

# Accelerate AI agents with Multimodal RAG using Friendli Endpoints and Milvus

Zilliz Webinar  
Feb 2025  
Soomin Chun



## About me

- Software Engineer at FriendliAI, leading provider of generative AI infrastructure
- CS at MIT
- Previously at ed-tech start-up, Meta, quantitative finance

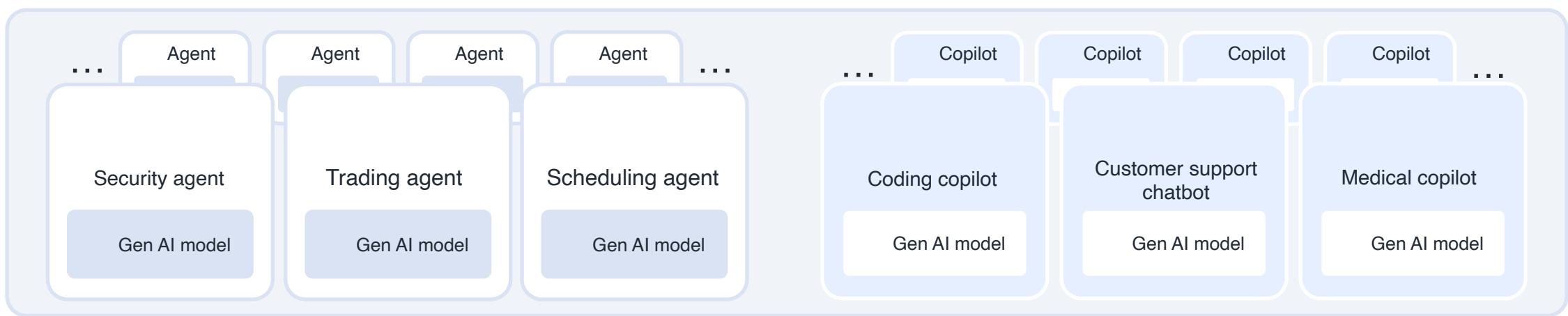


Our mission is to empower organizations to harness  
the power of generative AI with ease and cost-efficiency.

# Agenda

1. Background: LLMs, LLM inference serving, and AI agents
2. Challenges in building AI agents
3. Advantages of FriendliAI and Milvus
4. Live demo – Colab notebook walkthrough

# Increasing use of open-source generative AI models



Many companies are embracing (custom) open-source generative AI models.

High quality open-source gen AI models

(Meta Llama models get 350 million downloads (2024.8)<sup>1</sup>)

Cohere Command R+

Google Gemma

MistralAI Mixtral

Snowflake Arctic

Meta Llama

Microsoft Phi

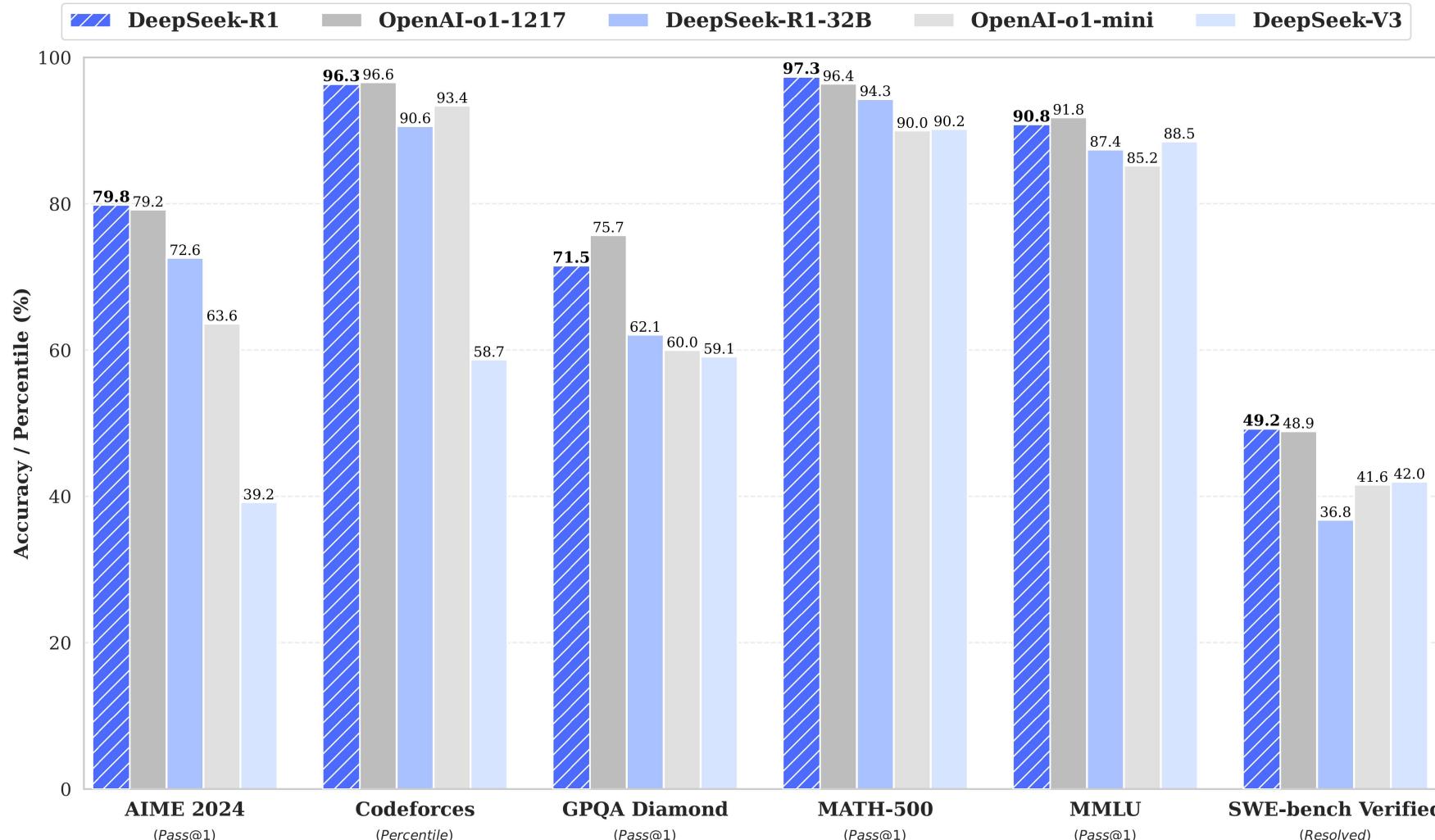
deepseek

Databricks DBRX

1. <https://www.computerworld.com/article/3499062/metas-llama-models-get-350-million-downloads.html>

# DeepSeek-R1

## Test-time compute



Ref. <https://github.com/deepseek-ai/DeepSeek-R1>

# What is a Large Language Model (LLM)?

Hugging Face Search models, datasets, users...

meta-llama/Llama-3.1-70B-Instruct like 781 Follow M

Text Generation Transformers Safetensors PyTorch 8 languages

Inference Endpoints arxiv:2204.05149 License: llama3.1

Model card Files and versions Community 43

A newer version of this model is available: meta-llama/Llama-3.3-70B-Instruct

Gated model You have been granted access to this model

### Model Information

The Meta Llama 3.1 collection of multilingual large language models (LLMs) is a collection of pretrained and instruction tuned generative models in 8B, 70B and 405B sizes (text in/text out). The Llama 3.1 instruction tuned text only models (8B, 70B, 405B) are optimized for multilingual dialogue use cases and outperform many of the available open source and closed chat models on common industry benchmarks.

Model developer: Meta

**Model Architecture:** Llama 3.1 is an auto-regressive language model that uses an optimized transformer architecture. The tuned versions use supervised fine-tuning (SFT) and reinforcement learning with human feedback (RLHF) to align with human preferences for helpfulness and safety.

Training Data	Params	Input modalities	Output modalities	Context length	GQA	Token count	Knowledge cutoff

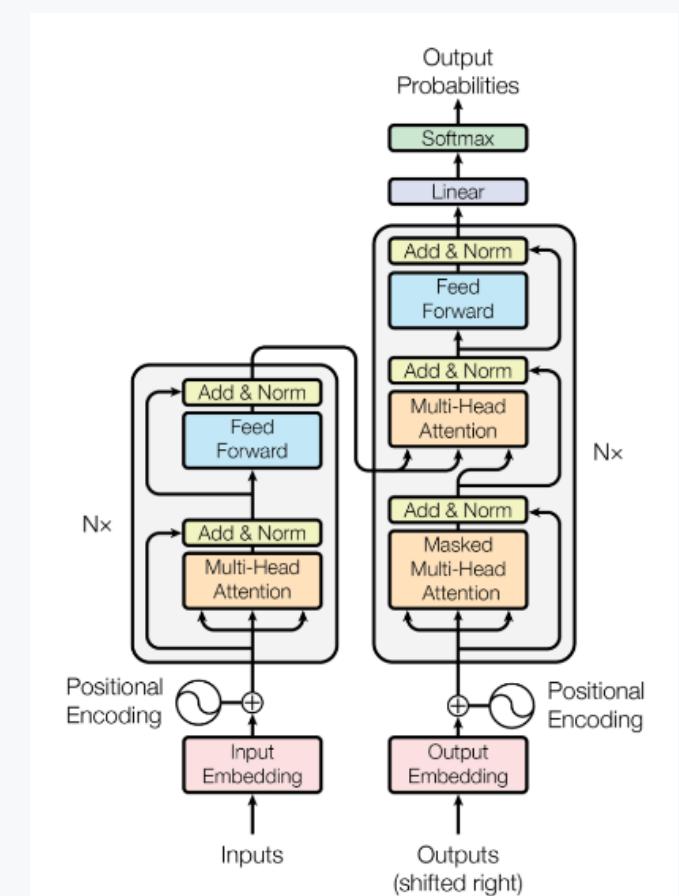
File

model.safetensors.index.json 59.6 kB

1/31 Download View all files

Tensors	Shape	Precision
model		
model.embed_tokens.weight	[128 256, 8 192]	BF16
model.layers.0		
model.layers.0.input_layernorm.weight	[8 192]	BF16
model.layers.0.mlp.down_proj.weight	[8 192, 28 672]	BF16
model.layers.0.mlp.gate_proj.weight	[28 672, 8 192]	BF16
model.layers.0.mlp.up_proj.weight	[28 672, 8 192]	BF16
model.layers.0.post_attention_layernorm.weight	[8 192]	BF16
model.layers.0.self_attn.k_proj.weight	[1 024, 8 192]	BF16
model.layers.0.self_attn.o_proj.weight	[8 192, 8 192]	BF16
model.layers.0.self_attn.q_proj.weight	[8 192, 8 192]	BF16
model.layers.0.self_attn.v_proj.weight	[1 024, 8 192]	BF16
model.layers.1		
model.layers.1.input_layernorm.weight	[8 192]	BF16
model.layers.1.mlp.down_proj.weight	[8 192, 28 672]	BF16
model.layers.1.mlp.gate_proj.weight	[28 672, 8 192]	BF16
model.layers.1.mlp.up_proj.weight	[28 672, 8 192]	BF16
model.layers.1.post_attention_layernorm.weight	[8 192]	BF16
model.layers.1.self_attn.k_proj.weight	[1 024, 8 192]	BF16
model.layers.1.self_attn.o_proj.weight	[8 192, 8 192]	BF16
model.layers.1.self_attn.q_proj.weight	[8 192, 8 192]	BF16
model.layers.1.self_attn.v_proj.weight	[1 024, 8 192]	BF16
model.layers.2		
model.layers.2.input_layernorm.weight	[8 192]	BF16
model.layers.2.mlp.down_proj.weight	[8 192, 28 672]	BF16
model.layers.2.mlp.gate_proj.weight	[28 672, 8 192]	BF16
model.layers.2.mlp.up_proj.weight	[28 672, 8 192]	BF16
model.layers.2.post_attention_layernorm.weight	[8 192]	BF16
model.layers.2.self_attn.k_proj.weight	[1 024, 8 192]	BF16
model.layers.2.self_attn.o_proj.weight	[8 192, 8 192]	BF16
model.layers.2.self_attn.q_proj.weight	[8 192, 8 192]	BF16
model.layers.2.self_attn.v_proj.weight	[1 024, 8 192]	BF16

Tensors	Shape	Precision
model		
model.embed_tokens.weight	[128 256, 8 192]	BF16
model.layers.0		
model.layers.0.input_layernorm.weight	[8 192]	BF16
model.layers.0.mlp.down_proj.weight	[8 192, 28 672]	BF16
model.layers.0.mlp.gate_proj.weight	[28 672, 8 192]	BF16
model.layers.0.mlp.up_proj.weight	[28 672, 8 192]	BF16
model.layers.0.post_attention_layernorm.weight	[8 192]	BF16
model.layers.0.self_attn.k_proj.weight	[1 024, 8 192]	BF16
model.layers.0.self_attn.o_proj.weight	[8 192, 8 192]	BF16
model.layers.0.self_attn.q_proj.weight	[8 192, 8 192]	BF16
model.layers.0.self_attn.v_proj.weight	[1 024, 8 192]	BF16
model.layers.1		
model.layers.1.input_layernorm.weight	[8 192]	BF16
model.layers.1.mlp.down_proj.weight	[8 192, 28 672]	BF16
model.layers.1.mlp.gate_proj.weight	[28 672, 8 192]	BF16
model.layers.1.mlp.up_proj.weight	[28 672, 8 192]	BF16
model.layers.1.post_attention_layernorm.weight	[8 192]	BF16
model.layers.1.self_attn.k_proj.weight	[1 024, 8 192]	BF16
model.layers.1.self_attn.o_proj.weight	[8 192, 8 192]	BF16
model.layers.1.self_attn.q_proj.weight	[8 192, 8 192]	BF16
model.layers.1.self_attn.v_proj.weight	[1 024, 8 192]	BF16
model.layers.2		
model.layers.2.input_layernorm.weight	[8 192]	BF16
model.layers.2.mlp.down_proj.weight	[8 192, 28 672]	BF16
model.layers.2.mlp.gate_proj.weight	[28 672, 8 192]	BF16
model.layers.2.mlp.up_proj.weight	[28 672, 8 192]	BF16
model.layers.2.post_attention_layernorm.weight	[8 192]	BF16
model.layers.2.self_attn.k_proj.weight	[1 024, 8 192]	BF16
model.layers.2.self_attn.o_proj.weight	[8 192, 8 192]	BF16
model.layers.2.self_attn.q_proj.weight	[8 192, 8 192]	BF16
model.layers.2.self_attn.v_proj.weight	[1 024, 8 192]	BF16



Ref. Attention is All you Need

# Evolution of generative AI

## Vision language models

Describe this image.



This image shows a blue llama with the words "Meta Llama 3.2 Vision".

## Image generation models

Generate an image of a purple cow in a digital art style.



Ref. Medium article: Llama 3.2 Vision: Multimodal AI with Image Reasoning

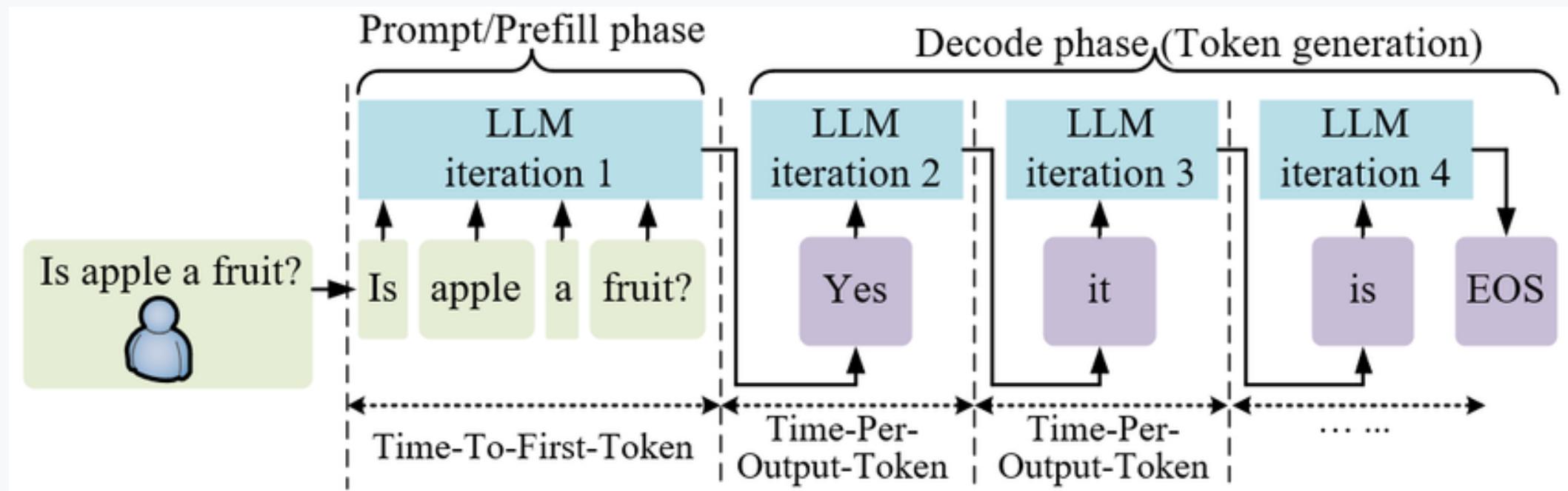
Confidential

Ref. Zapier blog: How to use Stable Diffusion

© FriendliAI Corp. All Rights Reserved.

# What is generative AI inference serving?

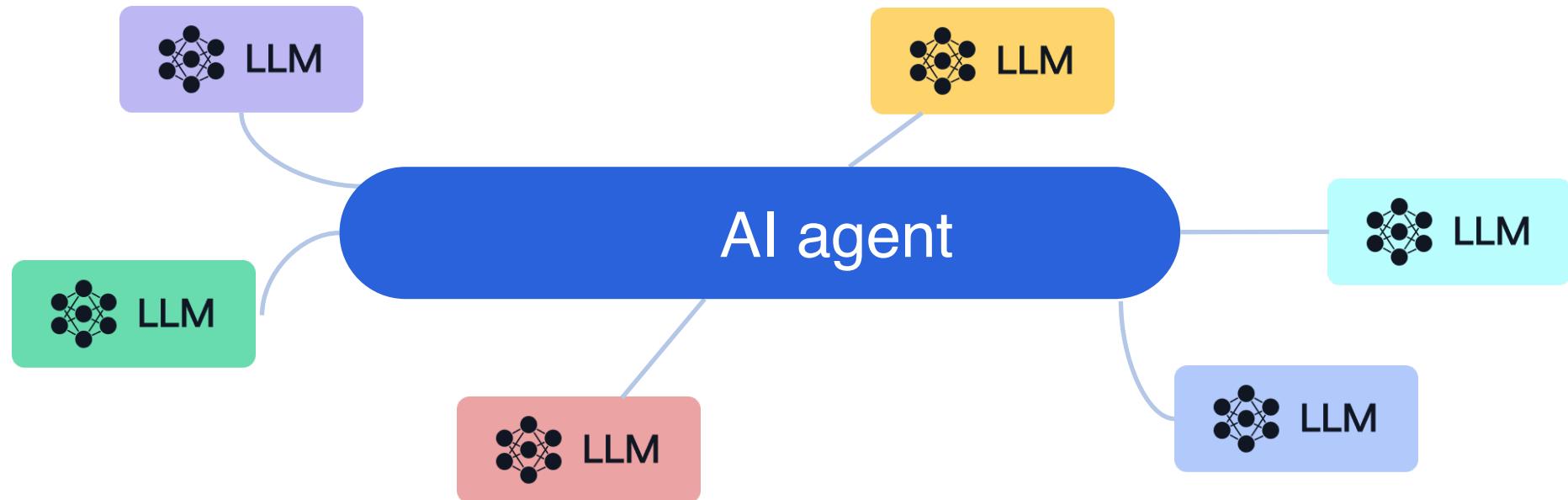
LLM inference



This process is very computationally intensive. For efficient inference, we need AI Hardware (GPUs), which can be very expensive.

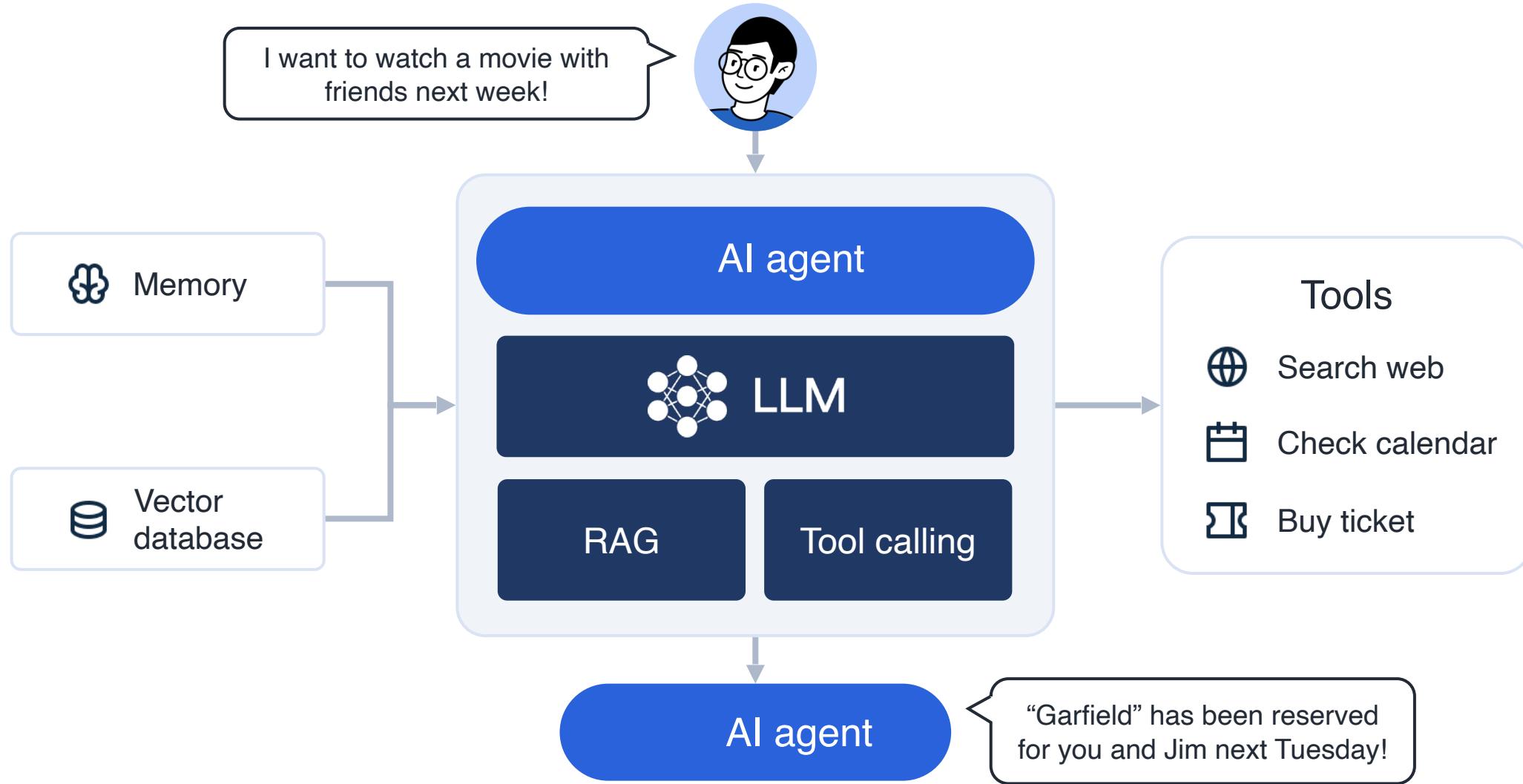
# What is an AI agent?

An AI agent is a software system that autonomously interacts with its environment to achieve goals, often powered by large language models (LLMs).



# Agentic AI

Many more LLM calls



# Challenges in building/running AI agents

## AI agent

### Goals

Responses should be:



High quality



Fast



Affordable

### Hurdles

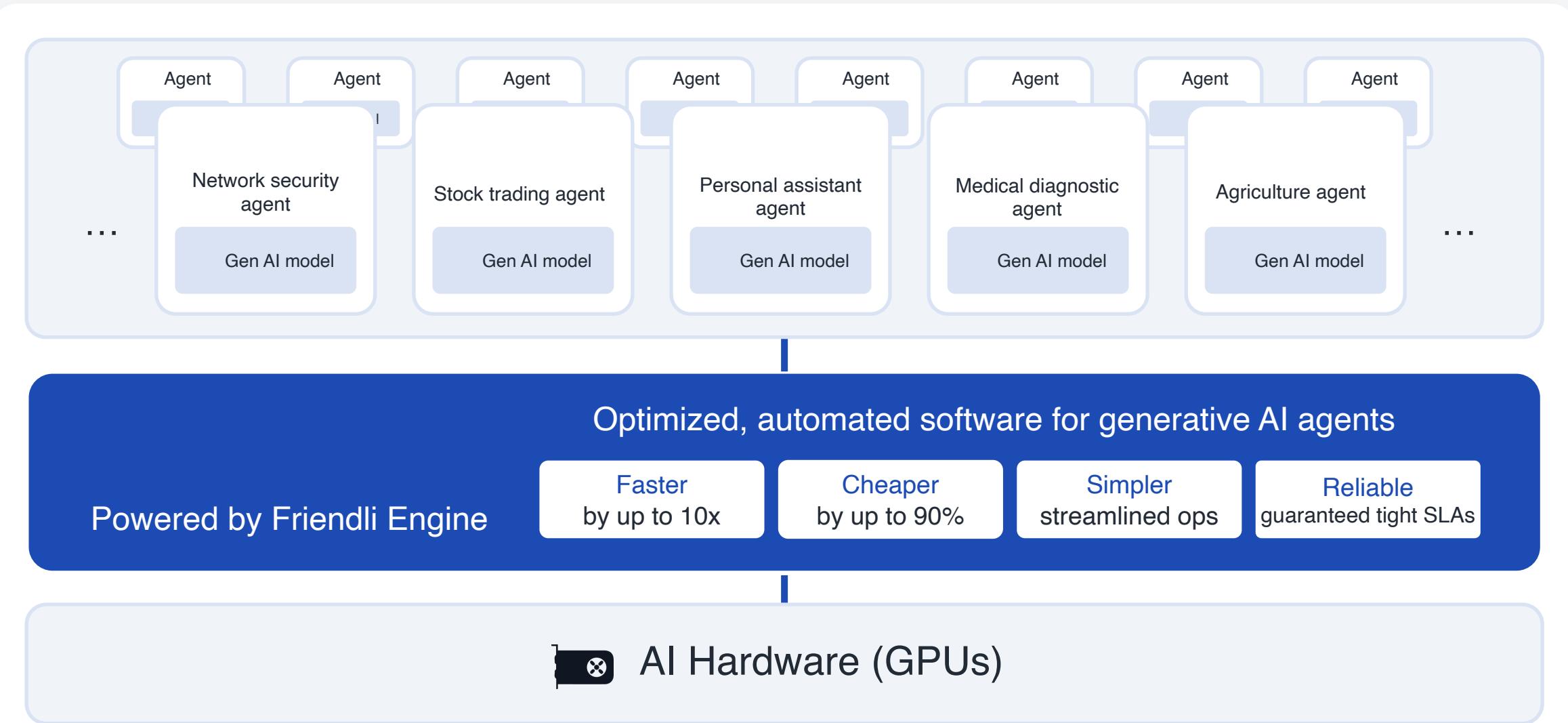
#### 1. Building

Complicated, with many parts

#### 2. Running

LLMs can be very costly

# FriendliAI: Infrastructure for running generative AI



The screenshot shows the Hugging Face Model Card for the Llama-3.3-70B-Instruct model. At the top, there's a navigation bar with links for Models, Datasets, Spaces, Posts, Docs, Enterprise, and Pricing. Below the navigation is a search bar and a user profile icon.

The main title is "meta-llama/Llama-3.3-70B-Instruct". Below the title, there are several categories: Text Generation, Transformers, Safetensors, PyTorch, 8 languages, llama, facebook, meta, llama-3, conversational, and text-generation-inference. There are also links for Inference Endpoints, arxiv:2204.05149, and License: llama3.3.

The "Model card" tab is selected. On the right side of the card, there are buttons for Edit model card, Train, Deploy, and Use this model. A chart shows "Downloads last month" at 610,427, with a cursor pointing at it. Below the chart, there are sections for Safetensors (Model size: 70.6B params, Tensor type: BF16), Inference Providers (Text Generation, SambaNova), and Examples.

**Gated model**: You have been granted access to this model.

**Model Information**

The Meta Llama 3.3 multilingual large language model (LLM) is an instruction tuned generative model in 70B (text in/text out). The Llama 3.3 instruction tuned text only model is optimized for multilingual dialogue use cases and outperforms many of the available open source and closed chat models on common industry benchmarks.

**Model developer:** Meta

**Model Architecture:** Llama 3.3 is an auto-regressive language model that uses an optimized transformer architecture. The tuned versions use supervised fine-tuning (SFT) and reinforcement learning with human feedback (RLHF) to align with human preferences for helpfulness and safety.

Training Data	Params	Input modalities	Output modalities	Context length	GQA	Token count	Knowledge cutoff
---------------	--------	------------------	-------------------	----------------	-----	-------------	------------------

Input a message to start chatting with meta-llama/Llama-3.3-70B-Instruct.  
Your sentence here...

# Why use FriendliAI for inference?

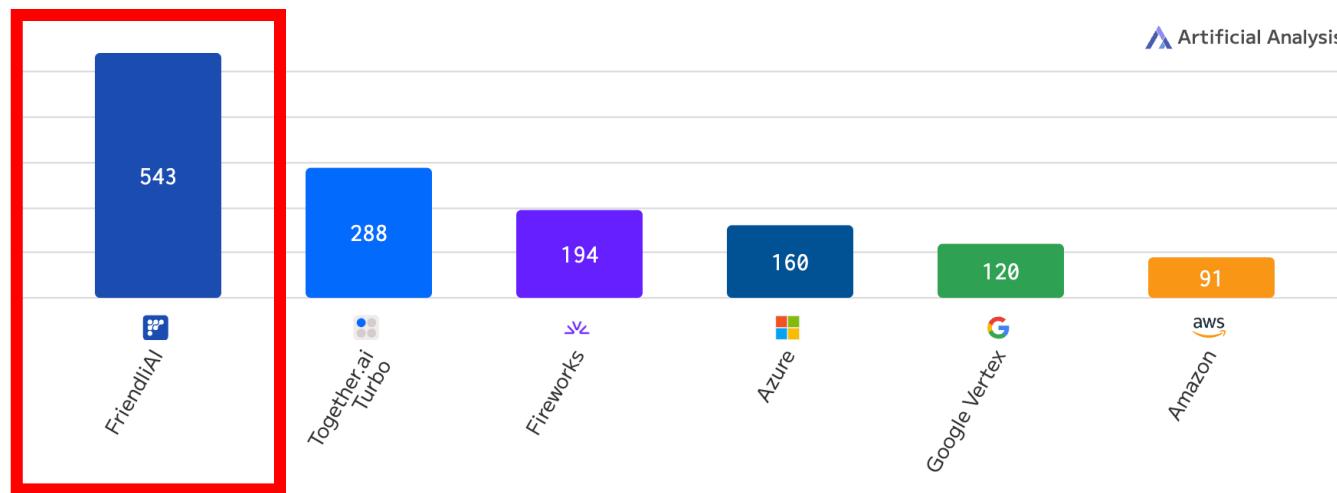
## The world's No.1 GPU-based API provider

Llama 3.1 8B  
A single request

(Artificial Analysis <https://artificialanalysis.ai/>)

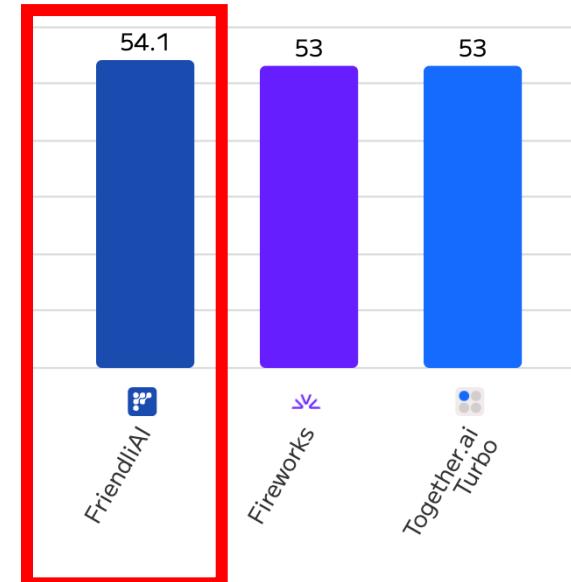
### Output Speed

Output Tokens per Second; Higher is better



High speed

### Artificial Analysis Quality Index



High quality

# High throughput of Friendli: reduction in required GPUs

Model

Llama 3.1 8B

GPU

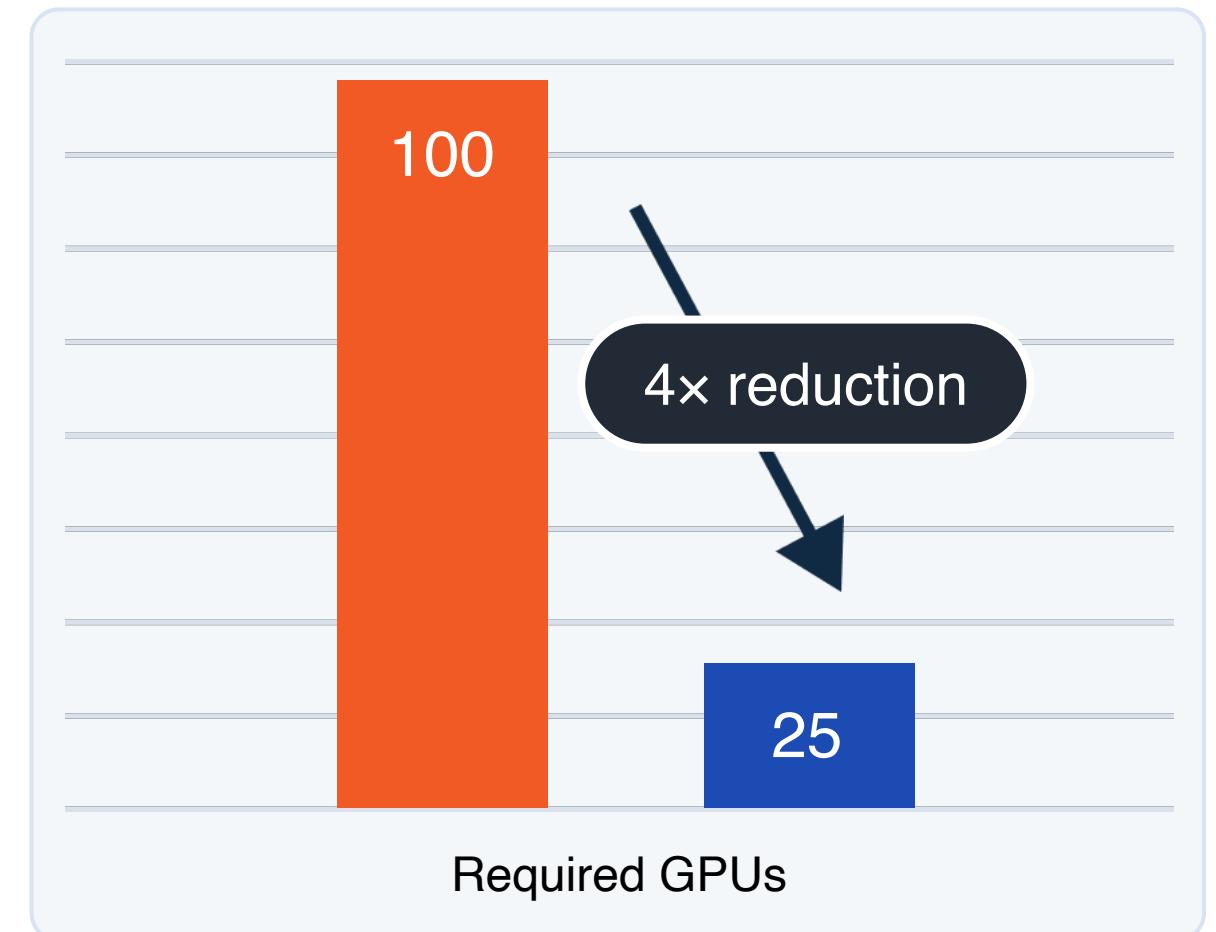
NVIDIA H100 80GB GPU

Concurrent Users (\* 1 rpm per user)

75,000

vLLM

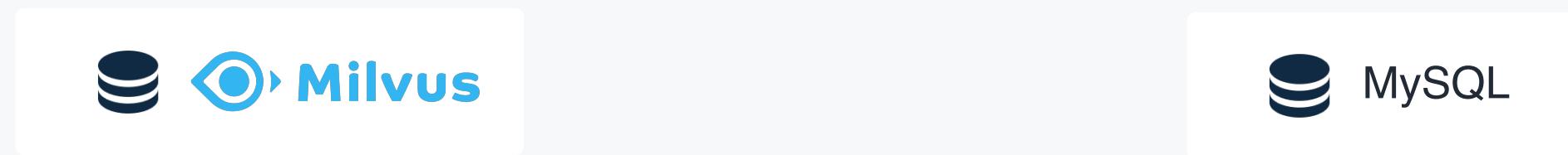
FriendliAI



# Why use Milvus?

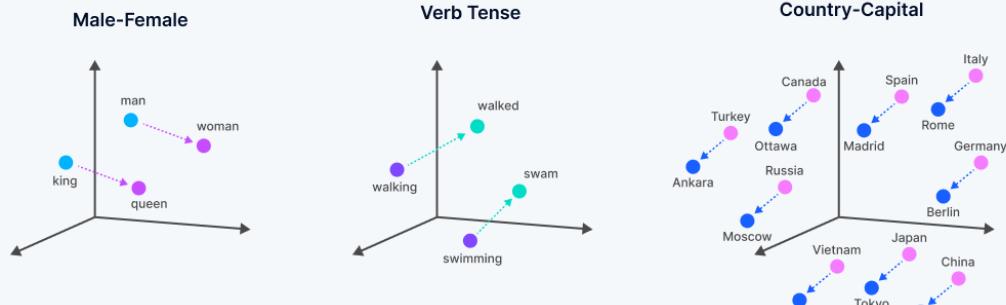
Vector database for generative AI

Query: "What is the capital of France?"



Vector Database

Vector similarity search



Embedding arithmetic in action

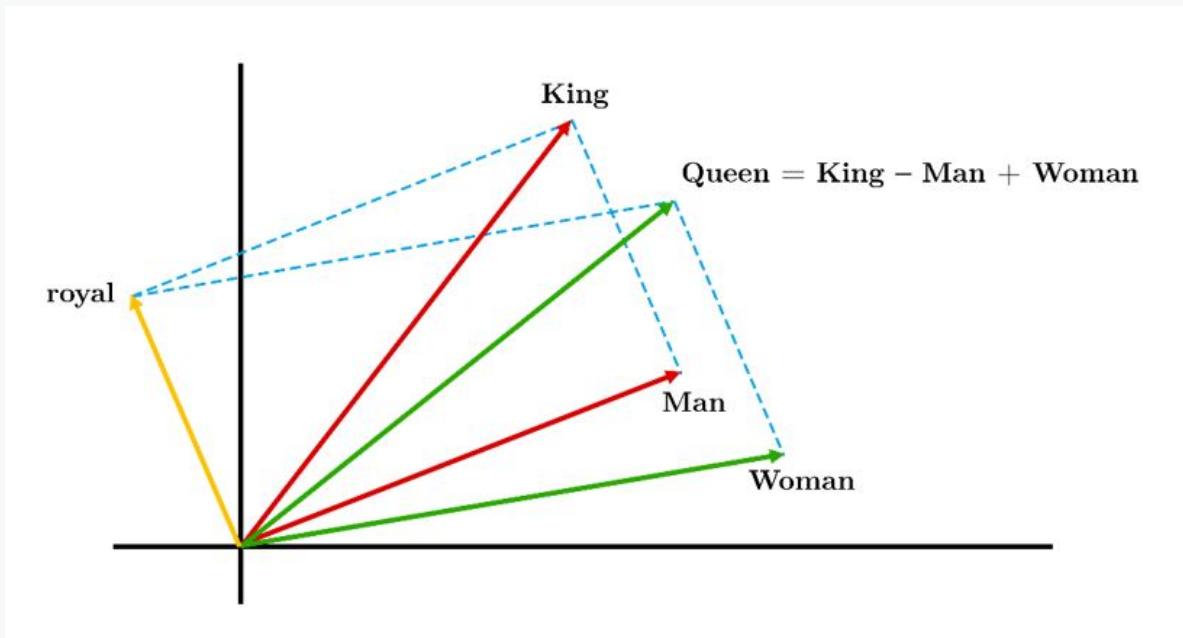


Looks for exact matches of “capital” and  
“France” in a structured table.

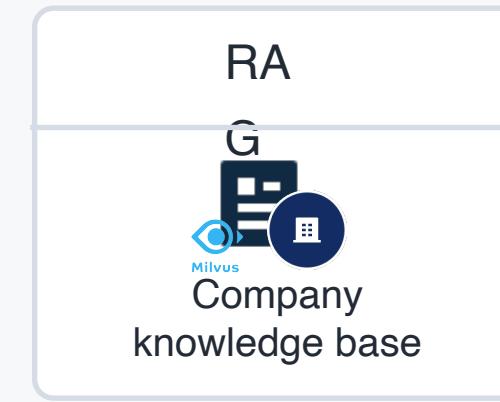
```
mysql> SELECT * FROM students;
+----+-----+-----+
| id | name | class |
+----+-----+-----+
| 1  | Stephen | 6    |
| 2  | Bob     | 7    |
| 3  | Steven   | 8    |
| 4  | Donald   | 6    |
| 5  | Jenifer  | 9    |
| 6  | Peter    | 9    |
| 7  | Alexandar | 7    |
+----+-----+-----+
7 rows in set (0.00 sec)
```

# Quick overview of Retrieval Augmented Generation (RAG)

Vector representations capture semantic similarity



<https://lamarr-institute.org/blog/dedicom-matrix-factorization/>



Query  
+  
Retrieved information

# Demo

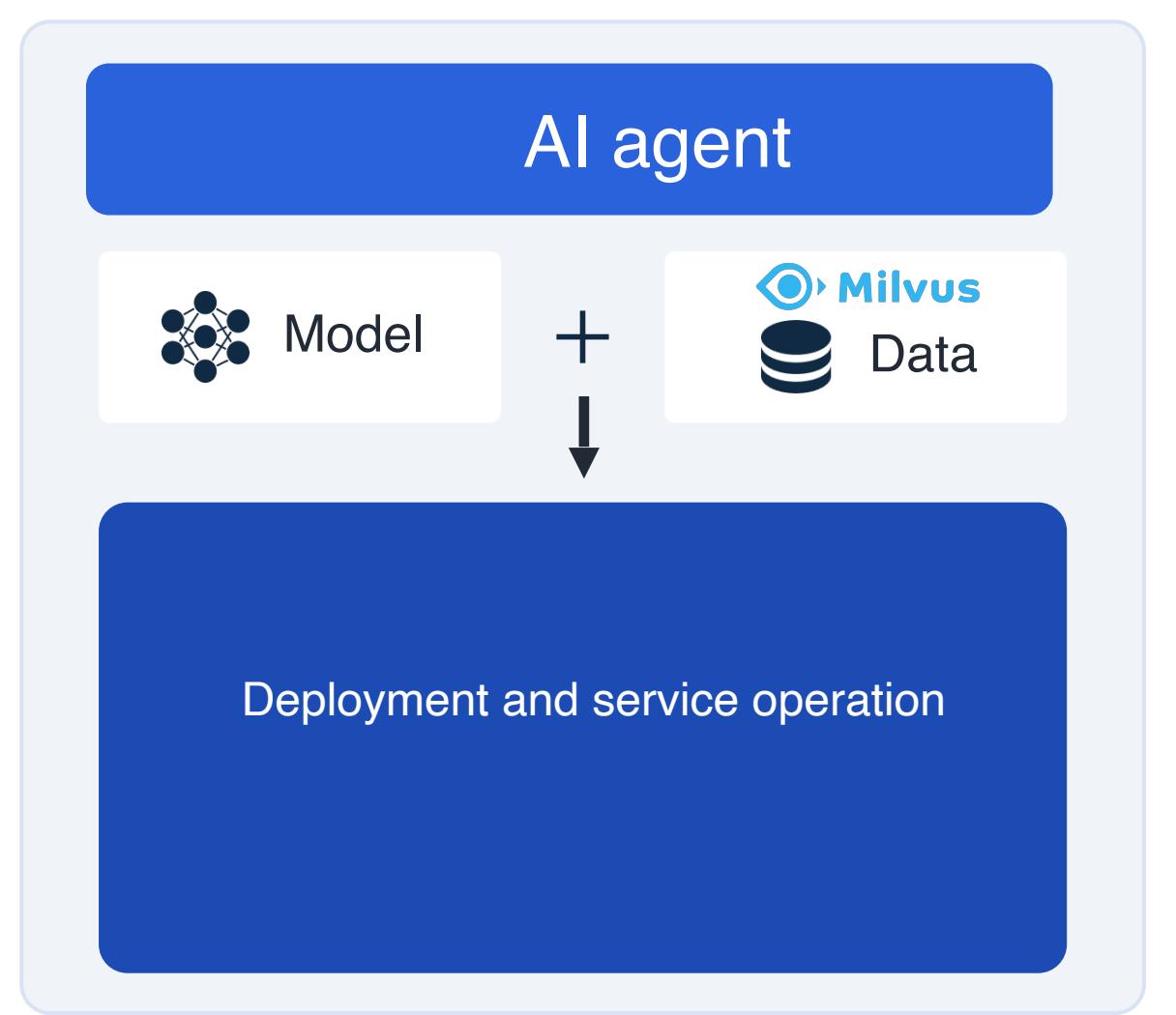
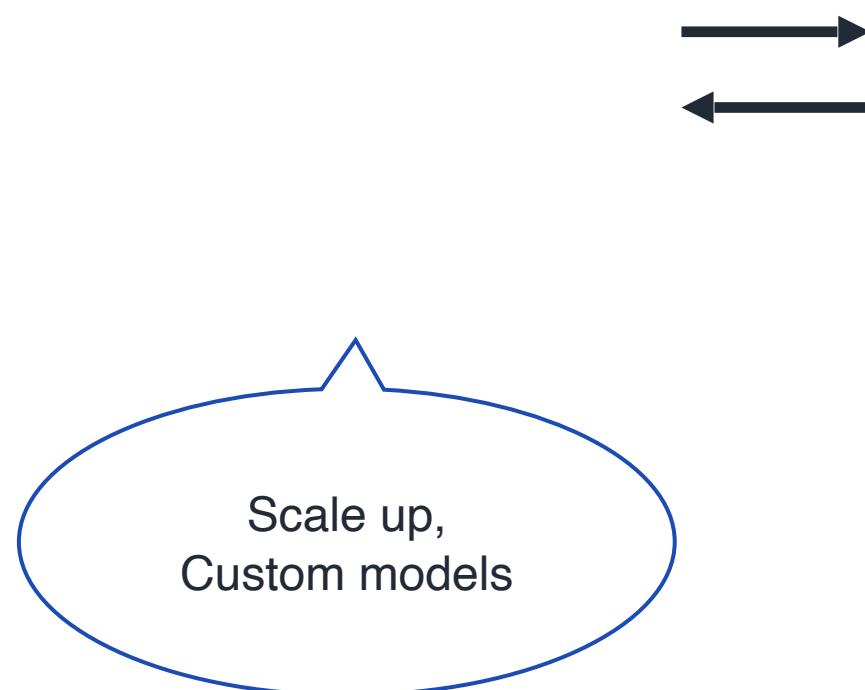
- Milvus & Friendli Serverless Endpoints
- Tool Calling (pre-defined search tool & custom tool)
- Retrieval-Augmented Generation

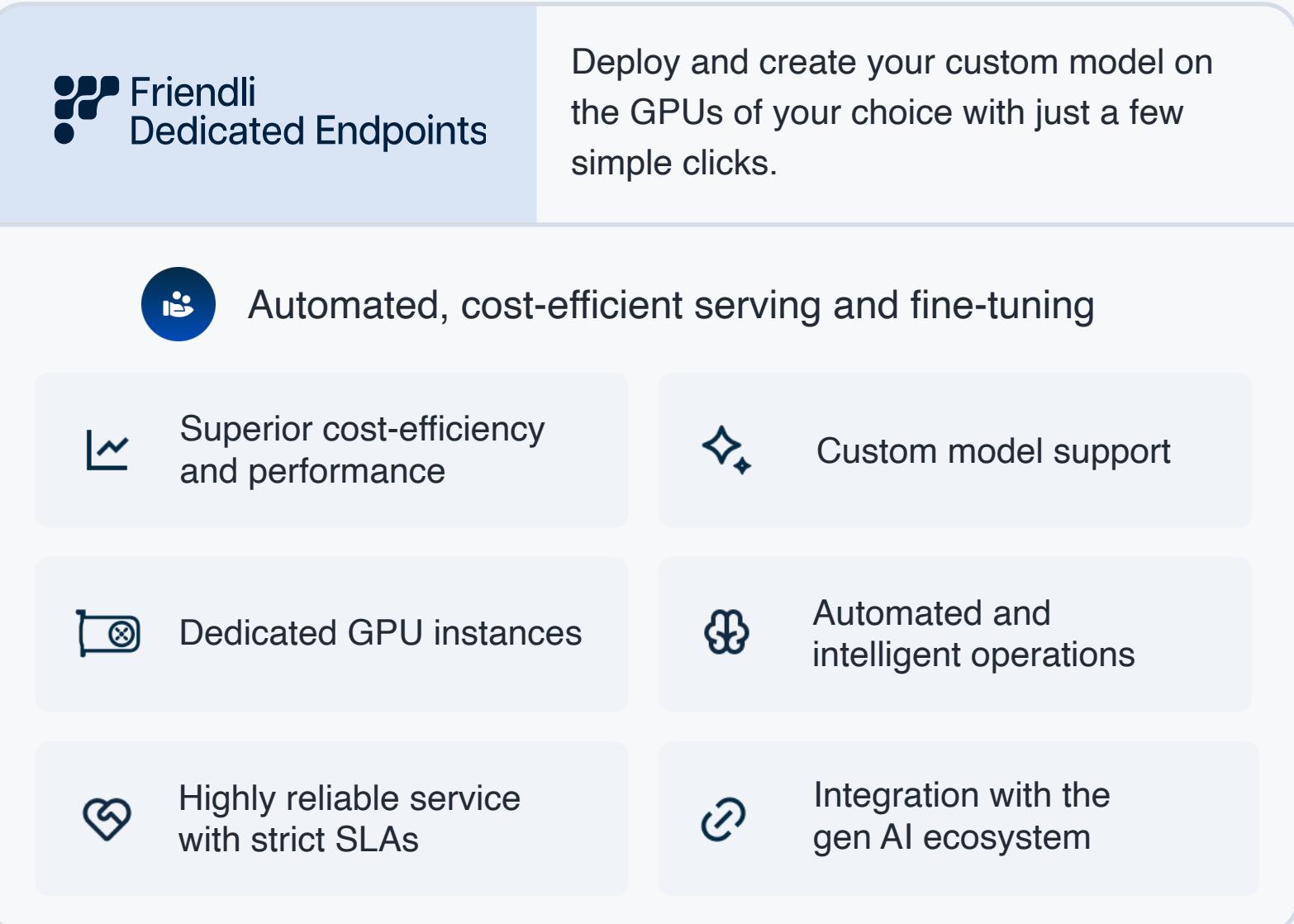
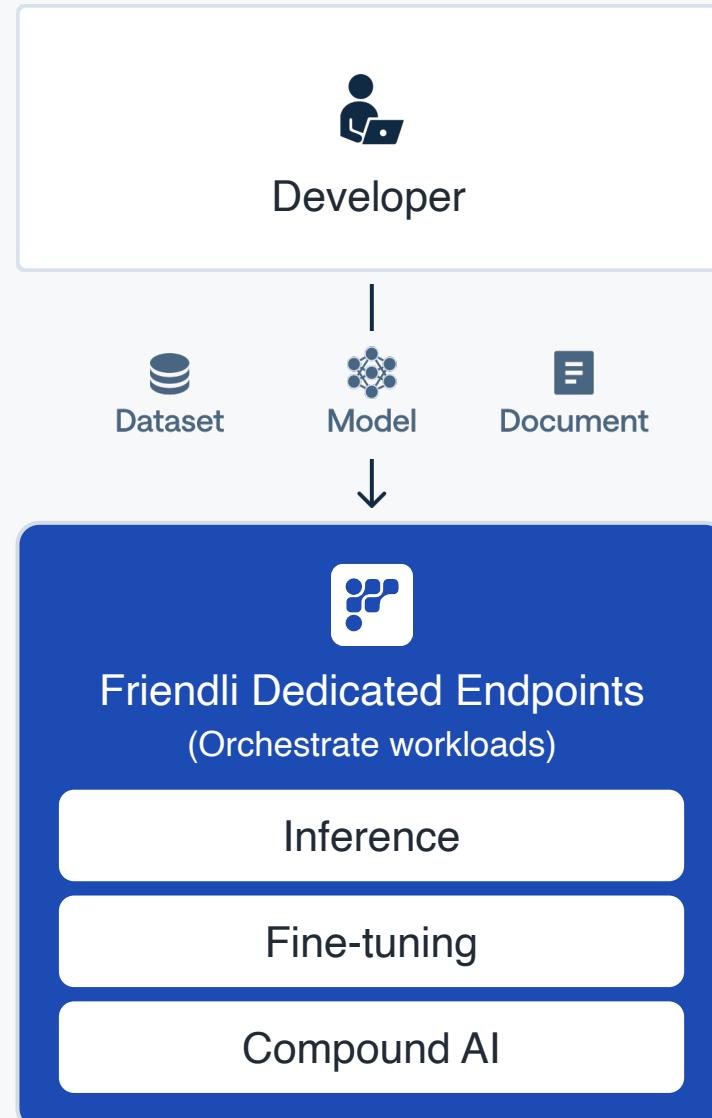


← [Google Colab Link](#)

# Operating a large-scale AI agent service **with** **Friendli**

AI agent service operator





# 1. Fine-tuning

Create your custom model by fine-tuning

Fine-tuning   Documents   Project settings   \$0.0 remaining

[← Back to fine-tuning](#)

## Create a new job

Job name: llama-3-8b-it

Hugging Face integration: Hugging Face integrated (checked)

Account: Soeun Token new

Model: Hugging Face model

Search bar: meta-llama/Meta-Llama-3-8B-Instruct

Gated model: main

You have been granted access to this model.

Dataset: Hugging Face dataset

Search bar: FriendliAI/gsm8k

main

Weights & Biases (W&B):

W&B API key (optional): Your W&B API key

You can find your W&B API key at [this link](#)

Hyperparameters:

learning\_rate float (required) ⓘ  
0.0001

batch\_size int (required) ⓘ  
16

# 2. Inference

Deploy your fine-tuned model

Endpoints   Fine-tuning   Documents   Project settings   \$0.0 remaining

[← Back to endpoints](#)

## Create a new endpoint

Endpoint name

Endpoint name

Model

We only support LLM models. Learn more about [Supported models](#).

Single   Multi-LoRA

Multi-LoRA serving  
Friendli Dedicated Endpoints support multi-LoRA serving, allowing you to serve pre-trained LLMs with multiple fine-tuned adapters.

Base model: meta-llama/Meta-Llama-3-8B-Instruct   [Edit](#)

Adapter model: meta-llama-3-8b-it   [Edit](#)

Instance configuration

NVIDIA A100 80GB  
1x GPU \$3.8/h

NVIDIA L4  
1x GPU \$1/h

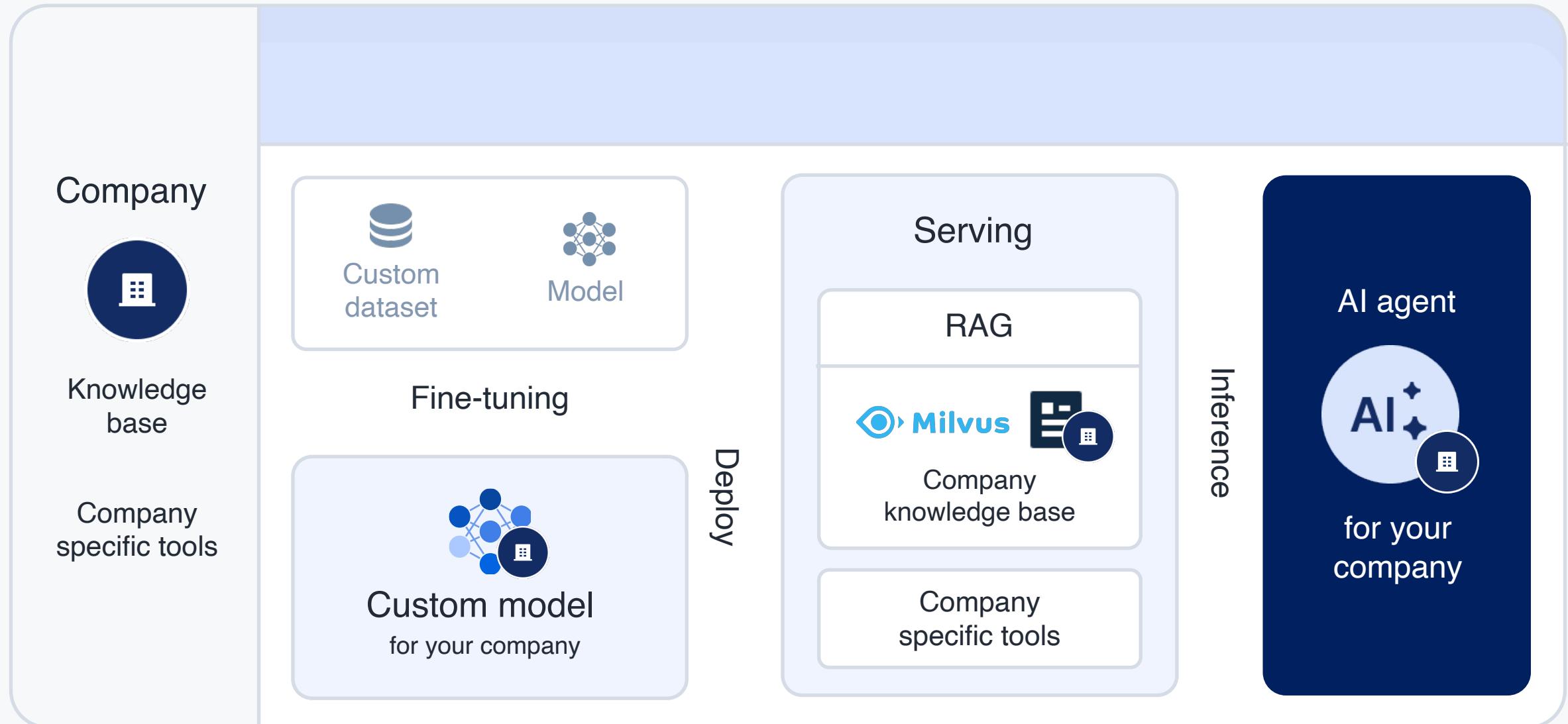
Configuration   [Edit configuration](#)

Autoscaling  
Autoscaling handles highly variable traffic while minimizing spend on idle compute resources. [Enabled](#)

Number of replicas [i](#)   Cooldown period [i](#)



# FriendliAI & Milvus: Build AI agents





Google  
Colab



Let's start building!



LinkedIn



friendli.ai



Friendli Suite

<https://suite.friendli.ai>