Integrating AI into Web Applications With Paperspace by DigitalOcean

James Skelton
Technical Evangelist
Paperspace

DigitalOcean

Paperspace provides GPU power for AI model training and deployment at scale

Develop

Launch a notebook to build a proof of concept

Train

Train or fine-tune AI models with machines

Deploy

Convert models into scalable API endpoints

Built for AI Developers
key technical differentiation

GPU Acceleration

Access powerful GPU instances, such as NVIDIA H100s, Tesla, and Quadro GPUs, which are specifically designed for AI and machine learning workloads.

Pre-configured Deep Learning Containers

Run pre-configured Docker containers with popular deep learning frameworks like TensorFlow, PyTorch, and Keras.

Powerful Machine Learning Platform: Gradient

Gradient is Paperspace's machine learning platform, which provides a suite of tools and services to streamline the development, training, and deployment of AI models.

Collaboration and Sharing

Paperspace provides collaboration and sharing capabilities, allowing AI developers to work together on projects more effectively. Developers can easily share and collaborate on experiments, models, and data with their teammates, accelerating the development and iteration process.

How Paperspace empowers AI developers

Simple and powerful

Take your AI models from concept to production quickly and easily with intuitive tools and access to blazing fast GPUs and IPUs.

Predictable Costs

With per-second billing options for more powerful GPUs, you can save up to 70% on compute costs.

Built for teams

Collaborate with teams so you can share progress in real time and easily manage server infrastructure without restrictions.

Secure and Reliable

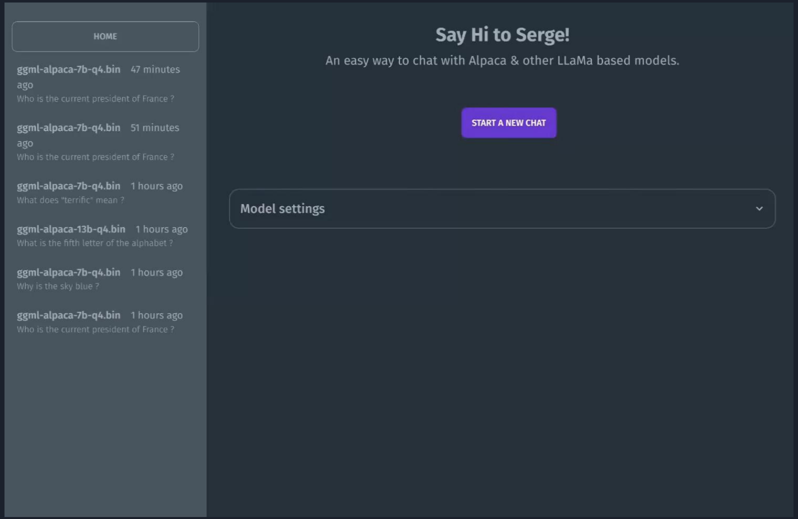
Our ultra reliable high-performance cloud compute is monitored 24/7 so you can focus on scaling your AI model training to an entire cluster powered by the latest NVIDIA GPUs.

Overview

- The Serge application
  - What does it do?
  - How does it use LangChain?
  - Additional useful information
- Developing with Language Models on Paperspace
- The Paperspace toolset
- Paperspace GPU offerings
- How to use Serge with Paperspace
  - Setting up Paperspace
  - Creating a Deployment through the UI
  - Interacting with your deployment through the API Endpoint

What is Serge?

- Per the Github repo: “Serge is a chat interface crafted with llama.cpp for running GGUF models. No API keys, entirely self-hosted!”
- In practice, it is an application for serving LLMs with a SvelteKit frontend, Redis chat history & storage management, and a FastAPI + LangChain API to wrap calls to llama.cpp using Python bindings



ggml-alpaca-7b-q4.bin 47 minutes ago
Who is the current president of France ?

ggml-alpaca-7b-q4.bin 51 minutes ago
Who is the current president of France ?

ggml-alpaca-7b-q4.bin 1 hours ago
What does "terrific" mean ?

ggml-alpaca-13b-q4.bin 1 hours ago
What is the fifth letter of the alphabet ?

ggml-alpaca-7b-q4.bin 1 hours ago
Why is the sky blue ?

ggml-alpaca-7b-q4.bin 1 hours ago
Who is the current president of France ?

Say Hi to Serge!

An easy way to chat with Alpaca & other LLaMa based models.

How does Serge use LangChain?

- Looking deeper into the application files (namely at chat.py and stream.py), we can see that the `langchain` package is used specifically to enable the RedisChatMessageHistory for memory management of the chat & several schema related functions like AIMessage, HumanMessage, and SystemMessage, that help organize our chat history gradient

www.gradient.run

Developing with Language Models on Paperspace

The Paperspace toolset

-

Paperspace GPU offerings

How to deploy Serge with Paperspace

-

Interacting with your deployment through the API Endpoint

Demo

- Application demo
  - Github
- Deployment spec: paste these values into gradient deployments to test the application.
  - image: serge-chat/serge:latest
  - port: 8000
  - resources:
  - replicas: 1
  - instanceType: A100-80/g <- I recommend this machine. See list of GPUs for options.