National Institute of Diabetes and Digestive and Kidney Diseases

What's Next? Machine Learning Applications for your AI-Ready Dataset to gain AI insights

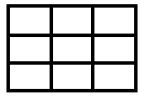
Anna Lu, Booz Allen Hamilton

NIDDK-CR Office Hours: January 19, 2024

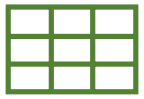
Reminders

- The deadline to submit your Challenge solution is on **Monday, January 22, 2024, at 11:59 PM (ET).**
  - Late submissions will not be accepted after this date. We encourage everyone to submit in advance of this deadline to ensure we can provide any technical support as needed during submission.
- Instructions on how to submit solutions are posted on Challenge.gov under the **How to Enter** tab
- Challenge Solution Submission Form was also recently updated. Please download and complete the latest V2 of this form from the **Resources** tab on Challenge.gov
- **Questions?** Check out the FAQs tab or contact niddk-crsupport@niddk.nih.gov

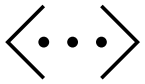
Data Centric Challenge – Submission Requirements



1. **Generate a "Raw" dataset by merging the study data files into a single dataset file.** This "Raw" dataset should be represented as a single rectangular file (i.e., tabular, spreadsheet, or matrix) in .csv file format **within the working directory of your Workspace**. You may use the R function write.csv(dataset-name, "file-name.csv") to achieve this



2. **Generate a single "AI-ready" dataset by enhancing the single raw dataset for AI-readiness.** This AI-ready dataset must be represented as a single rectangular file (i.e., tabular, spreadsheet, or matrix) in .csv file format **within the working directory of your Workspace**. You may use the R function write.csv(dataset-name, "file-name.csv") to achieve this.



3. **Submit the code script (i.e., .ipynb notebook file)** used to generate the "Raw" and "AI-Ready" files to your **Team's private GitHub repository.**

4. **Generate a human-readable data dictionary (i.e., codebook) documenting your AI-ready dataset**, preferably in Excel (.xlsx format). You may **either submit the code script to your Team's private GitHub repository** used to generate the data dictionary (recommended**) or provide this file within the working directory of your Workspace**. Both are acceptable.

5. **Complete the Challenge Solution Submission Form** describing the AI-ready dataset and methods for preparing the AI-ready dataset and **submit this form to Challenge.gov as an attachment**. Download and complete V2 from the Resources tab.

Please follow the **Submission Instructions** posted to Challenge.gov for Phase 2: Data Enhancement

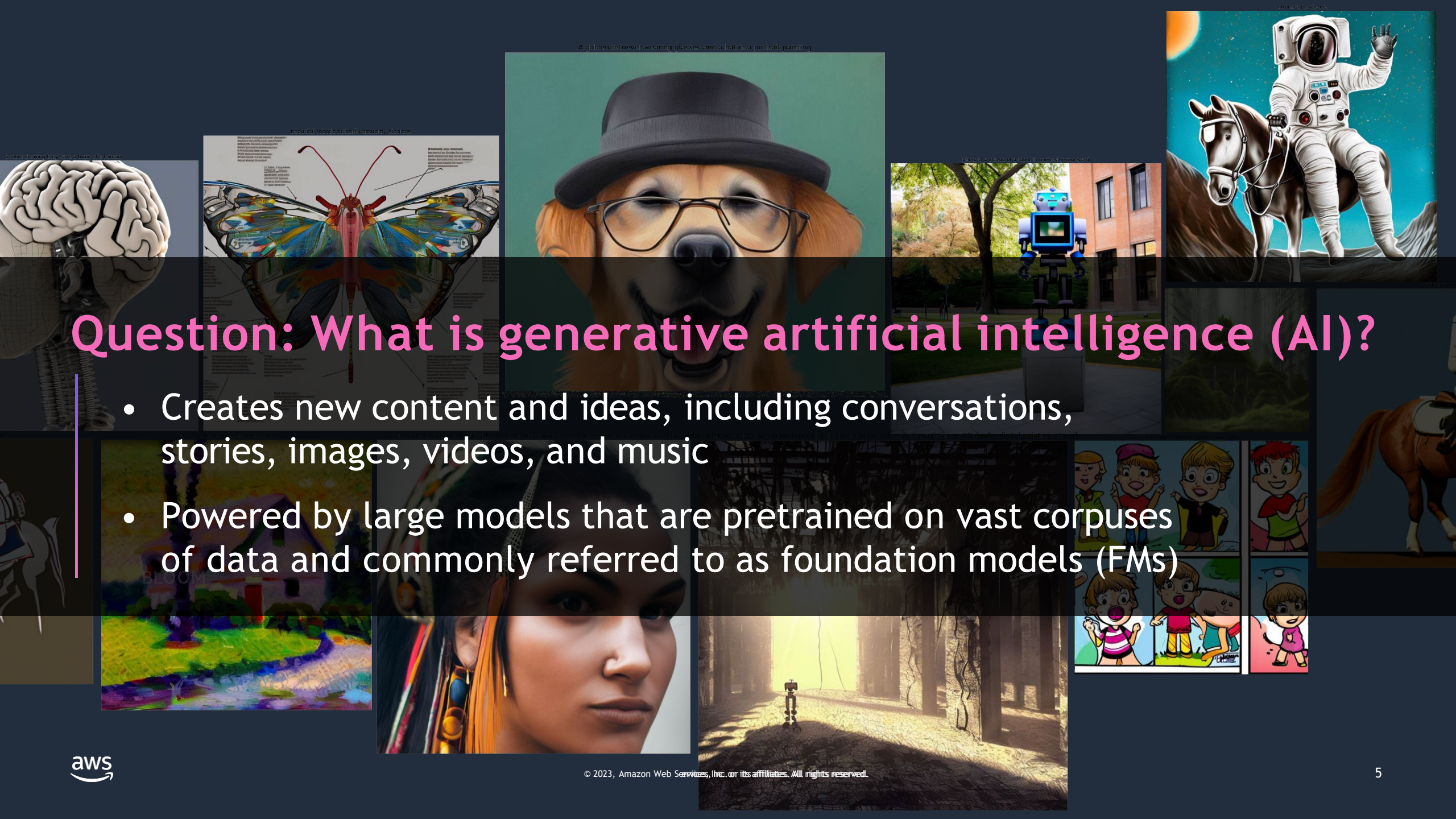
Frequently Asked Questions are also posted to Challenge.gov under the **FAQs** tab

Speaker Introduction

Anna Lu is a senior lead engineer on NIH projects: Rapid Acceleration of Diagnostics RADx DataHub, SeroHub for COVID-19 Seroprevalence, and Clinical Trials Reporting Program (CTRP), and Precision Medicine MATCH trials at National Cancer Institute (NCI).

She specializes in artificial intelligence on AWS environments implementing DevSecOps and data engineering best practices in designing analytics for biomedical researchers to leverage FAIR clinical and genomics datasets.

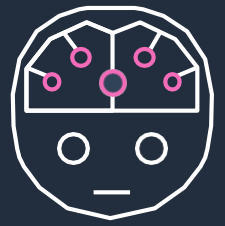
Anna has a B.S. degree in Biomedical Engineering from Drexel University. She mentors girls in STEM and teaches literacy.



Question: What is generative artificial intelligence (AI)?

- Creates new content and ideas, including conversations, stories, images, videos, and music
- Powered by large models that are pretrained on vast corpuses of data and commonly referred to as foundation models (FMs)

Where does generative AI fit?



Artificial intelligence (AI)

Any technique that allows computers to mimic human intelligence using logic, if-then statements, and machine learning

Machine learning (ML)

A subset of AI that uses machines to search for patterns in data to build logic models automatically

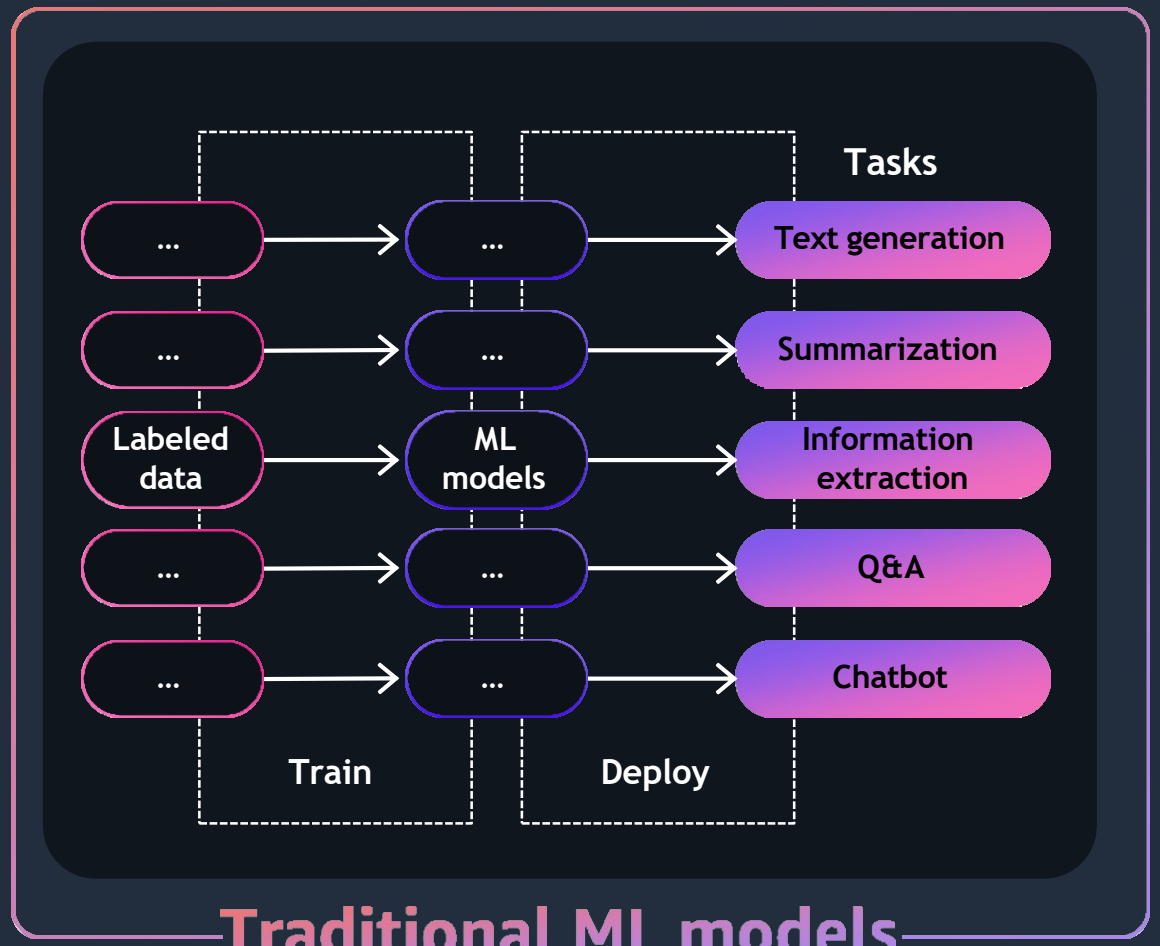
Deep learning (DL)

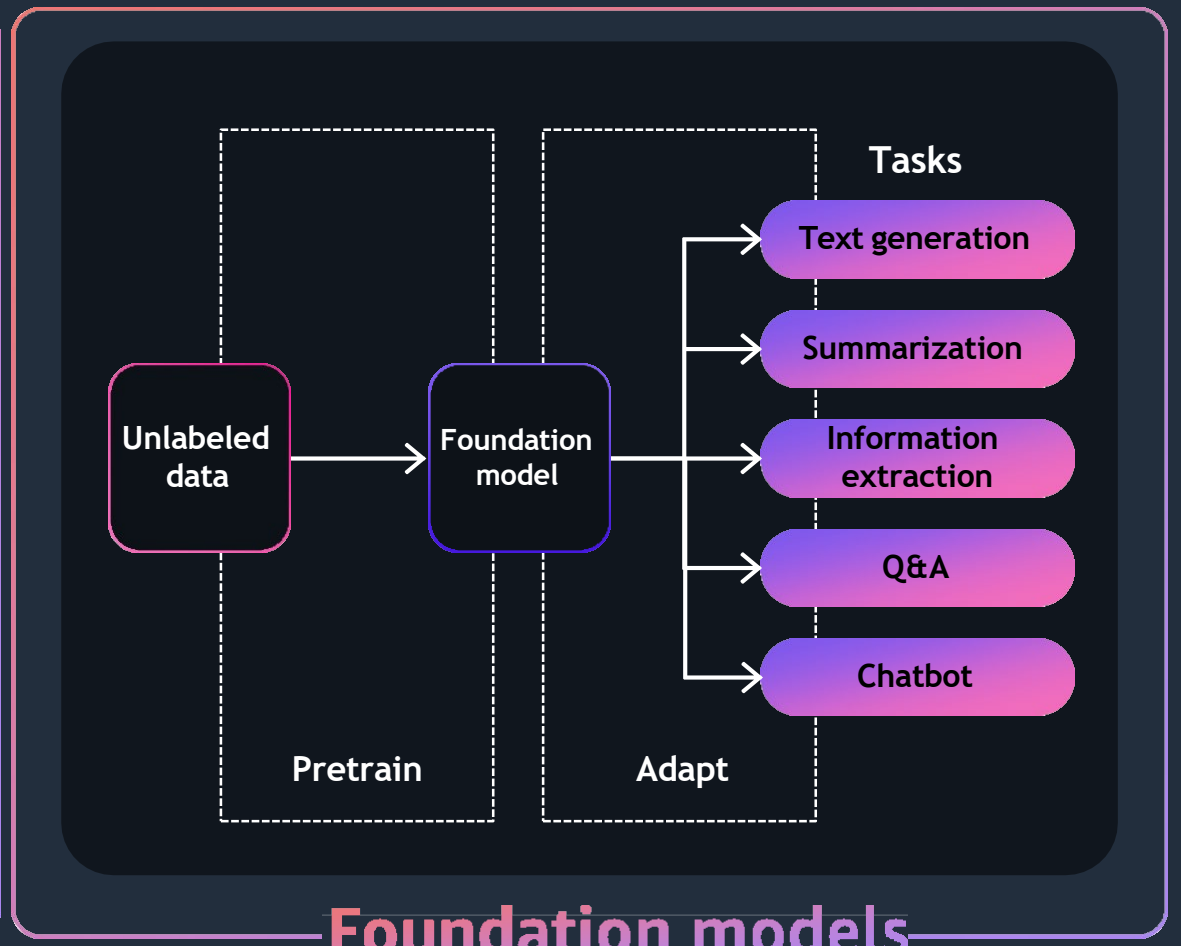
A subset of ML composed of deeply multi-layered neural networks that perform tasks like speech and image recognition

Generative AI

Powered by large models that are pretrained on vast corpora of data and commonly referred to as foundation models (FMs)

Why foundation models?





Types of foundation models

Input	Foundation model	Output
“Summarize the articles on impact of walking on heart health”	**Text-to-text** Generate text from simple natural-language prompts for various applications	“Ten thousand steps per day is optimum for maintaining a healthy heart”
“hand soap”	**Text-to-embeddings** Generate numerical representation of text for applications like search and finding similarities between documents	Numerical representation of “Hand soap refills Hand soap dispenser Hand soap antibacterial”
“a photo of an astronaut riding a horse on Mars”	**Multimodal** Generate and edit images from natural-language prompts	

Working with Foundation Models (FM)

In general, there are two ways you can work with an FM:

Directly access the knowledge encoded in the model using “Prompts”

“Fine-Tune” the FM for your own domain or use case

Prompt Engineering

The process of tailoring the prompt to extract accurate, consistent and fair outputs from the model is referred to as **“Prompt engineering”**

Prompt engineering is a **rapidly** emerging field.

Prompt engineering is often indicated by such terminology as **zero-, one-, few-, or many-shot** learning — all of which fall under the paradigm of in-context learning.

Fine-Tuning

Fine-tuning a FM is the process of adjusting and adapting the model to perform specific tasks or to cater to a particular domain more effectively.

This usually involves training the model further on a smaller, domain specific dataset that is more relevant.

The fine-tuned model is then accessed using Prompt Engineering

Common use cases

Text generation

Q&A

Text summarization

Text extraction

Paraphrase rephrase

Search

Code generation

Image generation

Image classification

Audio generation

Video generation

Generative AI application examples

Communications
Chatbot, question answering, search

Healthcare
Protein folding, drug development, personalized medicine, improved medical imaging

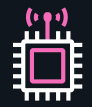
Media and entertainment
Video game generation, upscaling content, face synthesis, film preservation and coloring

Automotive
Autonomous vehicles, design parts for fuel efficiency

Financial services
Risk management, fraud detection

Consumer goods
Optimize pricing and inventory, correctly flag product brand and category

Energy and utilities
Design renewable energy sources optimized for geo, predictive maintenance



Technology hardware
Chip design, robotics

Amazon SageMaker

Foundation models available on SageMaker JumpStart for self-managed access

Publicly available

stability.ai

Models

Text2Image

Upscaling

Tasks

Generate photo-realistic images from text input

Improve quality of generated images

Features

Fine-tuning on SD 2.1 model

alexa+

Models

AlexaTM 20B

Tasks

Machine translation

Question answering

Summarization

Annotation

Data generation

Hugging Face

Models

Flan T-5 models (8 variants)

DistilGPT2, GPT2

Bloom models (3 variants)

Tasks

Machine translation

Question answering

Summarization

Annotation

Data generation

Features

Fine-tuning

Proprietary models

co:here

Models

Cohere generate-med

Tasks

Text generation

Information extraction

Question answering

Summarization

LightOn

Models

Lyra-Fr 10B

Tasks

Text generation

Keyword extraction

Information extraction

Question answering

Summarization

Sentiment analysis

Classification

AI21labs

Models

Jurassic-2 Grande 17B

+ 5 others

Tasks

Text generation

Long-form generation

Summarization

Paraphrasing

Chat

Information extraction

Question answering

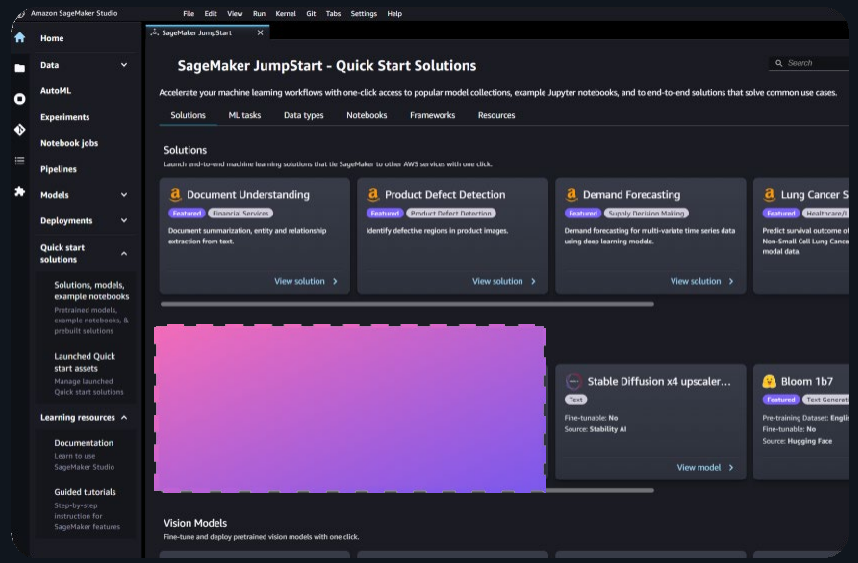
Classification

aws

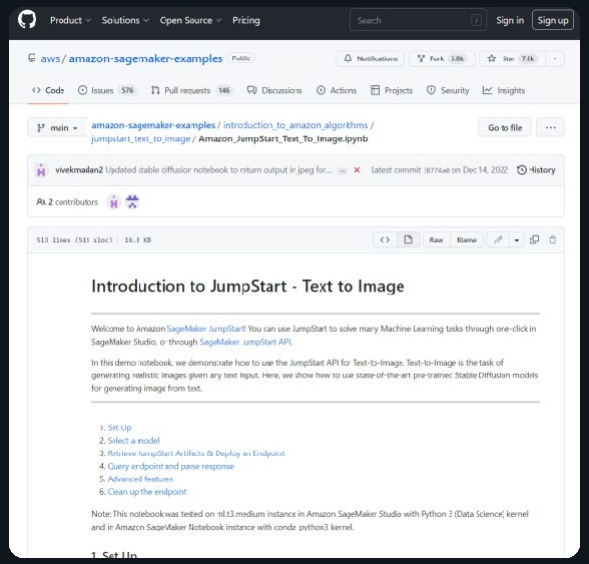
+ Research use only; not for commercial use (yet)

3 ways to use foundation models with SageMaker JumpStart

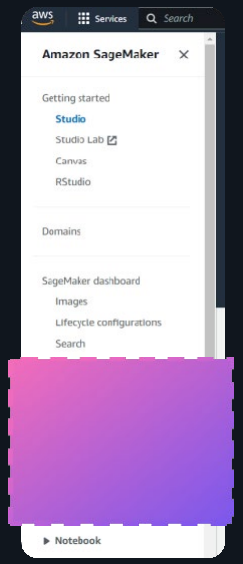
SageMaker Studio
One-step deploy



SageMaker Notebooks



AWS Management Console Preview



Foundation models with SageMaker JumpStart: How it works

Amazon SageMaker JumpStart

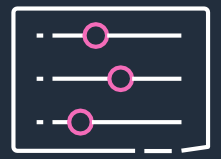
Access and try out public and proprietary FMs, and customize and integrate them into your generative AI applications

Browse

Browse public and proprietary FMs

Experiment

Experiment with FMs before choosing a model for deployment



Customize

Customize selected FM with your own dataset without training from scratch

Deploy

Deploy the model and run inference for your generative AI use case