

1. Initial setup, training, and deployment

a. Setup

First, we create a SageMaker notebook instance. We select the `ml.t2.medium` instance, which has 2 vCPU and 4GB of RAM, since it is quite cheap (\$0.0464 per hour). Additionally, we only run a lightweight task in the notebook to trigger SageMaker training jobs and create endpoint. Hence, this instance type should fit our needs.

Create notebook instance

Amazon SageMaker provides pre-built fully managed notebook instances that run Jupyter notebooks. The notebook instances include example code for common model training and hosting exercises. [Learn more](#)

Notebook instance settings

Notebook instance name

Maximum of 63 alphanumeric characters. Can include hyphens (-), but not spaces. Must be unique within your account in an AWS Region.

Notebook instance type

Elastic Inference [Learn more](#)

Amazon SageMaker Notebook Instance is ending its standard support on Amazon Linux AMI (AL1). [Learn more](#)

Platform Identifier [Learn more](#)

► Additional configuration

Amazon SageMaker > Notebook Instances

Notebook instances

< 1 >

	Name	Instance	Creation time	Status	Actions
<input type="radio"/>	project04-workspace	ml.t2.medium	Dec 12, 2021 22:25 UTC	InService	Open Jupyter Open JupyterLab

b. Training

Then, we open the Jupyterlab and upload the relevant notebook and Python scripts to that instance. Based on the best hyperparameter values from the hyperparameter tuning job, we run the training job twice: 1) using single instance, and 2) using multi-instance training (in this example, we use 4 instances of `ml.m5.xlarge`).

The single instance training takes around 22 minutes, while the multi-instance training takes around 21 minutes (no huge differences here).

Preview of the single-instance training job.

dog-pytorch-2021-12-12-22-49-12-527

CloneCreate model packageStopCreate model

Job settings

Job name
dog-pytorch-2021-12-12-22-49-12-527

ARN
arn:aws:sagemaker:us-east-1:183492708471:training-job/dog-pytorch-2021-12-12-22-49-12-527

Status
✔ Completed
[View history](#)

Creation time
Dec 12, 2021 22:49 UTC

Last modified time
Dec 12, 2021 23:14 UTC

SageMaker metrics time series
Enabled

Training time (seconds)
1327

Billable time (seconds)
1327
22 minute(s) and 7 second(s) X

Managed spot training savings
0%

Tuning job source/parent
-

IAM role ARN
[arn:aws:iam::183492708471:role/service-role/AmazonSageMaker-ExecutionRole-20211212T122741](#)

Algorithm

Algorithm ARN
-

Instance type
ml.m5.xlarge

Additional volume size (GB)
30

Volume encryption key
-

Training image
763104351884.dkr.ecr.us-east-1.amazonaws.com/pytorch-

Instance count
1

Maximum runtime (s)
86400

CloudWatch > Log groups > /aws/sagemaker/TrainingJobs

/aws/sagemaker/TrainingJobs

ActionsView in Logs InsightsSearch log group

▼ Log group details

Retention
Never expire

Creation time
23 hours ago

Stored bytes
-

ARN
arn:aws:logs:us-east-1:183492708471:log-group:/aws/sagemaker/TrainingJobs:

KMS key ID
-

Metric filters
0

Subscription filters
0

Contributor Insights rules
-

Log streamsMetric filtersSubscription filtersContributor InsightsTags

Log streams (17)

dog-pytorch-2021-12-12-22-49-12-5271 match

Log streamLast event time

dog-pytorch-2021-12-12-22-49-12-527/algo-1-16393494972021-12-13 11:54:15 (UTC+13:00)

Preview of the multi-instance training job.

dog-pytorch-2021-12-12-20-47-31-112

Clone

Create model package

Stop

Create model

Job settings

Job name	Status	SageMaker metrics time series	IAM role ARN
dog-pytorch-2021-12-12-20-47-31-112	<div>Completed</div> <div>View history</div>	Enabled	arn:aws:iam::183492708471:role/service-role/AmazonSageMaker-ExecutionRole-20211212T122741
ARN	Creation time	Training time (seconds)	
arn:aws:sagemaker:us-east-1:183492708471:training-job/dog-pytorch-2021-12-12-20-47-31-112	Dec 12, 2021 20:47 UTC	1294	
Last modified time		Billable time (seconds)	
Dec 12, 2021 21:12 UTC		1294 21 minute(s) and 34 second(s)	
		Managed spot training savings	
		0%	
		Tuning job source/parent	
		-	

Algorithm

Algorithm ARN	Instance type	Additional volume size (GB)	Volume encryption key
-	ml.m5.xlarge	30	-
Training image	Instance count	Maximum runtime (s)	
763104351884.dkr.ecr.us-east-1.amazonaws.com/pytorch-training:1.4.0-cpu-py3	4	86400	
		Maximum wait time for managed spot	

/aws/sagemaker/TrainingJobs

Actions

View in Logs Insights

Search log group

Log group details

Retention	Creation time	Stored bytes	ARN
Never expire	22 hours ago	-	arn:aws:logs:us-east-1:183492708471:log-group:/aws/sagemaker/TrainingJobs:*
KMS key ID	Metric filters	Subscription filters	Contributor Insights rules
-	0	0	-

Log streams

Metric filters

Subscription filters

Contributor Insights

Tags

Log streams (16)

dog-pytorch-2021-12-12-20-47-31-112

4 matches

Log stream

dog-pytorch-2021-12-12-20-47-31-112/algo-2-1639342204

2021-12-13 10:10:56 (UTC+13:00)

dog-pytorch-2021-12-12-20-47-31-112/algo-1-1639342206

2021-12-13 10:10:49 (UTC+13:00)

dog-pytorch-2021-12-12-20-47-31-112/algo-4-1639342198

2021-12-13 10:10:26 (UTC+13:00)

dog-pytorch-2021-12-12-20-47-31-112/algo-3-1639342194

2021-12-13 10:09:54 (UTC+13:00)

In terms of model performance, the difference is negligible. The single instance training results in 580 testing loss, while the multi-instance has 581 testing loss. Following figures show the logs from each training job.

3 / 9

2021-12-13T11:54:11.362+13:00	SM_USER_ARGS=["--batch_size", "128", "--learning_rate", "0.03872229152368018"]
2021-12-13T11:54:11.362+13:00	SM_OUTPUT_INTERMEDIATE_DIR=/opt/ml/output/intermediate
2021-12-13T11:54:11.362+13:00	SM_CHANNEL_TRAINING=/opt/ml/input/data/training
2021-12-13T11:54:11.362+13:00	SM_HP_BATCH_SIZE=128
2021-12-13T11:54:11.362+13:00	SM_HP_LEARNING_RATE=0.03872229152368018
2021-12-13T11:54:11.362+13:00	PYTHONPATH=/opt/ml/code:/opt/conda/bin:/opt/conda/lib/python3.6:/opt/conda/lib/python3.6/lib-dyn...
2021-12-13T11:54:11.362+13:00	Invoking script with the following command:
2021-12-13T11:54:11.362+13:00	/opt/conda/bin/python3.6 hpo.py --batch_size 128 --learning_rate 0.03872229152368018
2021-12-13T11:54:13.363+13:00	Namespace(batch_size=128, data='/opt/ml/input/data/training', learning_rate=0.03872229152368018, model_dir='/opt/ml/model...
2021-12-13T11:54:13.363+13:00	Hyperparameters are LR: 0.03872229152368018, Batch Size: 128
2021-12-13T11:54:13.363+13:00	Data Paths: /opt/ml/input/data/training
2021-12-13T11:54:14.363+13:00	Starting Model Training
2021-12-13T11:54:14.363+13:00	Epoch: 0
2021-12-13T11:54:15.364+13:00	[2021-12-12 22:54:15.209 algo-1:46 INFO json_config.py:90] Creating hook from json_config at /opt/ml/input/config/debughook...
2021-12-13T11:54:15.364+13:00	[2021-12-12 22:54:15.209 algo-1:46 INFO hook.py:192] tensorboard_dir has not been set for the hook. SMDebug will not be e...
2021-12-13T11:54:15.364+13:00	[2021-12-12 22:54:15.209 algo-1:46 INFO hook.py:237] Saving to /opt/ml/output/tensors
2021-12-13T11:54:15.364+13:00	[2021-12-12 22:54:15.209 algo-1:46 INFO state_store.py:67] The checkpoint config file /opt/ml/input/config/checkpointconf...
2021-12-13T11:54:15.364+13:00	[2021-12-12 22:54:15.211 algo-1:46 INFO hook.py:382] Monitoring the collections: relu_input, gradients, losses
2021-12-13T11:54:15.364+13:00	[2021-12-12 22:54:15.212 algo-1:46 INFO hook.py:443] Hook is writing from the hook with pid: 46
2021-12-13T12:09:20.884+13:00	train loss: 922.0000, acc: 0.0000, best loss: 1000000.0000
2021-12-13T12:11:04.915+13:00	valid loss: 581.0000, acc: 1.0000, best loss: 581.0000
2021-12-13T12:11:04.915+13:00	Testing Model
2021-12-13T12:12:46.942+13:00	Testing Loss: 580.0
2021-12-13T12:12:46.942+13:00	Testing Accuracy: 1.0
2021-12-13T12:12:46.942+13:00	Saving Model
2021-12-13T12:12:46.942+13:00	2021-12-12 23:12:46,678 sagemaker-training-toolkit INFO Reporting training SUCCESS

2021-12-13T09:52:01.859+13:00	SM_MODULE_DIR=s3://sagemaker-us-east-1-183492708471/dog-pytorch-2021-12-12-20-47-31-112/source/sourcedir.tar.gz
2021-12-13T09:52:01.859+13:00	SM_TRAINING_ENV={"additional_framework_parameters": {}, "channel_input_dirs": {"training": "/opt/ml/input/data/training"}, "curr...
2021-12-13T09:52:01.860+13:00	SM_USER_ARGS=["--batch_size", "128", "--learning_rate", "0.03872229152368018"]
2021-12-13T09:52:01.860+13:00	SM_OUTPUT_INTERMEDIATE_DIR=/opt/ml/output/intermediate
2021-12-13T09:52:01.860+13:00	SM_CHANNEL_TRAINING=/opt/ml/input/data/training
2021-12-13T09:52:01.860+13:00	SM_HP_BATCH_SIZE=128
2021-12-13T09:52:01.860+13:00	SM_HP_LEARNING_RATE=0.03872229152368018
2021-12-13T09:52:01.860+13:00	PYTHONPATH=/opt/ml/code:/opt/conda/bin:/opt/conda/lib/python3.6:/opt/conda/lib/python3.6/lib-dyn...
2021-12-13T09:52:01.860+13:00	Invoking script with the following command:
2021-12-13T09:52:01.860+13:00	/opt/conda/bin/python3.6 hpo.py --batch_size 128 --learning_rate 0.03872229152368018
2021-12-13T09:52:02.860+13:00	Namespace(batch_size=128, data='/opt/ml/input/data/training', learning_rate=0.03872229152368018, model_dir='/opt/ml/model', ...
2021-12-13T09:52:02.860+13:00	Hyperparameters are LR: 0.03872229152368018, Batch Size: 128
2021-12-13T09:52:02.860+13:00	Data Paths: /opt/ml/input/data/training
2021-12-13T09:52:03.866+13:00	Starting Model Training
2021-12-13T09:52:03.866+13:00	Epoch: 0
2021-12-13T09:52:05.866+13:00	[2021-12-12 20:52:05.348 algo-4:45 INFO json_config.py:90] Creating hook from json_config at /opt/ml/input/config/debughook...
2021-12-13T09:52:05.866+13:00	[2021-12-12 20:52:05.348 algo-4:45 INFO hook.py:192] tensorboard_dir has not been set for the hook. SMDebug will not be exp...
2021-12-13T09:52:05.866+13:00	[2021-12-12 20:52:05.348 algo-4:45 INFO hook.py:237] Saving to /opt/ml/output/tensors
2021-12-13T09:52:05.866+13:00	[2021-12-12 20:52:05.348 algo-4:45 INFO state_store.py:67] The checkpoint config file /opt/ml/input/config/checkpointconfig...
2021-12-13T09:52:05.866+13:00	[2021-12-12 20:52:05.351 algo-4:45 INFO hook.py:382] Monitoring the collections: relu_input, losses, gradients
2021-12-13T09:52:05.866+13:00	[2021-12-12 20:52:05.352 algo-4:45 INFO hook.py:443] Hook is writing from the hook with pid: 45
2021-12-13T10:06:58.399+13:00	train loss: 1205.0000, acc: 1.0000, best loss: 1000000.0000
2021-12-13T10:08:42.440+13:00	valid loss: 581.0000, acc: 1.0000, best loss: 581.0000
2021-12-13T10:08:42.440+13:00	Testing Model
2021-12-13T10:10:25.507+13:00	Testing Loss: 581.0
2021-12-13T10:10:25.507+13:00	Testing Accuracy: 1.0
2021-12-13T10:10:25.507+13:00	Saving Model
2021-12-13T10:10:26.507+13:00	2021-12-12 21:10:25,720 sagemaker-training-toolkit INFO Reporting training SUCCESS

2. Training on EC2 instance

Here, we train a similar image classification model using an EC2 instance without changing any default hyperparameters from the starter script. We decide to use `t2.medium` instance (2 vCPU and 4 GB of RAM) since the training script only requires 5 epochs and the batch size is only (2). If the batch size is larger, we will need to use an instance with bigger RAM. We don't request a spot instance to ensure the instance is preserved to our work.

Instances (1) Info

Refresh

Connect

Instance state ▾

Actions ▾

Launch instances ▾

Search

< 1 >

⚙

<input type="checkbox"/>	Name ▾	Instance ID	Instance state ▾	Instance type ▾	Status check	Alarm status	Availability Zone ▾	Public IPv4 DNS ▾
<input type="checkbox"/>	-	i-0f6c1373777ce373e	<div><div>✔</div>Running</div> 🔍	t2.medium	<div><div>✔</div>2/2 checks passed</div>	No alarms +	us-east-1b	ec2-3-84-72-176.comp...

The data are downloaded and unzipped using the command line. To ensure it's reproducible easily, we put the commands in `workspace/src/ec2-data-download.sh`.

```
[root@ip-172-31-86-26 ~]# chmod 755 ec2-data-download.sh
[root@ip-172-31-86-26 ~]# ls -al
total 24
dr-xr-x--- 6 root root 174 Dec 13 00:31 .
dr-xr-xr-x 18 root root 257 Nov 22 18:50 ..
-rw-r--r-- 1 root root 18 Oct 18 2017 .bash_logout
-rw-r--r-- 1 root root 176 Oct 18 2017 .bash_profile
-rw-r--r-- 1 root root 176 Oct 18 2017 .bashrc
drwxr-xr-x 3 root root 17 Nov 22 18:55 .cache
-rw-r--r-- 1 root root 100 Oct 18 2017 .cshrc
-rwxr-xr-x 1 root root 235 Dec 13 00:31 ec2-data-download.sh
drwxr-xr-x 2 root root 24 Nov 24 23:56 .keras
drwxr-xr-x 2 root root 24 Nov 22 20:23 .safety
drwx----- 2 root root 29 Dec 13 00:22 .ssh
-rw-r--r-- 1 root root 129 Oct 18 2017 .tcshrc
[root@ip-172-31-86-26 ~]# ./ec2-data-download.sh
Downloading the data ...
--2021-12-13 00:32:37-- https://s3-us-west-1.amazonaws.com/udacity-aind/dog-project/dogImages.zip
Resolving s3-us-west-1.amazonaws.com (s3-us-west-1.amazonaws.com)... 52.219.116.16
Connecting to s3-us-west-1.amazonaws.com (s3-us-west-1.amazonaws.com)|52.219.116.16|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1132023110 (1.1G) [application/zip]
Saving to: 'dogImages.zip'

10% [=====>] 115,580,555 5.29MB/s eta 2m 44s

inflating: dogImages/valid/129.Tibetan_mastiff/Tibetan_mastiff_08185.jpg
creating: dogImages/valid/130.Welsh_springer_spaniel/
inflating: dogImages/valid/130.Welsh_springer_spaniel/Welsh_springer_spaniel_08201.jpg
inflating: dogImages/valid/130.Welsh_springer_spaniel/Welsh_springer_spaniel_08206.jpg
inflating: dogImages/valid/130.Welsh_springer_spaniel/Welsh_springer_spaniel_08222.jpg
inflating: dogImages/valid/130.Welsh_springer_spaniel/Welsh_springer_spaniel_08228.jpg
inflating: dogImages/valid/130.Welsh_springer_spaniel/Welsh_springer_spaniel_08235.jpg
inflating: dogImages/valid/130.Welsh_springer_spaniel/Welsh_springer_spaniel_08240.jpg
creating: dogImages/valid/131.Wirehaired_pointing_griffon/
inflating: dogImages/valid/131.Wirehaired_pointing_griffon/Wirehaired_pointing_griffon_08251.jpg
inflating: dogImages/valid/131.Wirehaired_pointing_griffon/Wirehaired_pointing_griffon_08263.jpg
inflating: dogImages/valid/131.Wirehaired_pointing_griffon/Wirehaired_pointing_griffon_08266.jpg
inflating: dogImages/valid/131.Wirehaired_pointing_griffon/Wirehaired_pointing_griffon_08279.jpg
creating: dogImages/valid/132.Xoloitzcuintli/
inflating: dogImages/valid/132.Xoloitzcuintli/Xoloitzcuintli_08298.jpg
inflating: dogImages/valid/132.Xoloitzcuintli/Xoloitzcuintli_08299.jpg
inflating: dogImages/valid/132.Xoloitzcuintli/Xoloitzcuintli_08301.jpg
inflating: dogImages/valid/132.Xoloitzcuintli/Xoloitzcuintli_08304.jpg
creating: dogImages/valid/133.Yorkshire_terrier/
inflating: dogImages/valid/133.Yorkshire_terrier/Yorkshire_terrier_08333.jpg
inflating: dogImages/valid/133.Yorkshire_terrier/Yorkshire_terrier_08334.jpg
inflating: dogImages/valid/133.Yorkshire_terrier/Yorkshire_terrier_08336.jpg
inflating: dogImages/valid/133.Yorkshire_terrier/Yorkshire_terrier_08348.jpg
Preparing the model directory ...
[root@ip-172-31-86-26 ~]# ls
dogImages dogImages.zip ec2-data-download.sh TrainedModels
[root@ip-172-31-86-26 ~]# ls -l
total 1105496
drwxr-xr-x 5 root root 44 Mar 27 2017 dogImages
-rw-r--r-- 1 root root 1132023110 Apr 1 2017 dogImages.zip
-rwxr-xr-x 1 root root 235 Dec 13 00:31 ec2-data-download.sh
drwxr-xr-x 2 root root 6 Dec 13 00:36 TrainedModels
```

We need to activate the right conda environment: `source activate pytorch_latest_p37` - otherwise, the default python environment doesn't have Pytorch installed.

CPU utilization (%)

Percent

79.6

39.8

0

23:50 23:55 00:00 00:05 00:10 00:15 00:20 00:25 00:30 00:35 00:40 00:45 00:50

i-0f6c1373777ce373e

1h 3h 12h 1d 3d 1w custom Line Actions

All metrics Graphed metrics (1) Graph options Source

Math expression Dynamic labels Statistic: Average Period: 5 Minutes Remove all

Label	Details	Statistic	Period	Y Axis	Actions
i-0f6c1373777ce373e	EC2 • CPUUtilization • InstanceId: i-0f6c1373777ce373e	Average	5 Minutes		

3. Lambda function setup

Amazon SageMaker > Endpoints

Endpoints

Update endpoint
Actions ▾
Create endpoint

	Name ▾	ARN	Creation time ▾	Status ▾	Last updated
	pytorch-inference-2021-12-13-01-05-29-889	arn:aws:sagemaker:us-east-1:183492708471:endpoint/pytorch-inference-2021-12-13-01-05-29-889	Dec 13, 2021 01:05 UTC	InService	Dec 13, 2021 01:08 UTC

4. Security and testing

To ensure the Lambda function in part #3 can hit the endpoint, we need to give it access to Sagemaker. Here, we assign **SagemakerFullAccess** to the IAM role that we use on the Lambda function.

Roles > dog-project-lambda-role-7zeu5ucn

Summary

Delete role

Role ARN	arn:aws:iam::183492708471:role/service-role/dog-project-lambda-role-7zeu5ucn
Role description	Edit
Instance Profile ARNs	
Path	/service-role/
Creation time	2021-12-13 09:31 UTC+1300
Last activity	2021-12-13 10:25 UTC+1300 (Today)
Maximum session duration	1 hour Edit

Permissions

Trust relationships

Tags

Access Advisor

Revoke sessions

Permissions policies (2 policies applied)

Attach policies

[Add inline policy](#)

Policy name	Policy type	
 AmazonSageMakerFullAccess	AWS managed policy	
AWSLambdaBasicExecutionRole-33392992-2764-448c-a7b6-474a2d3a3cb9	Managed policy	

Test event

Delete

Format

Save changes

Test

Invoke your function with a test event. Choose a template that matches the service that triggers your function, or enter your event document in JSON.

- ☐ New event
- ☒ Saved event
- Saved event

InputURL



```
1 {
2   "url": "https://s3.amazonaws.com/cdn-origin-etr.akc.org/wp-content/uploads/2017/11/20113314/Carolina-Dog-standing-outdoors.jpg"
3 }
```

lambda_function x

Execution result: x

+

Execution results

Status: Succeeded Max memory used: 65 MB Time: 921.15 ms

Test Event Name

inputURL

Response

```
{
  "statusCode": 200,
  "headers": {
    "Content-Type": "text/plain",
    "Access-Control-Allow-Origin": "*"
  },
  "type-result": "<class 'str'>",
  "Content-Type-In": "<_main_.LambdaContext object at 0x7f97e693ec70>",
  "body": "[[0.559682309627533, 0.3627522885799408, 0.18836575746536255, 0.3803918659687042, 0.6505036950111389, 0.4576735496520996, 0.21353593468666077, 0.4198894798755646, -0.1774848848581314, 0.]
}
```

Function Logs

```
START RequestId: d4dba3e7-1070-42fe-819b-54c5cc6dbe93 Version: $LATEST
Loading Lambda function
Context:: <_main_.LambdaContext object at 0x7f97e693ec70>
EventType:: <class 'dict'>
END RequestId: d4dba3e7-1070-42fe-819b-54c5cc6dbe93
REPORT RequestId: d4dba3e7-1070-42fe-819b-54c5cc6dbe93 Duration: 921.15 ms Billed Duration: 922 ms Memory Size: 128 MB Max Memory Used: 65 MB Init Duration: 347.10 ms
```

Request ID

d4dba3e7-1070-42fe-819b-54c5cc6dbe93

The following figure shows a preview of our IAM role dashboard. There aren't a lot of custom roles there, as this account is used specifically for this course. There are no sensitive information stored anywhere in this account (so it should be safe). However, since we assigned **SagemakerFullAccess** to the Lambda function, it allows any operations to the Sagemaker resources. Unfortunately, I don't find another role that allows invoking endpoint without giving too much control of the resources.

IAM > Roles

Roles (21) [Info](#)
An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.

< 1 2 > ⚙

<input type="checkbox"/>	Role name	Trusted entities	Last activity
<input type="checkbox"/>	AmazonSageMaker-ExecutionRole-20211212T122741	AWS Service: sagemaker	1 hour ago
<input type="checkbox"/>	AWSServiceRoleForAmazonSageMakerNotebooks	AWS Service: sagemaker (Service-Linked Role)	2 hours ago
<input type="checkbox"/>	dog-project-lambda-role-7zeu5ucn	AWS Service: lambda	3 hours ago
<input type="checkbox"/>	aws-ec2-spot-fleet-tagging-role	AWS Service: spotfleet	Yesterday
<input type="checkbox"/>	AWSServiceRoleForEC2SpotFleet	AWS Service: spotfleet (Service-Linked Role)	Yesterday
<input type="checkbox"/>	AWSServiceRoleForAWSCloud9	AWS Service: cloud9 (Service-Linked Role)	-
<input type="checkbox"/>	AWSServiceRoleForCloudWatchEvents	AWS Service: events (Service-Linked Role)	-
<input type="checkbox"/>	AWSServiceRoleForEC2Spot	AWS Service: spot (Service-Linked Role)	-
<input type="checkbox"/>	AWSServiceRoleForElasticCache	AWS Service: elasticache (Service-Linked Role)	-
<input type="checkbox"/>	AWSServiceRoleForGlobalAccelerator	AWS Service: globalaccelerator (Service-Linked Role)	-
<input type="checkbox"/>	AWSServiceRoleForOrganizations	AWS Service: organizations (Service-Linked Role)	-

5. Concurrency and autoscaling

Usually, Lambda function is used to bridge the application request to the endpoint. In this example, we define a reserved concurrency and set it to 10, i.e., the Lambda function can process 10 requests at the same time.

Lambda > Functions > dog-project-lambda > Edit concurrency

Edit concurrency

Concurrency
Unreserved account concurrency **990**
☐ Use unreserved account concurrency
☒ Reserve concurrency

Besides, we also configure autoscaling on our endpoint. We define the maximum instance to 2 (since it is just a course example), with the threshold of 5 invocations per instance, i.e., if there are more than 5 invocations, another instance will be spinned up. The scale-in and scale-out duration are set to 30 seconds to avoid downtime.

Amazon SageMaker > Endpoints > pytorch-inference-2021-12-13-01-05-29-889 > AllTraffic

Configure variant automatic scaling

Deregister auto scaling

Variant automatic scaling [Learn more](#)

Variant name AllTraffic	Instance type ml.m5.large Elastic Inference -	Current instance count 1 Current weight 1
----------------------------	--	--

Minimum instance count

1

-

Maximum instance count

2

IAM role

Amazon SageMaker uses the following service-linked role for automatic scaling. [Learn more](#)

AWSServiceRoleForApplicationAutoScaling_SageMakerEndpoint

Built-in scaling policy [Learn more](#)

Policy name

SageMakerEndpointInvocationScalingPolicy

Target metric

[SageMakerVariantInvocationsPerInstance](#)

Target value

5

Scale in cool down (seconds) - optional

30

Scale out cool down (seconds) - optional

30

☐ Disable scale in

9 / 9