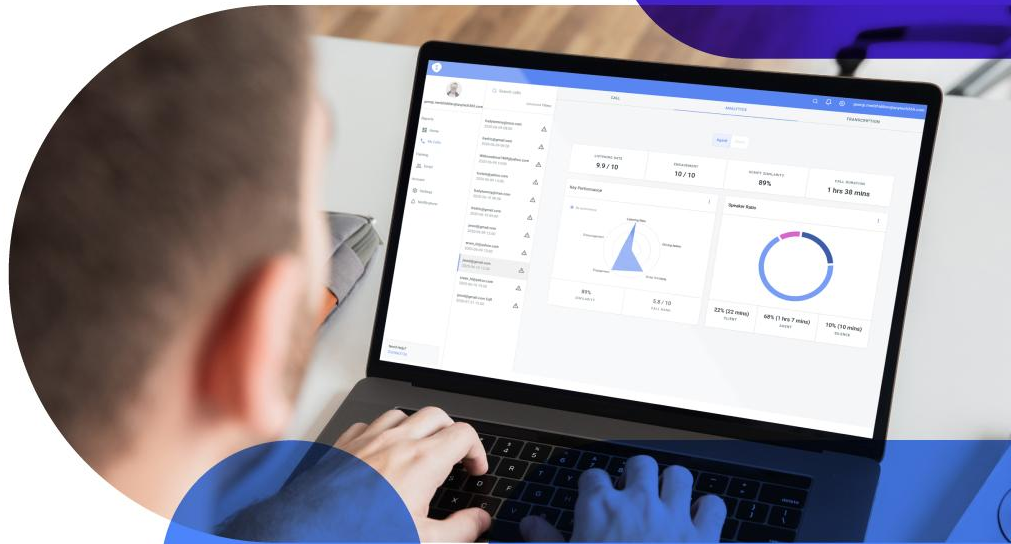# SageMaker Technical Introduction
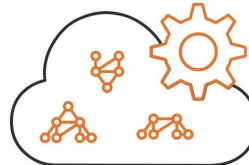
at-bay

How it works?

Label

Set up and manage labeling jobs for highly accurate training datasets within Amazon SageMaker, using active learning and human labeling

Build

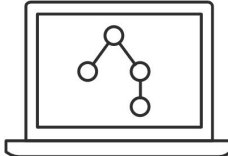Connect to other AWS services and transform data in Amazon SageMaker notebooks

Train

Use Amazon SageMaker's algorithms and frameworks, or bring your own, for distributed training

Tune

Amazon SageMaker automatically tunes your model by adjusting multiple combinations of algorithm parameters
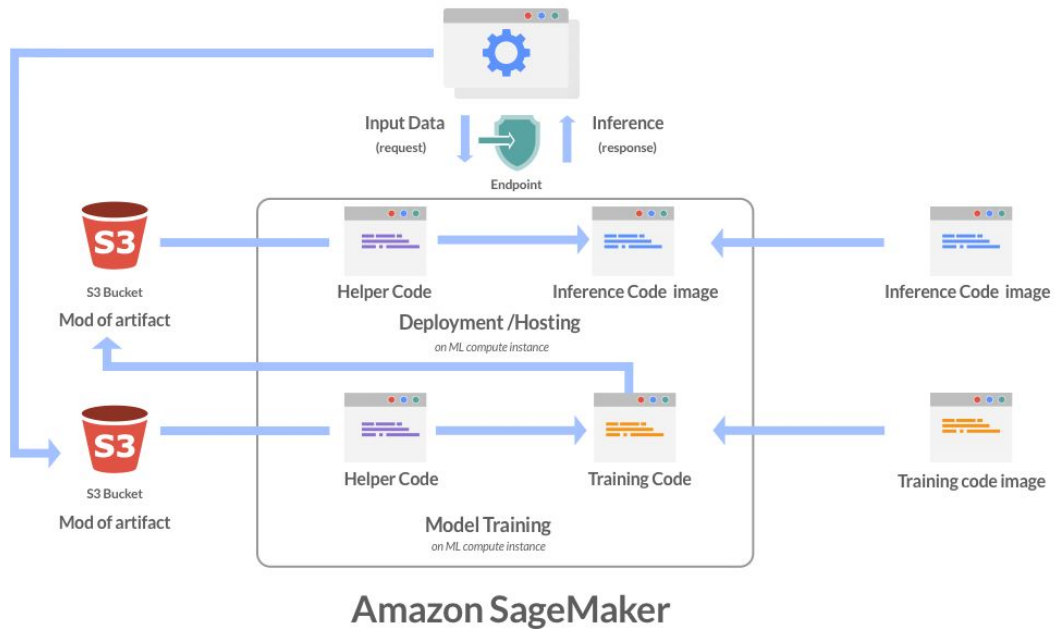
Deploy

Once training is completed, models can be deployed to Amazon SageMaker endpoints, for real-time predictions

Discover

Find, buy, and deploy ready to use model packages, algorithms, and data products in AWS Marketplace

Use Prebuilt SageMaker Docker images

PDF | Kindle | RSS

Amazon SageMaker provides containers for its built-in algorithms and prebuilt Docker images for some of the most common machine learning frameworks, such as Apache MXNet, TensorFlow, PyTorch, and Chainer. It also supports machine learning libraries such as scikit-learn and SparkML.

You can use these images from your SageMaker notebook instance or SageMaker Studio. You can also extend the prebuilt SageMaker images to include libraries and needed functionality. The following topics give information about the available images and how to use them.

> ⓘ **Note**
>
> For information on Docker images for developing reinforcement learning (RL) solutions in SageMaker, see SageMaker RL Containers ↗.

Topics

- Prebuilt SageMaker Docker Images for Deep Learning
- Prebuilt Amazon SageMaker Docker Images for Scikit-learn and Spark ML
- Train a Deep Graph Network
- Extend a Prebuilt Container

https://docs.aws.amazon.com/sagemaker/latest/dg/docker-containers-prebuilt.html

Automating Model Retraining

Create a Choice State Step

In the following cell, we create a choice step in order to build a dynamic workflow. This choice step branches based off of the results of our SageMaker training step: did the training job fail or should the model be saved and the endpoint be updated? We will add specfic rules to this choice step later on in section 8 of this notebook.

```
In [ ]: check_accuracy_step = steps.states.Choice("Accuracy > 90%")
```

Create an Endpoint Configuration Step

In the following cell we create an endpoint configuration step. See EndpointConfigStep in the AWS Step Functions Data Science SDK documentation to learn more.

```
In [ ]: endpoint_config_step = steps.EndpointConfigStep(
            "Create Model Endpoint Config",
            endpoint_config_name=execution_input["ModelName"],
            model_name=execution_input["ModelName"],
            initial_instance_count=1,
            instance_type="ml.m4.xlarge",
        )
```

SERVERLESS

Pay as you go inference with AWS Lambda (Container Image Support)

This repository contains resources to help you deploy Lambda functions based on Python and Java Docker Images.

The applications deployed illustrate how to perform inference for scikit-learn, XGBoost, TensorFlow and PyTorch models using Lambda Function.
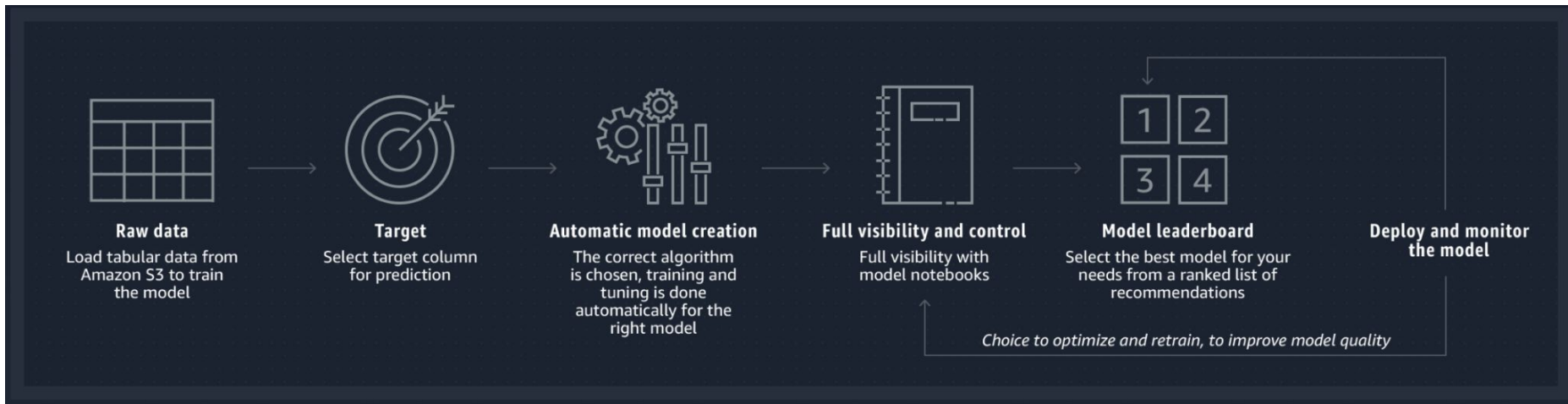
Overview

AWS Lambda is one of the most cost effective service that lets you run code without provisioning or managing servers.

It offers many advantages when working with serverless infrastructure. When you break down the logic of your machine learning service into a single Lambda function for a single request, things become much simpler and easy to scale.
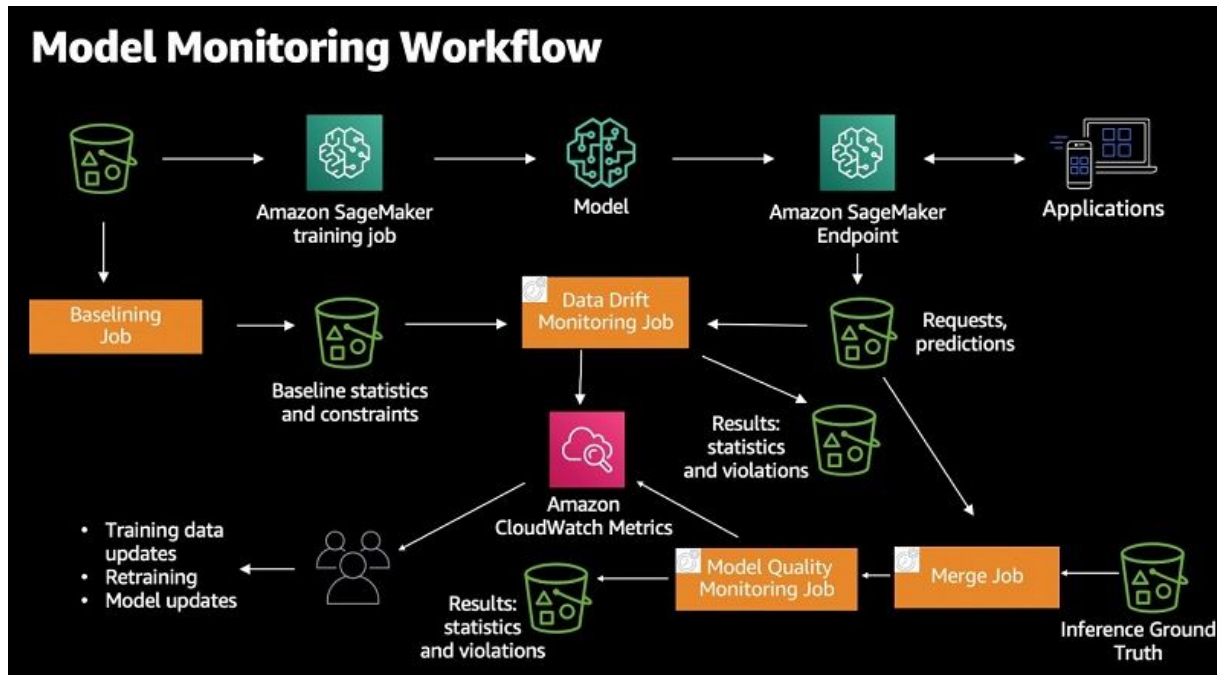
You can forget all about the resource handling needed for the parallel requests coming into your model.

If your usage is sparse and tolerable to a higher latency, Lambda is a great choice among various solutions.

Sedric

Auto-Pilot

Raw data
Load tabular data from Amazon S3 to train the model

Target
Select target column for prediction

Automatic model creation
The correct algorithm is chosen, training and tuning is done automatically for the right model

Full visibility and control
Full visibility with model notebooks

Model leaderboard
Select the best model for your needs from a ranked list of recommendations

Deploy and monitor the model

Choice to optimize and retrain, to improve model quality

Monitoring

Model Monitoring Workflow

Amazon SageMaker training job → Model → Amazon SageMaker Endpoint → Applications

Baselining Job → Baseline statistics and constraints → Data Drift Monitoring Job ← Requests, predictions

Results: statistics and violations

Amazon CloudWatch Metrics

- Training data updates
- Retraining
- Model updates

Results: statistics and violations ← Model Quality Monitoring Job ← Merge Job ← Inference Ground Truth
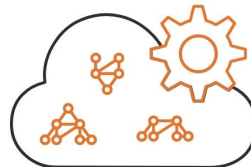
Summary

Label

Set up and manage labeling jobs for highly accurate training datasets within Amazon SageMaker, using active learning and human labeling
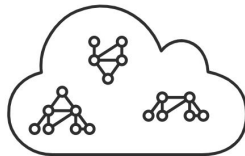
Build

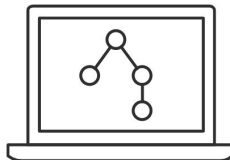Connect to other AWS services and transform data in Amazon SageMaker notebooks

Train

Use Amazon SageMaker's algorithms and frameworks, or bring your own, for distributed training

Tune

Amazon SageMaker automatically tunes your model by adjusting multiple combinations of algorithm parameters

Deploy

Once training is completed, models can be deployed to Amazon SageMaker endpoints, for real-time predictions

Discover

Find, buy, and deploy ready to use model packages, algorithms, and data products in AWS Marketplace

Thank you!