
Generative AI on AWS

Prashant Sahu

Create an account and sign-in



Sign in as IAM user

Account ID (12 digits) or account alias

avtrainers

IAM user name

dipanjan

Password

.....

Remember this account

Sign in

Sign in using root user email

Forgot password?

Generative AI podcast

This episode covers
generative AI solutions
for manufacturing from
industry leader, Autodesk

[Learn more ›](#)

English ▾

[Terms of Use](#) [Privacy Policy](#) © 1996-2024, Amazon Web Services, Inc. or its affiliates.

Search in Services and go to the Amazon Bedrock Page

The screenshot shows the AWS Console Home page. At the top, there is a search bar with the placeholder "Search" and a keyboard shortcut "[Alt+S]". Below the search bar, the "Recently visited" section lists several services: IAM, Amazon Bedrock (which is circled in red), Billing and Cost Management, Amazon SageMaker, and Amazon Q. A red arrow points from the search bar down to the "Amazon Bedrock" entry. On the right side of the screen, there is a sidebar titled "Applications (0)" with a "Create application" button. Below the sidebar, there is a "Find applications" search bar and a message stating "No applications. Get started by creating an application." with a "Create application" button. At the bottom of the screen, there are three cards: "Welcome to AWS", "AWS Health", and "Cost and usage".

Click on Get Started

The screenshot shows the AWS Machine Learning homepage for the Amazon Bedrock service. At the top, there's a navigation bar with the AWS logo, a 'Services' dropdown, a search bar containing 'Search', a keyboard shortcut '[Alt+S]', and various status icons like signal strength and battery level. Below the navigation, a sidebar has a 'Machine Learning' section. The main content area features a large heading 'Amazon Bedrock' and a sub-headline: 'The easiest way to build and scale generative AI applications with foundation models (FMs)'. To the right, a call-to-action box titled 'Try Bedrock' contains a prominent orange 'Get started' button, which is circled in red. The overall background is dark.

Overview

Amazon Bedrock is a fully managed service that makes FMs from leading AI startups and Amazon available via an API, so you can choose from a wide range of FMs to find the model that is best suited for your use case. With Bedrock's serverless experience, you can get started quickly, privately customize FMs with your own data, and easily integrate and deploy them into your applications using the AWS tools without having to manage any infrastructure.

Benefits

Accelerate development of generative AI applications using FMs through an API, without managing infrastructure.

Choose FMs from AI21 Labs, Anthropic, Stability AI, and Amazon to find the right FM for your use case.

Model Access -> Manage Model Access

AWS Services Search [Alt+S] Oregon dipanjan@avt

Amazon Bedrock <

Getting started Overview Examples Providers

Foundation models Base models Custom models

Playgrounds Chat Text Image

Safeguards Guardrails Watermark detection

Orchestration Knowledge bases Agents

Assessment & deployment Model Evaluation Provisioned Throughput

Model access 3 new Settings User guide EULA Bedrock Service Terms

Amazon Bedrock > Model access

Model access Info

To use Bedrock, you must request access to Bedrock's FMs. To do so, you will need to have the correct IAM Permissions. For certain models, you may first need to submit use case details before you are able to request access. More information about these models is available on the Providers page.

Base models (27)

Models	Access status	Modality	EULA
AI21 Labs			
Jurassic-2 Ultra	Request failed	Text	EULA
Jurassic-2 Mid	Request failed	Text	EULA
Amazon			
Titan Embeddings G1 - Text	Access granted	Embedding	EULA
Titan Text G1 - Lite	Access granted	Text	EULA
Titan Text G1 - Express	Access granted	Text	EULA
Titan Image Generator G1	Access granted	Image	EULA
Titan Multimodal Embeddings G1	Access granted	Embedding	EULA
Anthropic			
Claude 3 Opus	Request failed	Text & Vision	EULA
Claude 3 Sonnet	Request failed	Text & Vision	EULA
Claude 3 Haiku	Request failed	Text & Vision	EULA
Claude	Request failed	Text	EULA
Claude Instant	Request failed	Text	EULA
Cohere			

Manage model access

Analytics /idhya

Open Sagemaker Studio based on your previously created domain and user

The screenshot shows the AWS SageMaker Studio interface. On the left, there's a navigation sidebar with options like 'Getting started', 'Studio' (which is circled in red), 'Studio Lab', 'Canvas', 'RStudio', 'TensorBoard', and 'Profiler'. Below that is a section for 'Admin configurations' with 'Domains', 'Role manager', 'Images', and 'Lifecycle configurations'. The main content area has a dark header 'Amazon SageMaker' and a sub-header 'Amazon SageMaker'. It features a large title 'SageMaker Studio' and a subtitle 'The first fully integrated development environment (IDE) for machine learning.' At the bottom of this area is a button labeled 'How it works'. To the right, there's a 'Get Started' panel with two dropdown menus: 'Select Domain' set to 'QuickSetupDomain-20240423T105429' and 'Select user profile' set to 'dj-test'. A large orange button at the bottom of this panel is labeled 'Open Studio' and is also circled in red. Red arrows point from the circled 'Studio' in the sidebar to the 'Select Domain' dropdown, and from the circled 'Open Studio' button to the 'Open Studio' button itself.

Go to Jupyterlab

The screenshot shows the SageMaker Studio interface. On the left, a sidebar lists various applications and services. At the top, a blue banner announces the availability of Llama 3 models on SageMaker JumpStart. The main content area is titled 'Home' and provides an overview of ML workflows. A prominent feature is the 'JupyterLab' section, which includes a button to 'Create, manage, and run durable instances of JupyterLab using spaces'. A red circle highlights the 'JupyterLab' icon and its associated text. Below this, there's a 'View JupyterLab spaces >' link. The 'Overview' tab is selected in the navigation bar. Other sections visible include 'Prebuilt and automated solutions' (JumpStart), 'Model evaluations', and 'Workflows and tasks'.

SageMaker Studio

Applications (5)

JupyterLab RStudio Canvas

Code Edi... Studio Cl...

Home

Running instances

Data

Auto ML

Experiments

Jobs

Pipelines

Models

JumpStart

Deployments

Llama 3 models are now available on SageMaker JumpStart
Deploy Llama3 8B, 70B, and instruct variants today

Home

Launch workflows, manage your applications and spaces, and view getting started materials

Overview Getting started

Overview

Start a new ML workflow or jump back into your workflow

JupyterLab

Create, manage, and run durable instances of JupyterLab using spaces

View JupyterLab spaces >

Prebuilt and automated solutions

JumpStart

Quickly deploy, fine-tune, and evaluate pre-trained models

Model evaluations

Evaluate LLMs for model quality and responsibility

Workflows and tasks

Prepare data

- Connect to data sources
- Transform, analyze, and export data

Build, train, tune models

- View all training jobs
- Create an AutoML experiment

Analytics Vidhya

Select ml.t3.medium, >20GB storage and Run Space

The screenshot shows the SageMaker Studio interface. On the left, there's a sidebar with various application icons: JupyterLab (orange), RStudio (blue), Canvas (purple), Code Editor (grey), and Studio Client (green). The main area displays a JupyterLab space named "dj-test-jupyter". The space has a status of "Stopped" and is using an "ml.t3.medium" instance. A red circle highlights the "Run space" button, and another red circle highlights the "Instance" dropdown set to "ml.t3.medium". Below this, the "Space Settings" section is visible, showing a "Storage (GB)" input field set to "50", a "Lifecycle Configuration" dropdown set to "No Script", and an "Attach custom EFS filesystem - optional" dropdown set to "None".

Start Jupyterlab

The screenshot shows the SageMaker Studio interface with the following details:

- Title Bar:** SageMaker Studio
- Address Bar:** studio-d-c3o5vkuonpqg.studio.us-west-2.sagemaker.aws/jupyterlab/dj-test-jupyter
- Header:** SageMaker Studio > Jupyterlab > Dj Test Jupyter
- Feedback:** Provide feedback
- Left Sidebar (Applications):** Applications (5) - JupyterLab (selected), RStudio, Canvas, Code Edi..., Studio Cl...
- Space Overview:** dj-test-jupyter (Private) - JupyterLab • 50 GB • ml.t3.medium
- Control Buttons:** Stop space, Open JupyterLab (circled in red), Status (Running), Instance (ml.t3.medium), Image (SageMaker Distribution 1.6)
- Space Settings:** Space Settings (New)
 - A space is a named, self-contained, durable storage container (like a filesystem), to which an app can be attached.
 - Storage (GB): 50
 - Lifecycle Configuration: No Script
 - Attach custom EFS filesystem - optional: None
- Bottom Navigation:** Home, Running instances, Data, Auto ML, Experiments, Jobs, Pipelines

Generative AI on AWS

What is generative AI?



AI that can produce original content close enough to human generated content for real-world tasks



Powered by foundation models pre-trained on large sets of data with several hundred billion parameters



Tasks can be customized for specific domains with minimal fine-tuning



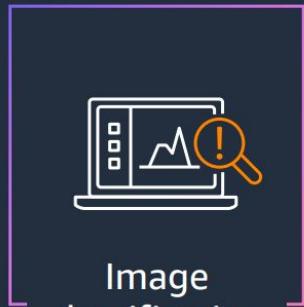
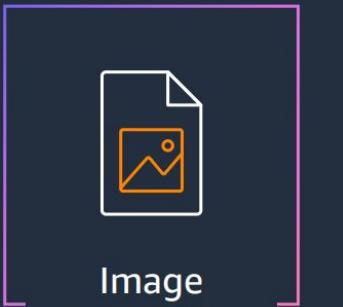
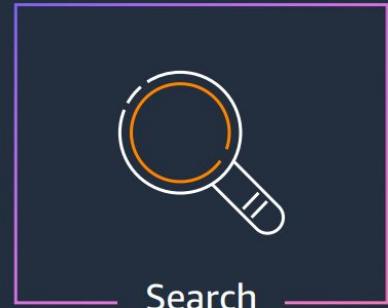
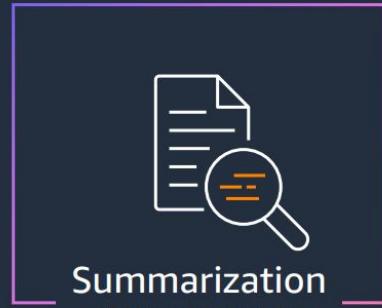
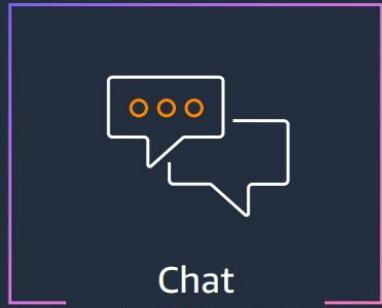
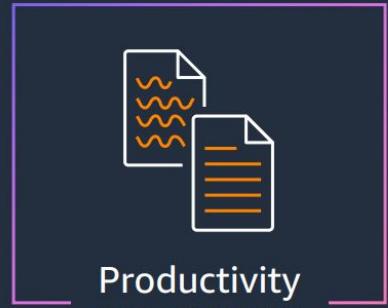
Applicable to many use cases like text summarization, question answering, digital art creation, code generation, etc.



Reduces time and cost to develop ML models and innovate faster

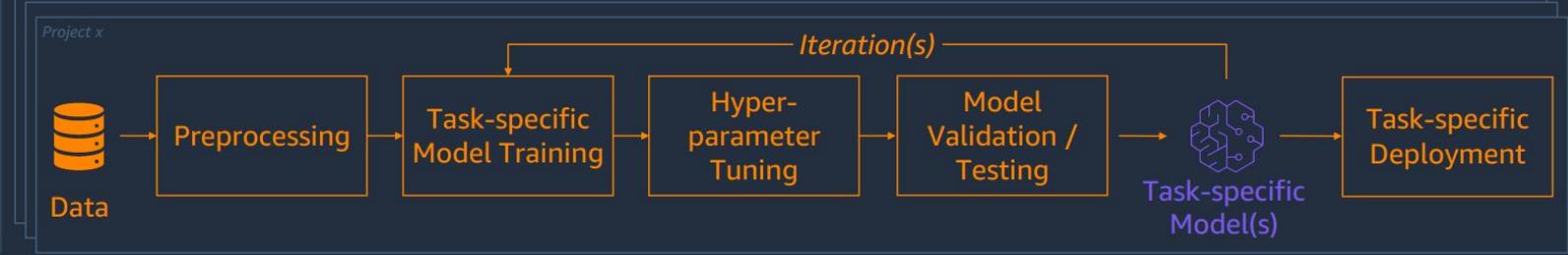
Generative AI on AWS

Generative AI is emerging across a range of domains ...

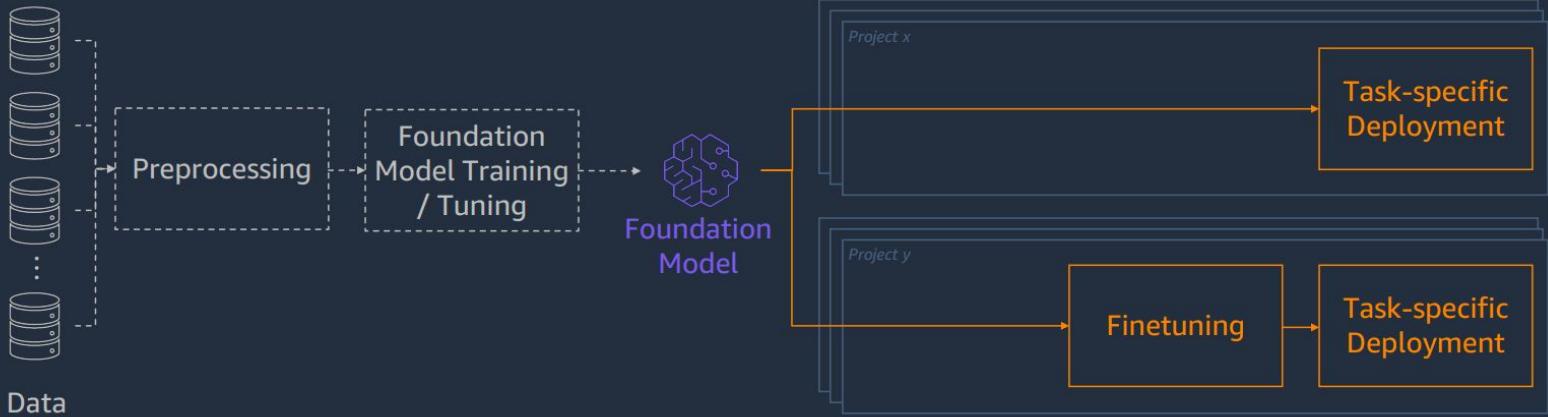


AWS Foundation Models

Conventional
ML flow



ML flow with
Foundation
Models



Gen AI on AWS

Option 1: Foundation models available through SageMaker JumpStart

The screenshot shows the "Getting started with Amazon SageMaker JumpStart" page. At the top, there's a search bar with the placeholder "foundation models". Below it, a sidebar has a "Sort By" dropdown set to "Popularity". The main area displays several foundation models:

- Stable Diffusion 2** (StabilityAI): A pink card featuring the stability.ai logo and a yellow emoji. It says "Text Generation" and "Proprietary Models". Below it, there's a detailed description of the model.
- AlexaTM (20b)** (HuggingFace): A white card featuring the Alexa logo. It says "Text Generation" and "Proprietary Models". Below it, there's a detailed description of the model.
- Bloom 1b7** (HuggingFace): A white card featuring the Bloom logo. It says "Text Generation" and "Proprietary Models". Below it, there's a detailed description of the model.

At the bottom of the page, there's a summary section with cards for "AI21 Labs", "LightOn", "co:here", and "alexa".

You can try out models for free..!



Gen AI on AWS

SageMaker JumpStart

Easily access ML assets and quickly bring ML applications to market



Machine Learning Hub for SageMaker

Browse through ~400 contents including, built-in algorithms with pre-trained models, (New) Foundation Models, solution templates, and example notebooks



Pre-built training and inference scripts

Compatible with SageMaker and configurable with custom dataset



UI as well as API based machine learning

Use UI for single click model deployment or API for Python SDK based workflow



Notebook with examples

JumpStart lets you jump into notebook to use selected model with examples to guide customers through entire ML workflow

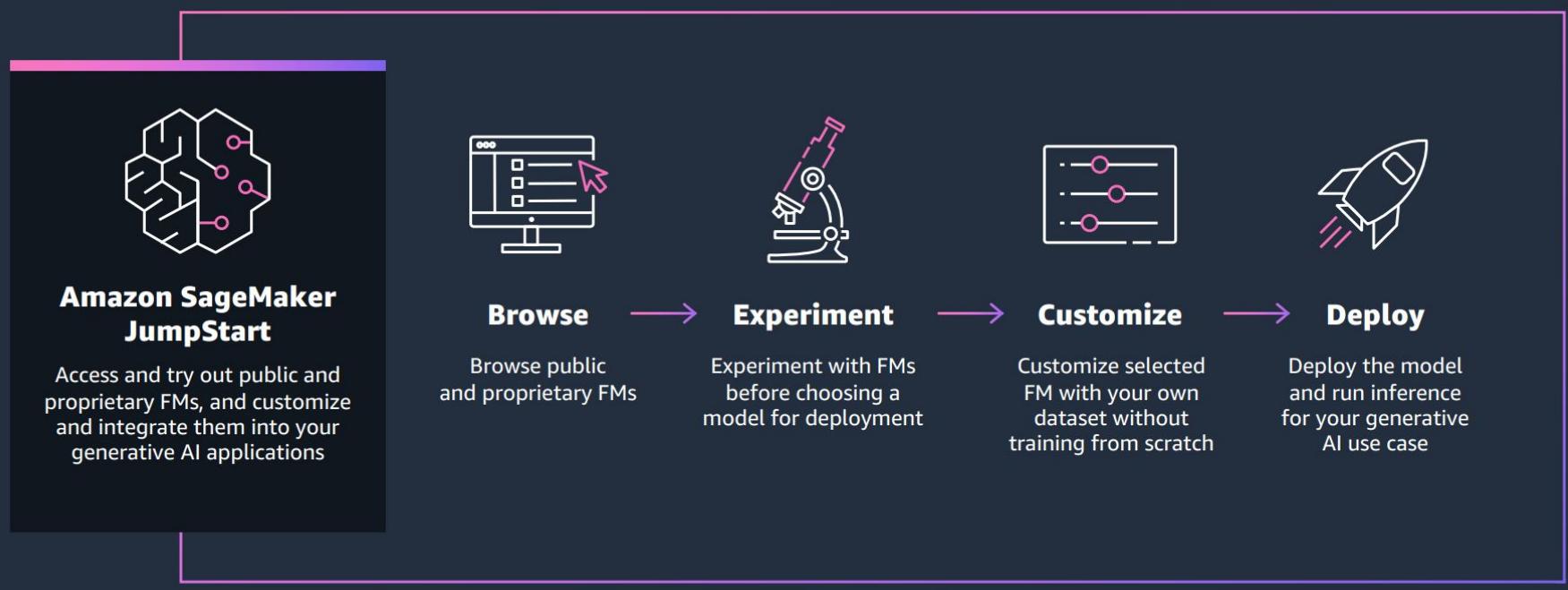


(New) Share and collaborate within an organization

Share models and notebooks with others within your organization, and allow them to train with their own data or deploy as-is for inferencing

Gen AI on AWS

Foundation models with SageMaker JumpStart: How it works



Gen AI on AWS

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

Models

AlexaTM 20B

Tasks

Machine translation

Question answering

Summarization

Annotation

Data generation

Models

Flan T-5 models (8 variants)

DistilGPT2, GPT2

Bloom models (3 variants)

Tasks

Machine translation

Question answering

Summarization

Annotation

Data generation

co:here



Models

Cohere generate-med

Tasks

Text generation

Information extraction

Question answering

Summarization

Models

Lyra-Fr 10B

Tasks

Text generation

Keyword extraction

Information extraction

Question answering

Summarization

Sentiment analysis

Classification

Models

Jurassic-1 Grande 17B

Tasks

Text generation

Long-form generation

Summarization

Paraphrasing

Chat

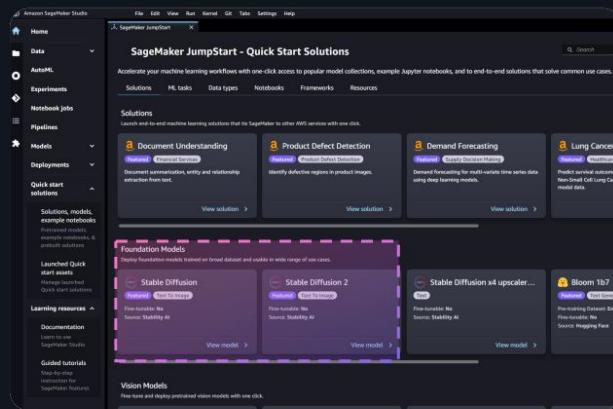
Information extraction

Classification

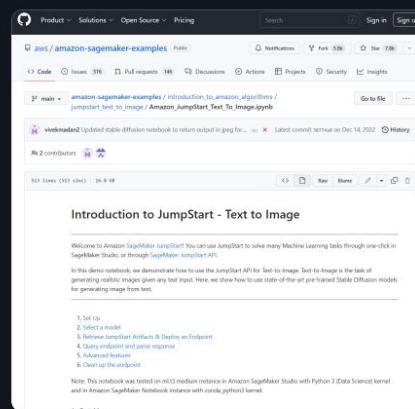
Gen AI on AWS

3 ways to use foundation models with SageMaker JumpStart

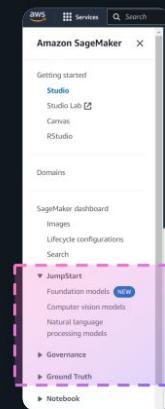
SageMaker Studio One-step deploy



SageMaker Notebooks



AWS Management Console Preview



Gen AI on AWS

Key factors in decision making



Cost

Optimize for cost with a variety of models, size, and instance for your needs with AWS pay-as-you go pricing



Accuracy

Use highly accurate models per HELM benchmarks



Speed (latency)

Optimize for performance with different model sizes and instance types



Ease of use

Instantly try models in playground; deploy with SageMaker using managed inference scripts



Data security

Host models on customer-dedicated endpoints inside your VPC

Gen AI on AWS



Amazon SageMaker

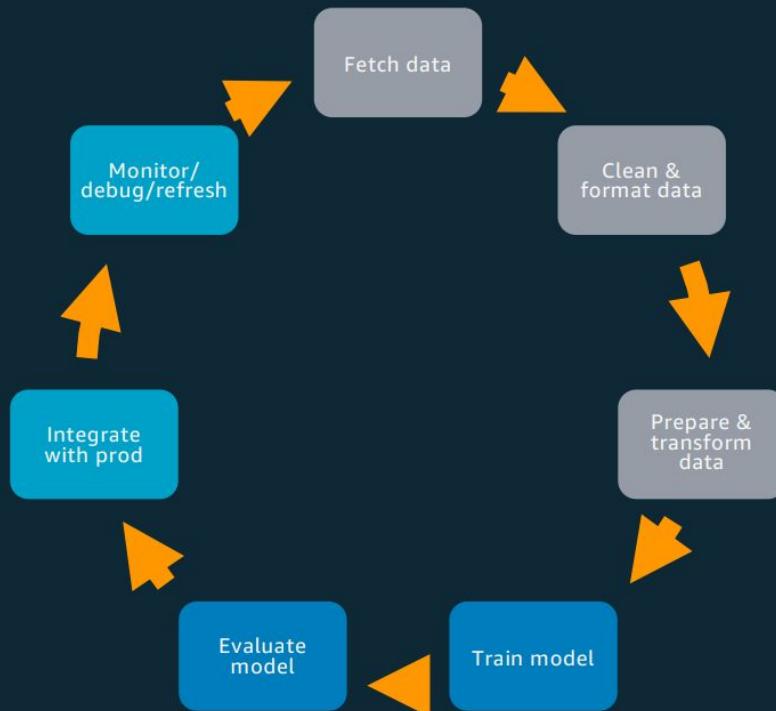
A **fully managed service** that enables **data scientists** and **developers** to quickly and easily **build** machine-learning based models **into production** smart applications.

Gen AI on AWS

Machine learning process is hard...

3. Deployment

- Setup and manage inference clusters
- Manage and auto scale inference APIs
- Testing, versioning, and monitoring



1. Data wrangling

- Setup and manage Notebook environments
- Get data to notebooks securely

2. Experimentation

- Setup and manage clusters
- Scale/distribute ML algorithms

Gen AI on AWS

Amazon SageMaker

Build, train, and deploy machine learning models at scale



End-to-End
Machine Learning
Platform



Zero setup



TensorFlow



Pay by the second

Gen AI on AWS



Amazon SageMaker

1



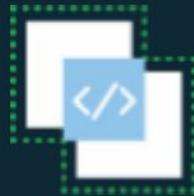
Notebook Instances

2



Algorithms

3



ML Training Service

4



ML Hosting Service

Gen AI on AWS

1

Zero Setup For Exploratory Data Analysis



Notebook Instances



"Just add data"

- Recommendations/Personalization
- Fraud Detection
- Forecasting
- Image Classification
- Churn Prediction
- Marketing Email/Campaign Targeting
- Log processing and anomaly detection
- Speech to Text
- More...

CS

Gen AI on AWS

②

Amazon SageMaker: 10x better algorithms



Algorithms

- Matrix Factorization
- Regression
- Principal Component Analysis
- K-Means Clustering
- Gradient Boosted Trees
- And More!

Amazon provided Algorithms



mxnet

TensorFlow™

Bring Your Own Script (SM builds the Container)



SM Estimators in
Apache Spark



Bring Your Own Algorithm (You build the Container)



Streaming datasets, for
cheaper training



Train faster, in a
single pass



Greater
reliability on
extremely large
datasets

tics
a

Gen AI on AWS

3

Managed Distributed Training with Flexibility



ML Training Service



Batch Training Data



Training Code



Save Model Artifacts



Save Inference Model



Secured



- Matrix Factorization
- Regression
- Principal Component Analysis
- K-Means Clustering
- Gradient Boosted Trees
- And More!

Amazon provided Algorithms



Bring Your Own Script (SM builds the Container)



SM Estimators in
Apache Spark



Bring Your Own Algorithm (You build the Container)

CPU

GPU

HPO

Fully managed



ics
a

Gen AI on AWS

4

Easy Model Deployment to Amazon SageMaker

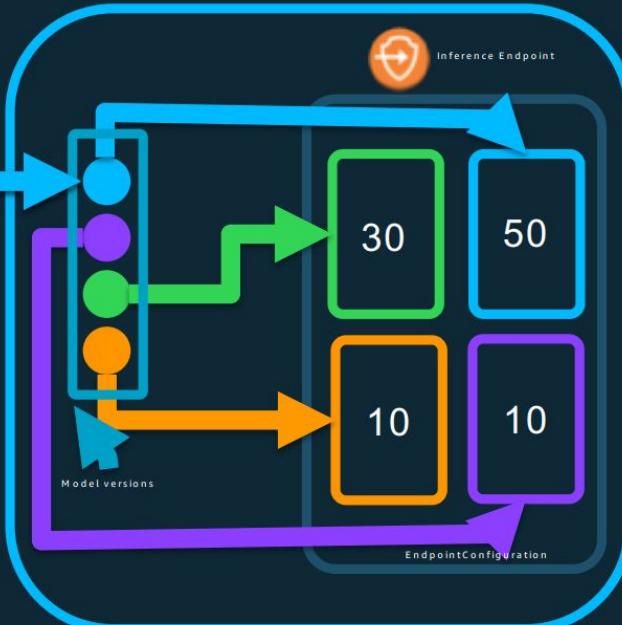


ML Hosting Service

Versions of the same inference code saved in inference containers. Prod is the primary one, 50% of the traffic must be served there!



Amazon ECR



Amazon SageMaker



Production Variant

One-Click!



Amazon Provided Algorithms



Analytics
Vidhya

NEW

Option 2 - Amazon Bedrock

The easiest way to build and
scale generative AI
applications with FMs

Gen AI on AWS

Amazon Bedrock key benefits



Accelerate development of generative AI applications using FMs through an API, without managing infrastructure

Choose FMs from AI21 Labs, Anthropic, Stability AI, and Amazon to find the right FM for your use case

Privately customize FMs using your organization's data

Enhance your data protection using comprehensive AWS security capabilities

Use AWS tools and capabilities that you are familiar with to deploy scalable, reliable, and secure generative AI applications

Gen AI on AWS

Bedrock supports a wide range of foundation models

FMs from Amazon



Titan Text



Titan
Embeddings

FMs from AI21 Labs, Anthropic, and Stability AI



Jurassic-2



Claude

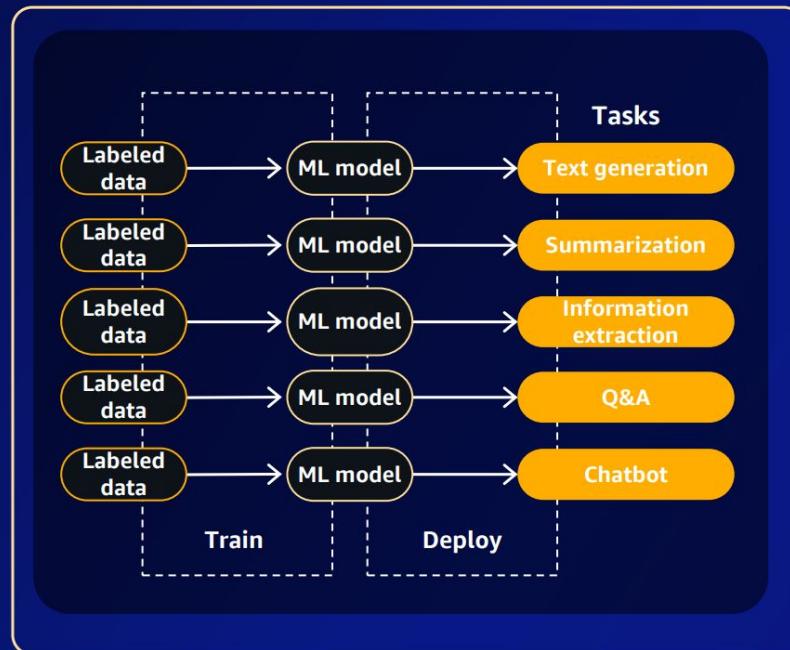


Stable
Diffusion

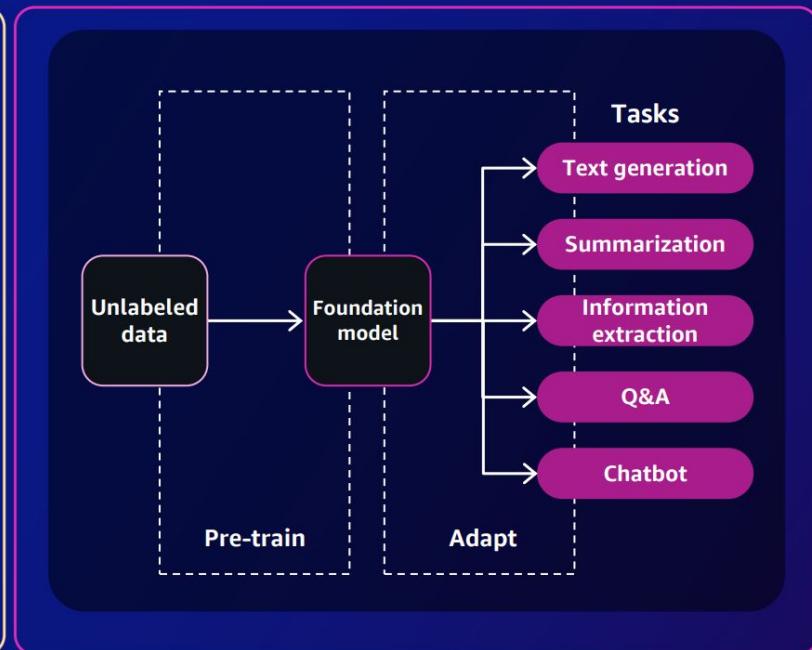
... with more to come

Gen AI on AWS

How foundation models differ from other ML models



Traditional ML models



Foundation models

Gen AI on AWS

Types of foundation models

Input

FM

Output

"Summarize articles on the impact
of walking on heart health"

Text-to-text
Generate text from natural language prompts

"Ten thousand steps per day
is optimum for maintaining
a healthy heart"

"hand soap"

Text-to-embeddings
Generate numerical representation of text

Numerical representation of
"Hand soap refills
Hand soap dispenser
Hand soap antibacterial"

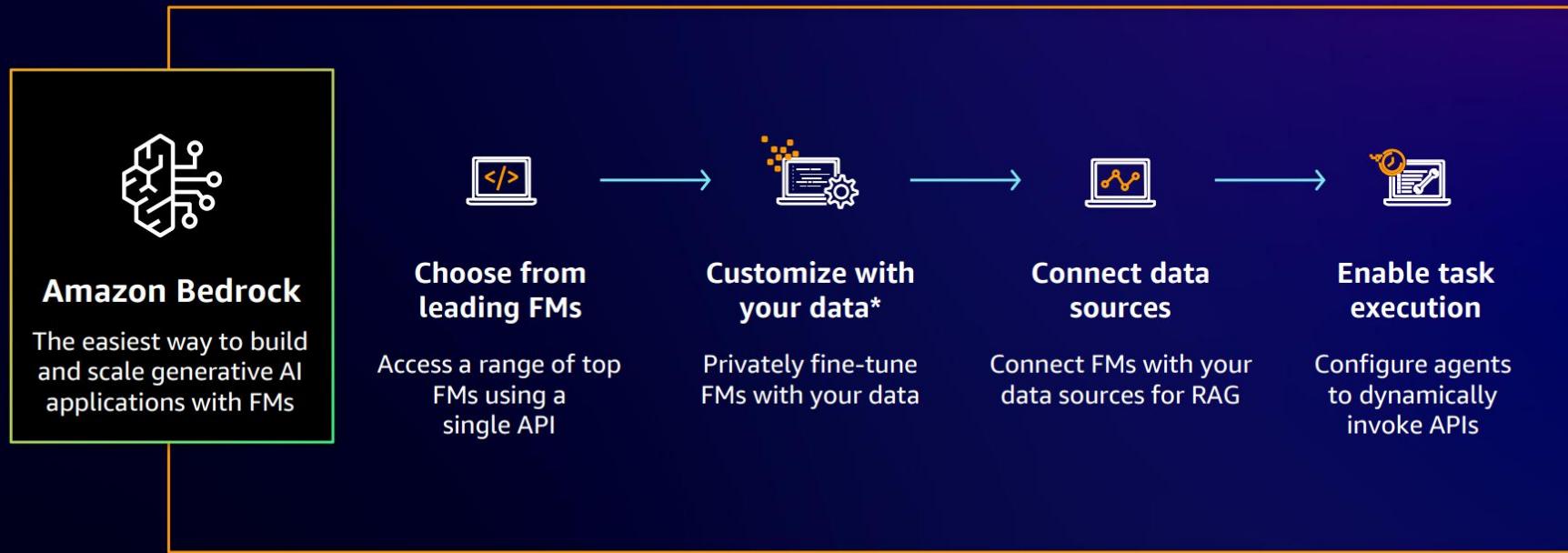
"a photo of an astronaut
riding a horse on mars"

Multimodal
Create and edit images using
natural language prompts



Gen AI on AWS

How does Amazon Bedrock work?



*your content is not used to improve the base models and is not shared with third-party model providers.

How to customise FMs

Prompt Engineering	Retrieval Augmented Generation (RAG)	Instruction fine-tuning	Fine-tuning	Pretraining
Guiding the model to generate useful response by teaching it the “pattern” of desired output using context instructions, examples and output indicators	Text generation based on specific corpus of data to generate accurate responses without hallucination	Fine-tuning a language model on a collection of tasks improves the zero-shot performance of language models on unseen tasks	Fine-tuning a model using proprietary or domain specific data to improve output quality and domain-relevant results	Retraining a model using a different dataset or building a model from scratch

Increasing complexity



Increasing cost

Gen AI on AWS

How to customise FMs

Prompt Engineering

Guiding the model to generate useful response by teaching it the “pattern” of desired output using context instructions, examples and output indicators

Retrieval Augmented Generation (RAG)

Text generation based on specific corpus of data to generate accurate responses without hallucination

Few-shot prompting

Label the following sentence by sentiment: **Frankly, my dear, I don't give a damn**

Labels: Indifferent, Apathetic, Neutral

##

Label the following sentence by sentiment: **I'm gonna make him an offer he can't refuse**

Labels: Threatening, Scheming

##

Label the following sentence by sentiment: **think before you act**

Labels:

Finetuning

Training a model on proprietary or specific data to achieve output with domain-specific results

Pretraining

Retraining a model using a different dataset or building a new model from scratch

Increasing cost

Gen AI on AWS

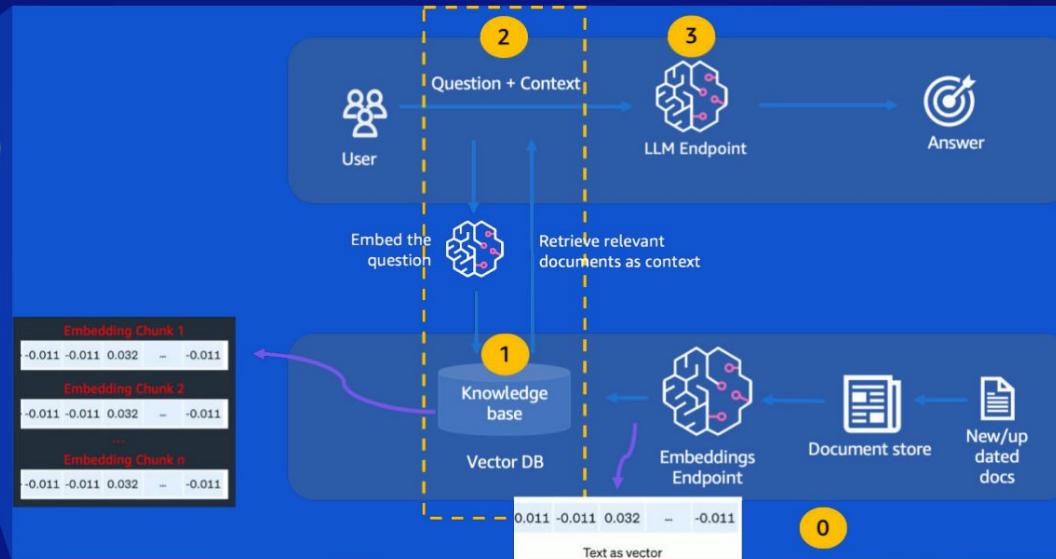
How to customise FMs

Prompt Engineering

Guiding the model to generate useful response by teaching it the “pattern” of desired output using context instructions, examples and output indicators

Retrieval Augmented Generation (RAG)

Text generation based on specific corpus of data to generate accurate responses without hallucination



Gen AI on AWS

How to customise FMs

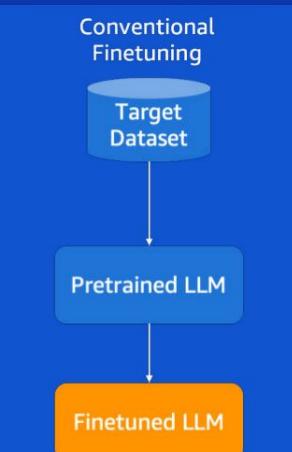
Prompt Engineering

Guiding the model to generate useful response by teaching it the “pattern” of desired output using context instructions, examples and output indicators

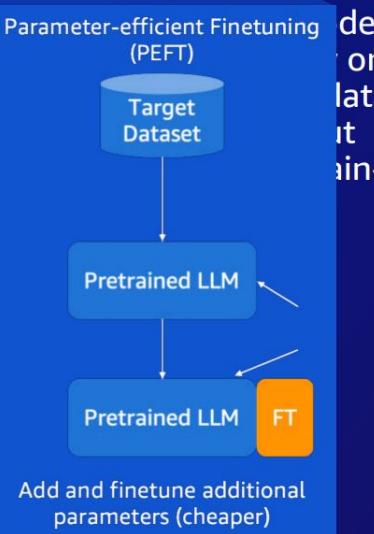
Retrieval Augmented Generation (RAG)



Instruction fine-tuning



Fine-tuning



Pretraining

Retraining a model using a different dataset or building a model from scratch

Gen AI on AWS

Amazon Bedrock

THE EASIEST WAY TO BUILD AND SCALE GENERATIVE AI APPLICATIONS WITH FMS

- Accelerate development of generative AI applications using FMs through an API
- No need to manage infrastructure
- Privately customize FMs using your organization's data
- Comprehensive AWS security capabilities
- Enable generative AI apps to complete tasks in just a few clicks using **agents** for Bedrock



Amazon Titan

Text summarization, generation, classification, open-ended Q&A, information extraction, embeddings, and search



Jurassic-2

Multilingual LLMs for text generation in Spanish, French, German, Portuguese, Italian, and Dutch



new Claude 2

LLM for thoughtful dialogue, content creation, complex reasoning, creativity, and coding, based on Constitutional AI and harmlessness training



new Command + Embed

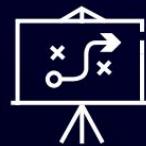
Text generation model for business applications and embeddings model for search, clustering, or classification in 100+ languages



new Stable Diffusion XL 1.0

Generation of unique, realistic, high-quality images, art, logos, and designs

Enabling foundation models to complete tasks



1

**Define instructions
and orchestration**



2

**Configure FM to
access data sources**



3

**Complete
actions with
API calls**



4

**Manage cloud
hosting and security**

Gen AI on AWS

Bedrock core API: InvokeModel

```
bedrock.invoke_model(  
    modelId = model_id,  
    contentType = "...",  
    accept = "...",  
    body = body)
```



Access
foundation
models

- Amazon Titan models
- Third-party models
- Fine-tuned models

```
prompt_data = """Write me a blog about making strong business decisions as a leader"""  
  
config = {"maxTokenCount":512,"stopSequences":[],"temperature":0.5,"topP":0.9}  
body = json.dumps({"inputText":prompt_data,"textGenerationConfig": config})  
modelId = "amazon.titan-tgl-large"  
accept = "*/*"  
contentType = "application/json"  
response = bedrock.invoke_model(  
    body=body, modelId=modelId, accept=accept, contentType=contentType  
)  
response_body = json.loads(response.get("body").read())  
print(response_body.get("results")[0].get("outputText"))
```

Titan Text

```
prompt_data = """Write me a blog about making strong business decisions as a leader"""  
  
body = json.dumps({"prompt": prompt_data,  
    "max_tokens_to_sample": 300,  
    "temperature": 0.5,  
    "top_k": 250,  
    "top_p": 1,  
    "stop_sequences": ["\n\nHuman:"]})  
modelId = "anthropic.claude-instant-v1"  
accept = "*/*"  
contentType = "application/json"  
response = bedrock.invoke_model(  
    body=body, modelId=modelId, accept=accept, contentType=contentType  
)  
response_body = json.loads(response.get("body").read())  
print(response_body.get("completion"))
```

Anthropic Claude

Data protection, privacy, and security

Data protection and privacy

- Your data used with Bedrock is not used for service improvement and not shared with third party model providers
- Private connectivity between Amazon Bedrock service and your virtual private cloud (VPC)
- Your data is encrypted in transit and at rest
- Customize FMs privately, retaining control over how your data is used and encrypted

Secure Generative AI application

- Use AWS security services to form your defense-in-depth security strategy
- Your customized FMs are encrypted using AWS Key Management Service (AWS KMS) keys and stored encrypted
- Control access to your customized FMs using AWS Identity and Access Management Service (IAM)

Gen AI on AWS

Model tenancy



Single-tenant endpoint



Multi-tenant endpoint

- | | |
|---|--|
| <ul style="list-style-type: none">1. Deployment available to a single customer2. Holds a single version of a baseline 1P/3P model that has been fine-tuned by a customer3. No inference request's input or output text is used to train any model(s) in the deployment4. Model deployments are inside an AWS account owned and operated by the Bedrock service team5. Model vendors have no access to any customer data | <ul style="list-style-type: none">1. Deployment available to all customers2. Holds a baseline version of each supported 1P/3P model |
|---|--|

Amazon Bedrock private fine-tuning

Organization-specific information

- Deliver more contextual and personalized responses by incorporating your organization's data
- Create secure integration with your organization data sources without retraining the FM
- Agents identify the data sources, retrieve the relevant information, incorporate the information into user query, and provide a response

Customized search capabilities

- Use Amazon Titan Embeddings or Cohere Embed to create vector of company data to enable semantic search
- The embedding can be stored in a database to use for quick and accurate searches

Base components and design patterns

- Models
- Prompts
- Memory
- Chains
- Tools
- Agents



Try Foundation Models in AWS Playground

Amazon Bedrock | us-west-2 + us-west-2.console.aws.amazon.com/bedrock/home?region=us-west-2#/image-playground

aws Services Search [Alt+S] Oregon dipanjan @ avtrainers

Amazon Bedrock < Image playground

Image playground [Info](#) Load examples

Titan Image Generator G1 v1 | ODT Change

Configurations

Mode: [Generate](#) [Edit](#)

Negative prompt: lowres

Reference image:  [X](#)

Response image: [Info](#)

Quality: Standard

Orientation: Landscape Portrait

Size:

create a photorealistic high resolution image of the city of Delhi in India [Run](#)

Getting started

- Overview
- Examples
- Providers

Foundation models

- Base models
- Custom models

Playgrounds

- Chat
- Text
- Image

Safeguards

- Guardrails
- Watermark detection

Orchestration

- Knowledge bases
- Agents

Assessment & deployment

- Model Evaluation