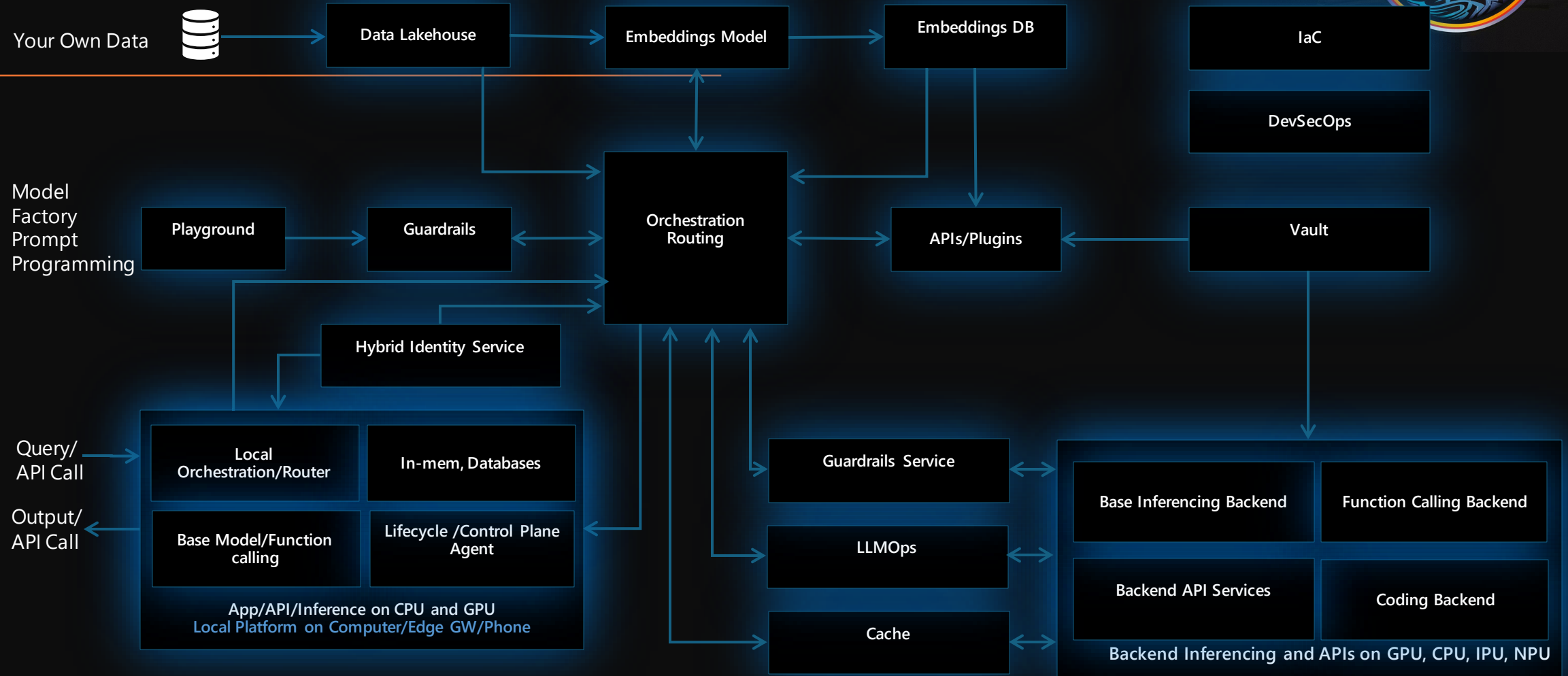
RLHF, RLAIIF

❑ **Inference:**

Quantization

Deployment

System of Systems

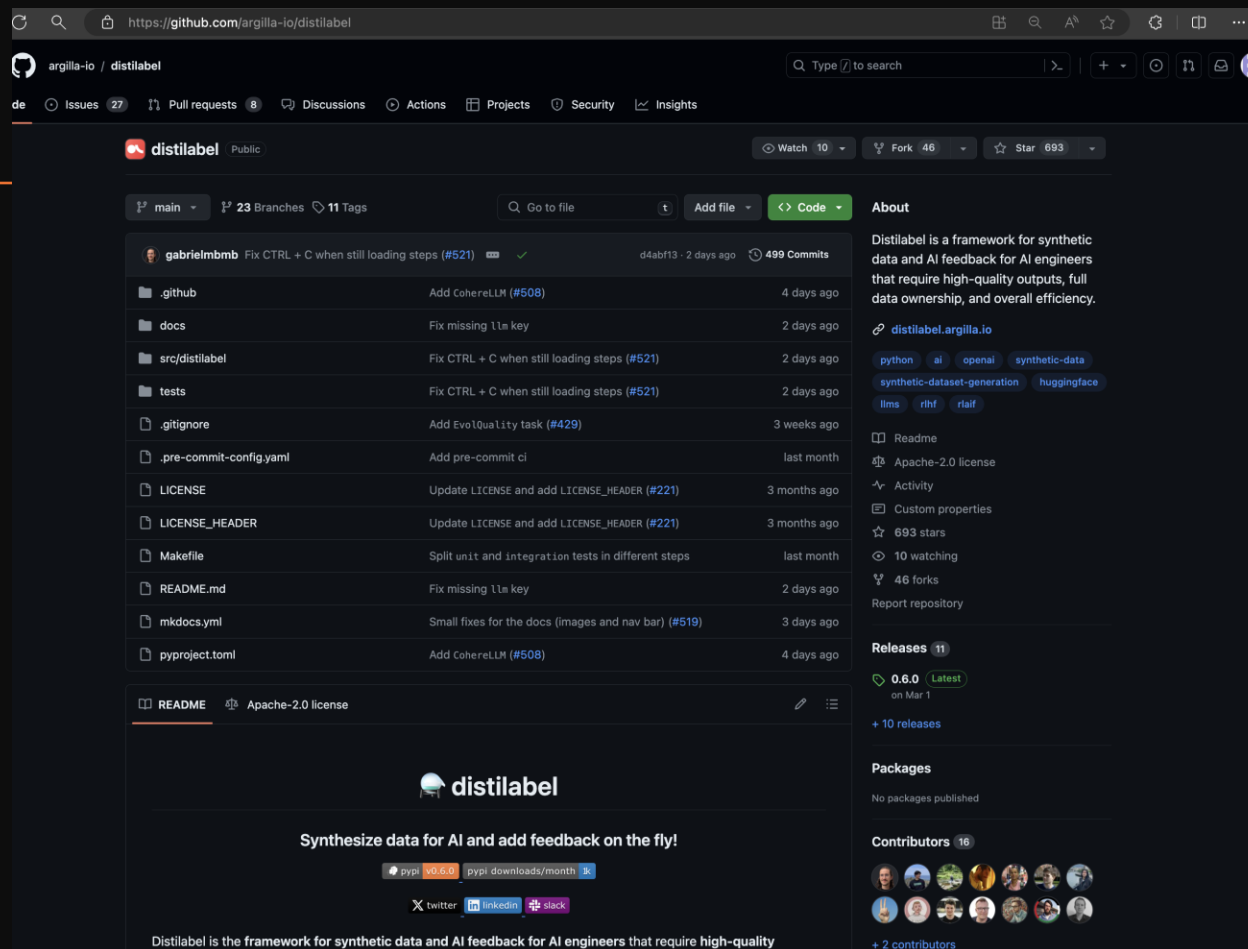


Complex Synthetic Datasets

Local AI Generation



Generate Synthetic Datasets Locally



Started In
2022

16

Contributors

693

stars on Github

8

Active PRs

Create a synthetic dataset seed locally on your own AI platform for aligning models to a **specific domain** [example](#)

Distilabel is a framework for synthetic data and AI feedback for AI engineers that require high-quality outputs, full data ownership, and overall efficiency.



Practical Use Cases

Practical Use Cases



1. Content Creation
2. Automation of Routine Tasks
3. Human-Computer Interface Personalization
4. Assisted Software Development
5. Design and Prototyping
6. Synthetic data generation



Thank You!
We Will Meet Again!