# DEPLOY VERTEX AI WITH CUSTOM CODE

# STEP 1: Enable APIs

- Enable **Compute Engine API** (Need this to create notebook instance)
- Enable **Vertex AI API**
- Enable **Container Registry API** (Need this to create container for custom training job)

# STEP 2: Vertex AI Workbench Configuration

Create Vertex AI Workbench Instance

- Go to **Vertex AI** and select workbench instance
- Go to **USER-MANAGED NOTEBOOKS**
- Click **NEW NOTEBOOK** button and select **Pytorch 1.11** with **1 NVIDIA Tesla T4 GPU**

Vertex AI

Workbench

NEW NOTEBOOK | REFRESH | START | STOP | RESET | DELETE | SHOW INFO PANEL | LEARN

- Dashboard
- Datasets
- Features
- Labelling tasks
- Workbench
- Pipelines
- Training
- Experiments
- Models
- Endpoints
- Batch predictions
- Metadata
- Matching engine
- Manage resources
- Marketplace

MANAGED NOTEBOOKS

Notebooks have Jupyte
machine-learning frame

Filter Enter property

Customize...

Python 3
Includes scikit-learn, pandas and more

Python 3 (CUDA Toolkit 11.0)
Optimized for NVIDIA GPUs

TensorFlow Enterprise
Includes Keras, scikit-learn, pandas, NLTK and more

PyTorch 1.11
Includes scikit-learn, pandas, NLTK and more

- Without GPUs
- With 1 NVIDIA Tesla T4

R 4.1
Includes basic R packages, scikit-learn, pandas, NLTK and more

RAPIDS 0.18 [EXPERIMENTAL]
Optimized for NVIDIA GPUs

Kaggle Python [BETA]
Python image for Kaggle Notebooks, supporting hundreds of machine learning libraries popular on Kaggle

Theia IDE [EXPERIMENTAL]
IDE with notebook support including scikit-learn, pandas, and more

Smart Analytics Frameworks
BigQuery, Apache Beam, Apache Spark, Apache Hive and more

| | | Auto-upgrade | Environment | Machine type | GPUs | Owner | Last modified |
|---|---|---|---|---|---|---|---|
| | de | | ch:1.8 | 4 vCPUs, 15 GB RAM | NVIDIA Tesla K80 x 1 | Service account | 28 Apr 2022, 17:27:21 |
| | mu | | ch:1.11 | 4 vCPUs, 15 GB RAM | NVIDIA Tesla T4 x 1 | Service account | 29 Apr 2022, 00:06:00 |
| | pr | | ch:1.4 | e2-custom-4-21760 | None | Service account | 28 Apr 2022, 17:27:21 |
| | te | – | TensorFlow:2.3 | 4 vCPUs, 15 GB RAM | None | Service account | 9 May 2022, 04:54:34 |

- A popup will appear where you have to provide the **notebook name** and select the **region**.
- You can go to notebook properties and change multiple properties according to your requirements but for now, we will continue with default settings.
- **Enable the checkbox** (**install NVIDIA GPU driver automatically for me**) before creating the notebook.

Google Cloud Platform

Search Products, resources, docs (/)

Vertex AI

Workbench

NEW NOTEBOOK REFRESH START STOP RESET DELETE LEARN SHOW INFO PANEL

MANAGED NOTEBOOKS USER-MANAGED NOTEBOOKS EXECUTIONS SCHEDULES

Notebooks have JupyterLab 3 pre-installed and are configured with GPU-enabled machine-learning frameworks. Learn more

Filter Enter property name or value

- Dashboard
- Datasets
- Features
- Labelling tasks
- Workbench
- Pipelines
- Training
- Experiments
- Models
- Endpoints
- Batch predictions
- Metadata
- Matching engine
- Manage resources
- Marketplace

| Notebook name | | chine type | GPUs | Owner | Last modified |
|---|---|---|---|---|---|
| | OPEN | CPUs, 15 GB RAM | NVIDIA Tesla K80 x 1 | Service account | 28 Apr 2022, 17:27:21 |
| movie-classifier | OPEN | CPUs, 15 GB RAM | NVIDIA Tesla T4 x 1 | Service account | 29 Apr 2022, 00:06:00 |
| | | custom-4-21760 | None | Service account | 28 Apr 2022, 17:27:21 |
| | OPEN | CPUs, 15 GB RAM | None | Service account | 9 May 2022, 04:54:34 |

## New notebook

Notebook name *

pytorch-1-11-20220511-020438

63-character limit with lowercase letters, digits or '-' only. Must start with a letter. Cannot end with a '-'.

Region *: us-west1 (Oregon)

Zone *: us-west1-b

Restricted by GPU selection.

### Notebook properties

| | |
|---|---|
| Environment | PyTorch 1.11 (with Intel® MKL-DNN/MKL) |
| Machine type | 4 vCPUs, 15 GB RAM |
| GPUs | 1 NVIDIA Tesla T4 |
| Boot disk | 100 GB Standard persistent disk |
| Data disk | 100 GB Standard persistent disk |
| Subnetwork | default(10.138.0.0/20) |
| External IP | Ephemeral(Automatic) |
| Permission | Compute Engine default service account |
| Estimated cost | US$281.58 monthly, $0.386 hourly |

Install NVIDIA GPU driver automatically for me

NVIDIA GPU drivers must be installed for the GPUs to work.

ADVANCED OPTIONS CANCEL CREATE

- Once the notebook is created, A button **(OPEN JUPYTERLAB)** will appear at the right side of notebook name
- Click on it and the notebook will open in a new tab.

Google Cloud Platform

Search Products, resources, docs (/)

Vertex AI

- Dashboard
- Datasets
- Features
- Labelling tasks
- Workbench
- Pipelines
- Training
- Experiments
- Models
- Endpoints
- Batch predictions
- Metadata
- Matching engine

Manage resources

Marketplace

# Workbench

NEW NOTEBOOK | REFRESH | START | STOP | RESET | DELETE | LEARN | SHOW INFO PANEL

MANAGED NOTEBOOKS | USER-MANAGED NOTEBOOKS | EXECUTIONS | SCHEDULES

Notebooks have JupyterLab 3 pre-installed and are configured with GPU-enabled machine-learning frameworks. Learn more

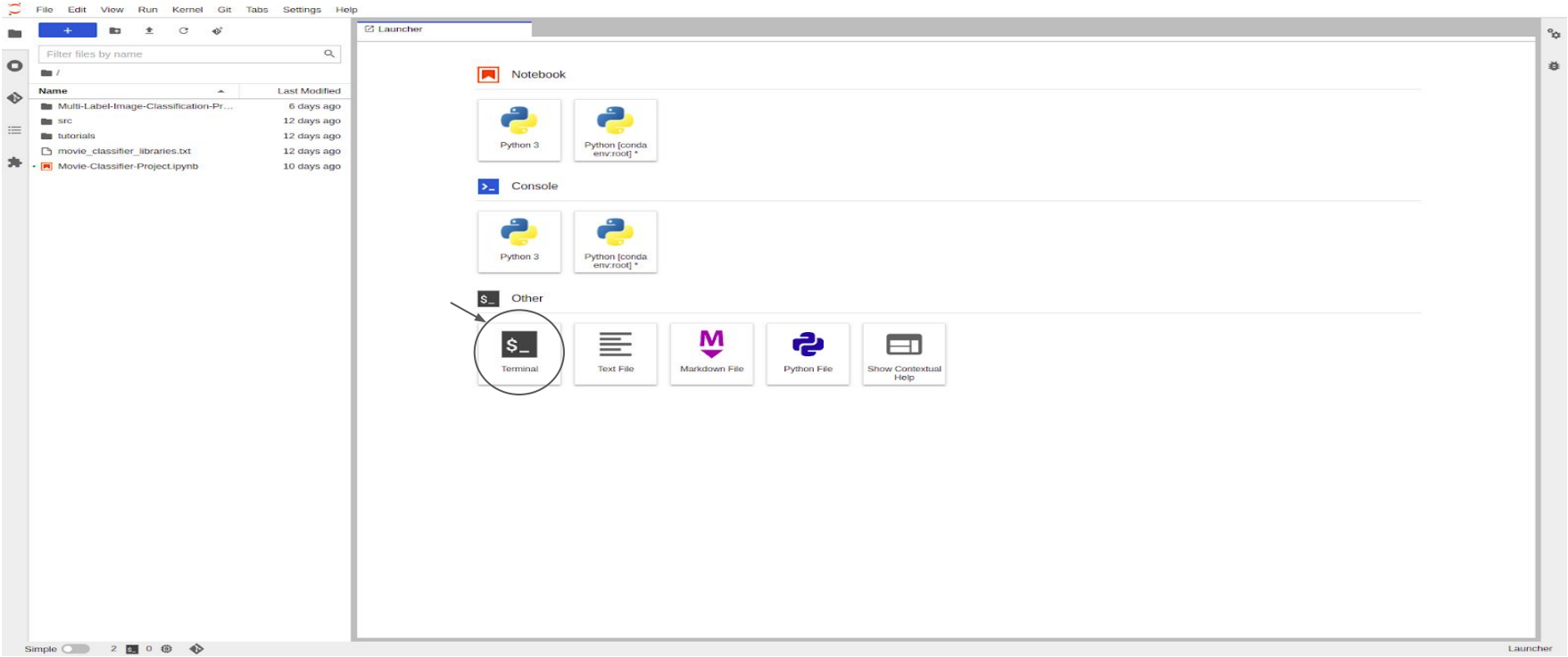
Filter Enter property name or value

| Notebook name | | Zone | Auto-upgrade | Environment | Machine type | GPUs | Owner | Last modified |
|---|---|---|---|---|---|---|---|---|
| | OPEN JUPYTERLAB | us-central1-a | – | PyTorch:1.8 | 4 vCPUs, 15 GB RAM | NVIDIA Tesla K80 x 1 | Service account | 28 Apr 2022, 17:27:21 |
| movie-classifier | OPEN JUPYTERLAB | us-central1-a | – | PyTorch:1.11 | 4 vCPUs, 15 GB RAM | NVIDIA Tesla T4 x 1 | Service account | 29 Apr 2022, 00:06:00 |
| | | us-central1-a | – | PyTorch:1.4 | e2-custom-4-21760 | None | Service account | 28 Apr 2022, 17:27:21 |
| | OPEN JUPYTERLAB | us-west1-b | – | TensorFlow:2.3 | 4 vCPUs, 15 GB RAM | None | Service account | 9 May 2022, 04:54:34 |

https://51235b93f9451150-dot-us-central1.notebooks.googleusercontent.com/?authuser=0&username=Syed_Ali_Naqi

# STEP 3: Move your code to Vertex AI

- Select **Terminal** under Other's tab.

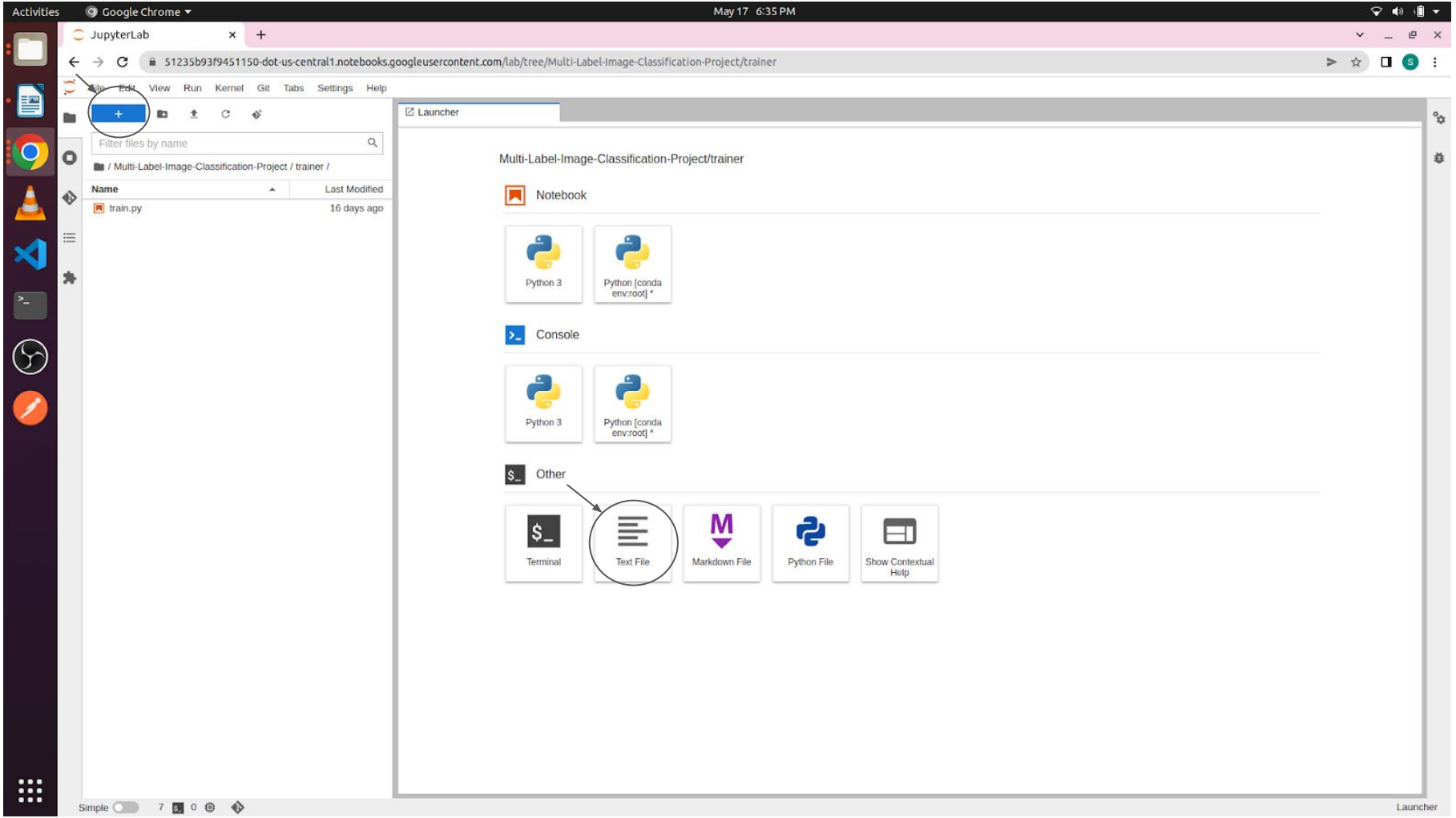


- [illegible] new directory and cd into it
- [illegible] **Image-Classification-Project**
- Now go to terminal and create two new variables with the URI of your container image, i.e.,
  **IMAGE_URI_FOR_TRAINING =** “gcr.io/$PROJECT_ID/movie-classifier-training_image:v1”
  **IMAGE_URI_FOR_DEPLOYMENT =** “gcr.io/$PROJECT_ID/movie-classifier-deployment_image:v1”
- Replace $PROJECT_ID with your Project ID

# STEP 4: Create Docker files

- In order to create Training Docker file, tap on **New Launcher button (+)** and create a text file (under Other's tab).
- Rename it from untitled.txt to **DockerfileForTraining** and put the following code inside the file:

```
FROM gcr.io/cloud-aiplatform/training/pytorch-gpu.1-7

WORKDIR /

COPY ./movie_classifier_libraries.txt ./

RUN pip install -r movie_classifier_libraries.txt

RUN mkdir model

RUN mkdir images

COPY . .

# Sets up the entry point to invoke the trainer.

ENTRYPOINT ["python", "-m", "train"]
```


Activities
Google Chrome
May 17 6:35 PM
JupyterLab
51235b93f9451150-dot-us-central1.notebooks.googleusercontent.com/lab/tree/Multi-Label-Image-Classification-Project/trainer
File Edit View Run Kernel Git Tabs Settings Help
Filter files by name
/ Multi-Label-Image-Classification-Project / trainer /
Name
Last Modified
train.py
16 days ago
Launcher
Multi-Label-Image-Classification-Project/trainer
Notebook
Python 3
Python [conda env:root] *
Console
Python 3
Python [conda env:root] *
Other
Terminal
Text File
Markdown File
Python File
Show Contextual Help
Simple
Launcher

- Similarly create the Docker file for deployment, rename it to **DockerfileForDeployment** and put following code inside it:

```
FROM python:3.7

WORKDIR /home/model-server/

COPY ./movie_classifier_libraries_deployment.txt /home/model-server/

RUN pip install -r movie_classifier_libraries_deployment.txt

RUN apt-get update

RUN apt-get install ffmpeg libsm6 libxext6  -y

ADD . /home/model-server/

RUN mkdir model

RUN python download_model_and_test_df.py

CMD exec gunicorn -b :5000 --max-requests 1 --graceful-timeout 300 -t 600 main:app
```

# STEP 5: Dockerize your Code

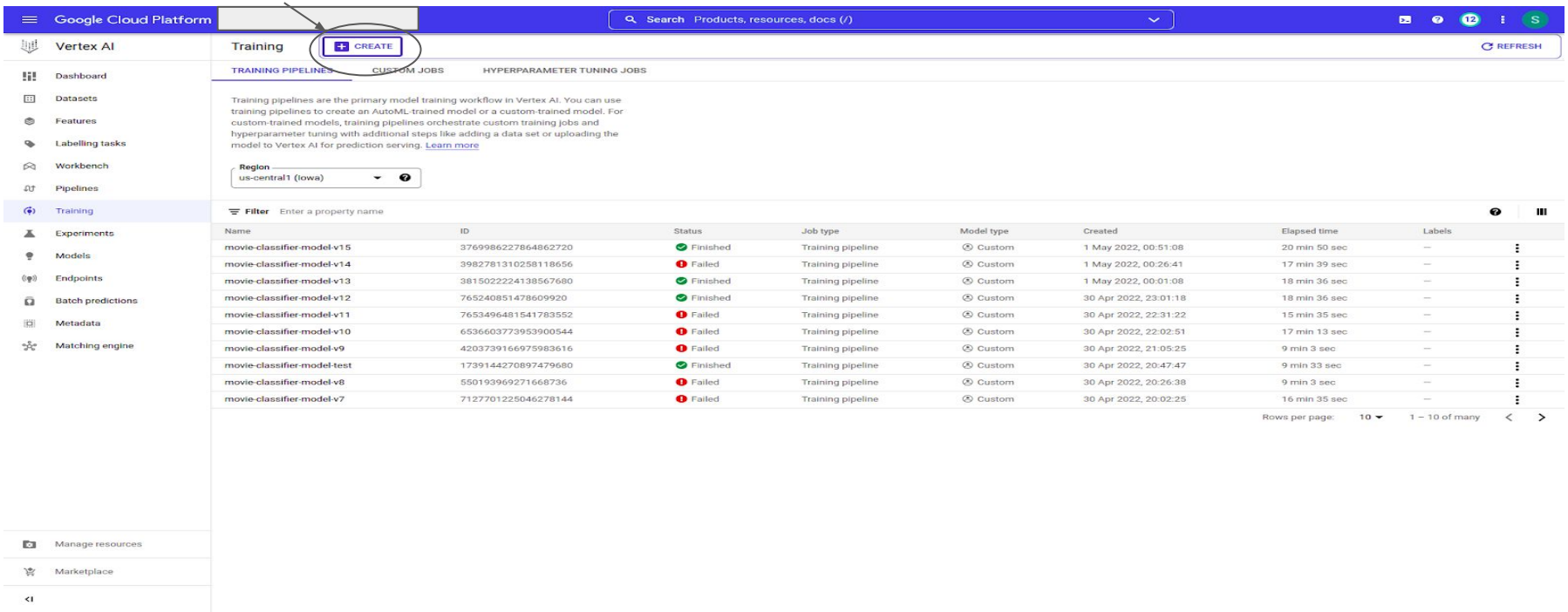
- Now that we have docker files, we can create docker images and push them to container registries (Azure Container Registry, ECR, Google Container Registry, etc). Once we have the image pushed to a container registry, we will be ready to kick off a customer model training job.

# STEP 6: Cloud Storage Bucket Creation

- Before starting the training job, you need to create a **Cloud Storage Bucket** and move your training data into it.
- You also need to create a Cloud Storage Bucket to store your trained model. Vertex will later use this bucket to read the exported model assets and deploy the model.
- Please make sure that the region selected while creating the model bucket should be same where you deploy model endpoint later.
- Run the following command in your **Workbench Terminal** to create a new bucket in your project:
- **PROJECT_ID=”your-cloud-project-id”**
- **BUCKET_NAME=”gs://${PROJECT_ID}-bucket”**
- **gsutil mb -l us-central1 $BUCKET_NAME**

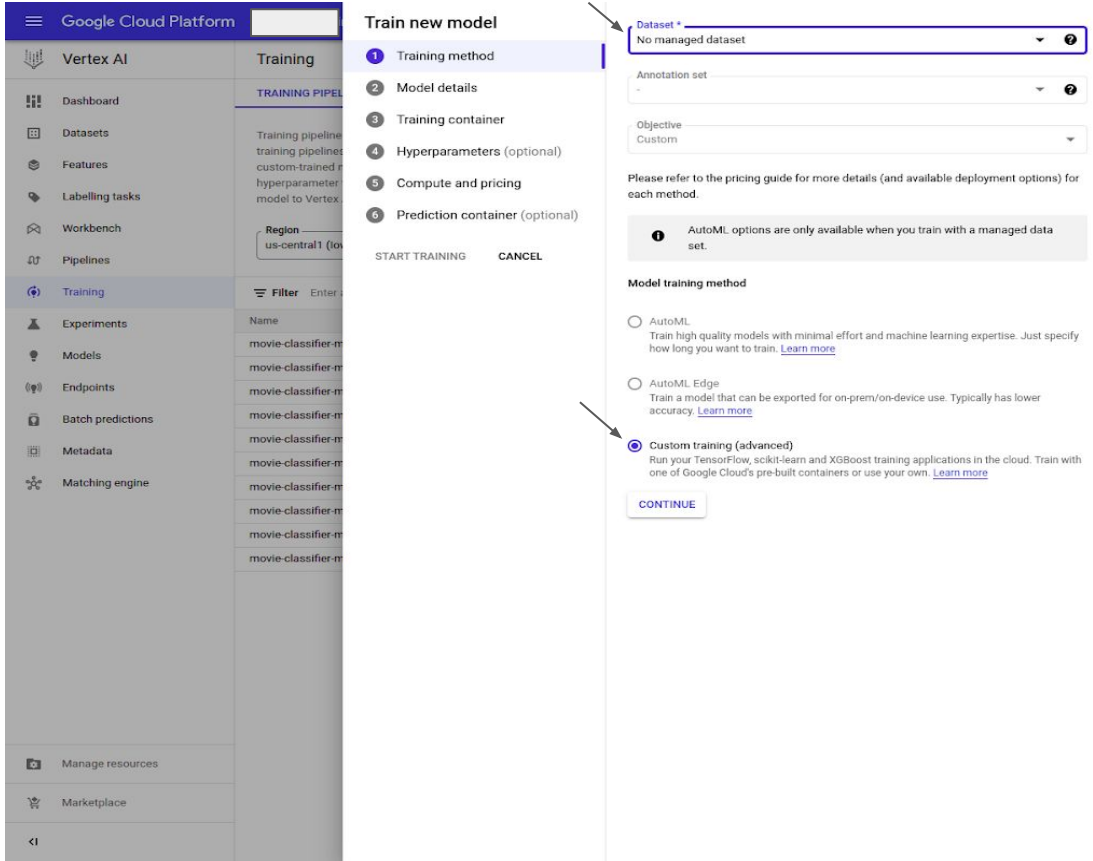
# STEP 7: Training Process and Configurations

Navigate to the **TRAINING** section in the Vertex AI of Cloud console and click **CREATE** button to enter the parameters for the training job.

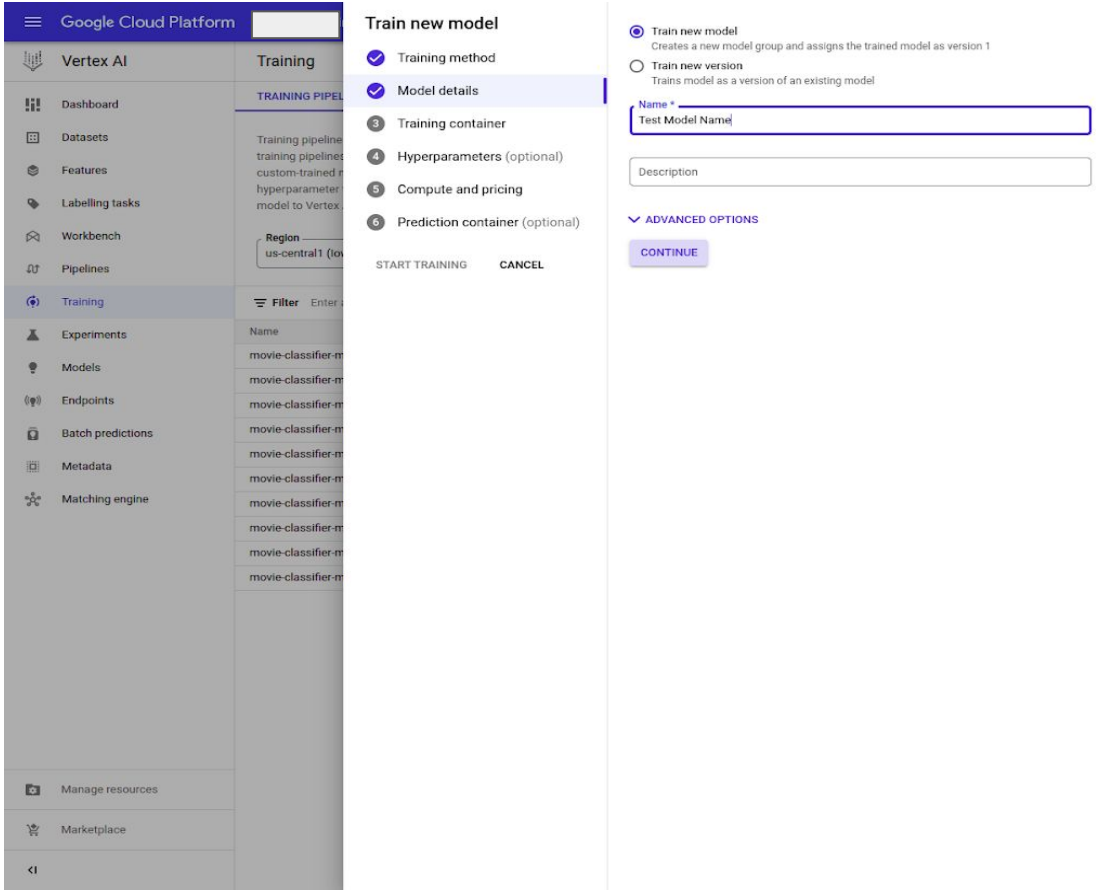



| Name | ID | Status | Job type | Model type | Created | Elapsed time | Labels |
|---|---|---|---|---|---|---|---|
| movie-classifier-model-v15 | 3769986227864862720 | Finished | Training pipeline | Custom | 1 May 2022, 00:51:08 | 20 min 50 sec | – |
| movie-classifier-model-v14 | 3982781310258118656 | Failed | Training pipeline | Custom | 1 May 2022, 00:26:41 | 17 min 39 sec | – |
| movie-classifier-model-v13 | 3815022224138567680 | Finished | Training pipeline | Custom | 1 May 2022, 00:01:08 | 18 min 36 sec | – |
| movie-classifier-model-v12 | 7652408514786099920 | Finished | Training pipeline | Custom | 30 Apr 2022, 23:01:18 | 18 min 36 sec | – |
| movie-classifier-model-v11 | 7653496481541783552 | Failed | Training pipeline | Custom | 30 Apr 2022, 22:31:22 | 15 min 35 sec | – |
| movie-classifier-model-v10 | 6536603773953900544 | Failed | Training pipeline | Custom | 30 Apr 2022, 22:02:51 | 17 min 13 sec | – |
| movie-classifier-model-v9 | 4203739166975983616 | Failed | Training pipeline | Custom | 30 Apr 2022, 21:05:25 | 9 min 3 sec | – |
| movie-classifier-model-test | 1739144270897479680 | Finished | Training pipeline | Custom | 30 Apr 2022, 20:47:47 | 9 min 33 sec | – |
| movie-classifier-model-v8 | 5501939692716687 36 | Failed | Training pipeline | Custom | 30 Apr 2022, 20:26:38 | 9 min 3 sec | – |
| movie-classifier-model-v7 | 7127701225046278144 | Failed | Training pipeline | Custom | 30 Apr 2022, 20:02:25 | 16 min 35 sec | – |

- Under **Dataset**, select **No managed dataset**, under **Model training method** select **Custom training (advanced)** as your training method and click **Continue**.



- In the next step, enter **Model name** (mandatory) and provide description (optional) and click **Continue**.



- Now select **Custom container** option and browse your container image (that you pushed earlier) from **Container Registry**.
- Also provide your model bucket directory where your model will be saved and click **Continue**.
- For now, we are not doing hyperparameter tuning so leave the **Enable hyperparameter tuning** checkbox unchecked and click **Continue**.

Google Cloud Platform

Vertex AI

Dashboard
Datasets
Features
Labelling tasks
Workbench
Pipelines
Training
Experiments
Models
Endpoints
Batch predictions
Metadata
Matching engine

Manage resources
Marketplace

Training

TRAINING PIPEL

Training pipeline
training pipelines
custom-trained m
hyperparameter
model to Vertex

Region
us-central1 (lo

Filter Enter

Name
movie-classifier-m
movie-classifier-m
movie-classifier-m
movie-classifier-m
movie-classifier-m
movie-classifier-m
movie-classifier-m
movie-classifier-m
movie-classifier-m
movie-classifier-m

## Train new model

1. Training method
2. Model details
3. Training container
4. Hyperparameters (optional)
5. Compute and pricing
6. Prediction container (optional)

START TRAINING CANCEL

Select a pre-built container or build a custom container using ML frameworks (as well as non-ML dependencies, libraries and binaries) that are not otherwise supported. Learn more

- Pre-built container
  View the list of supported runtimes including TensorFlow and scikit-learn versions
- Custom container
  Build a custom Docker container. Must be stored in Container Registry

### Custom container settings

Container image *
gcr.io/[redacted]/movie-genre-classifier@sha256:37e486dab0f1b1c BROWSE

gs:// Model output directory BROWSE

Your model artifacts and other data needed for training will be stored on Cloud Storage. You should specify a path here if you do not set an output directory in your application code or arguments.

### Arguments

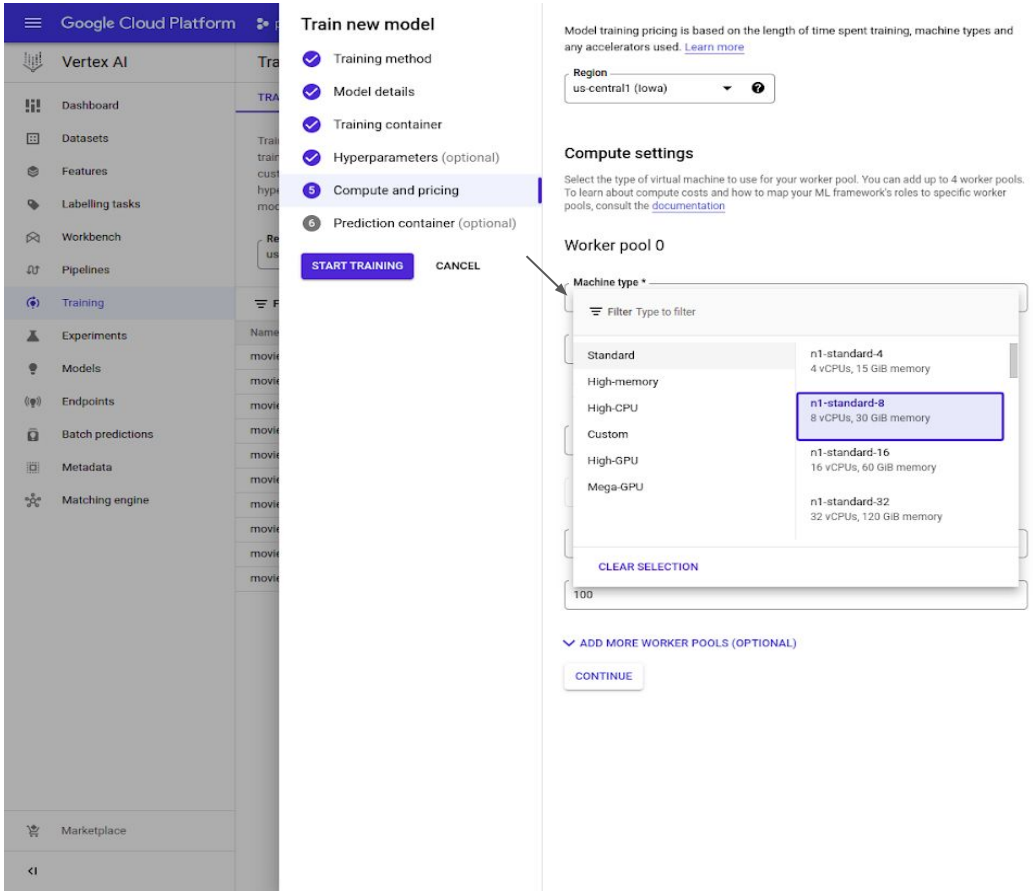
Optional. Add arguments for the command that runs when the container starts. Overrides the container's CMD instruction. Enter one parameter and its argument per line.

```
--flag_a=xxxx
-flag2
flag3
```

For parameters you want to tune with HyperTune, enter arguments of the hyperparameters you defined in the training code in the hyperTune setting below. If none, click Next to skip this step.

CONTINUE

- training. Step 5: Select the **Accelerator Type**. Accelerator Count and click **Continue**. Step 4: **Machine Type**: Choose your machine type and click **Continue**. Step 3: Set the **Region** to your choice. Then you will be ready to train your model



Google Cloud Platform

Vertex AI

- Dashboard
- Datasets
- Features
- Labelling tasks
- Workbench
- Pipelines
- Training
- Experiments
- Models
- Endpoints
- Batch predictions
- Metadata
- Matching engine
- Marketplace

## Train new model

- Training method
- Model details
- Training container
- Hyperparameters (optional)
- 5 Compute and pricing
- 6 Prediction container (optional)

START TRAINING CANCEL

Model training pricing is based on the length of time spent training, machine types and any accelerators used. Learn more

Region
us-central1 (Iowa)

### Compute settings

Select the type of virtual machine to use for your worker pool. You can add up to 4 worker pools. To learn about compute costs and how to map your ML framework's roles to specific worker pools, consult the documentation

### Worker pool 0

Machine type *
n1-standard-8, 8 vCPUs, 30 GiB memory

Accelerator type

- NVIDIA_TESLA_K80
- NVIDIA_TESLA_P100
- NVIDIA_TESLA_V100
- NVIDIA_TESLA_P4
- NVIDIA_TESLA_T4

Disk type
SSD

Disk size (GB)
100

ADD MORE WORKER POOLS (OPTIONAL)

CONTINUE

Google Cloud Platform

Vertex AI

- Dashboard
- Datasets
- Features
- Labelling tasks
- Workbench
- Pipelines
- Training
- Experiments
- Models
- Endpoints
- Batch predictions
- Metadata
- Matching engine
- Marketplace

## Train new model

- Training method
- Model details
- Training container
- Hyperparameters (optional)
- 5 Compute and pricing
- 6 Prediction container (optional)

START TRAINING CANCEL

Model training pricing is based on the length of time spent training, machine types and any accelerators used. Learn more

Region
us-central1 (Iowa)

### Compute settings

Select the type of virtual machine to use for your worker pool. You can add up to 4 worker pools. To learn about compute costs and how to map your ML framework's roles to specific worker pools, consult the documentation

### Worker pool 0

Machine type *
n1-standard-8, 8 vCPUs, 30 GiB memory

Accelerator type
NVIDIA_TESLA_K80

Accelerators can speed up model training that involves intensive compute tasks. Learn more

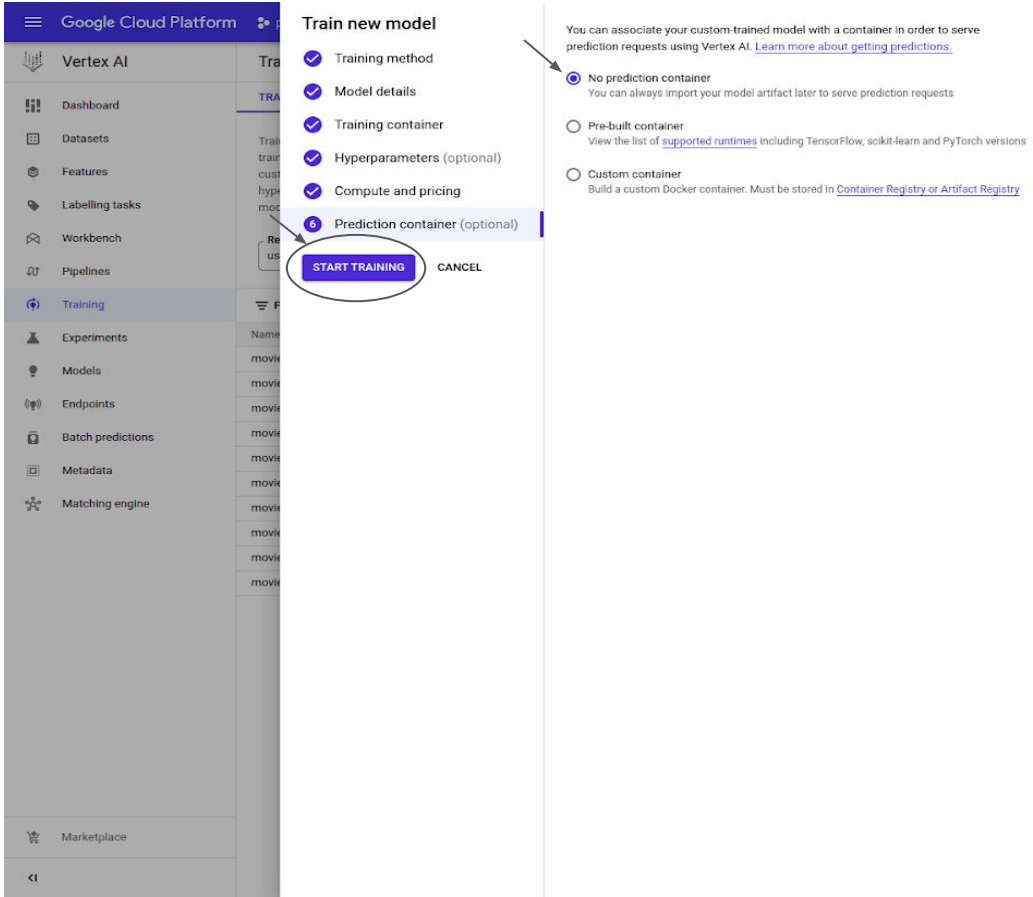
Accelerator count
1
2
4
8

Disk size (GB)
100

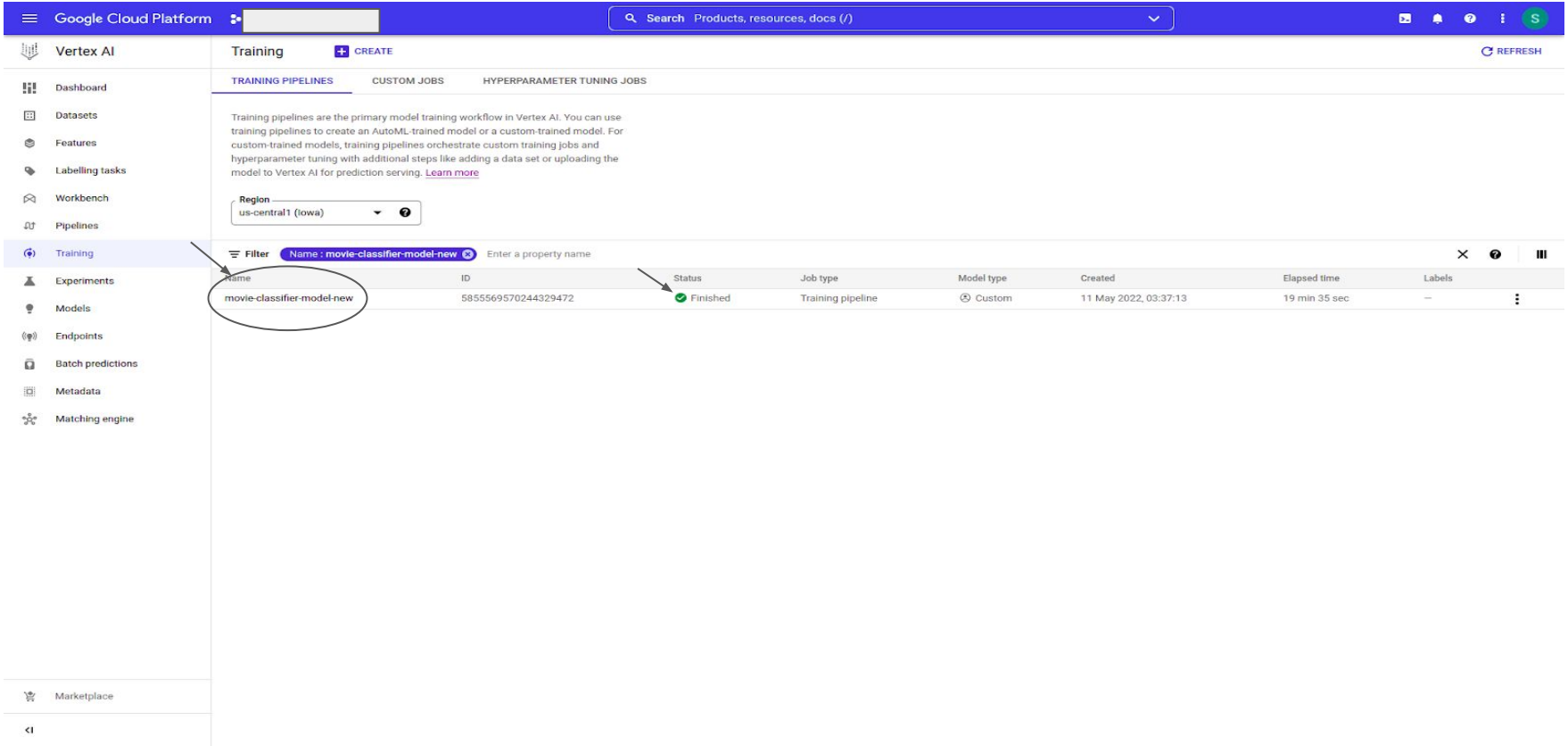
ADD MORE WORKER POOLS (OPTIONAL)

CONTINUE

- Leave the **No prediction container** option selected and hit **START TRAINING** button to kick off the training job.

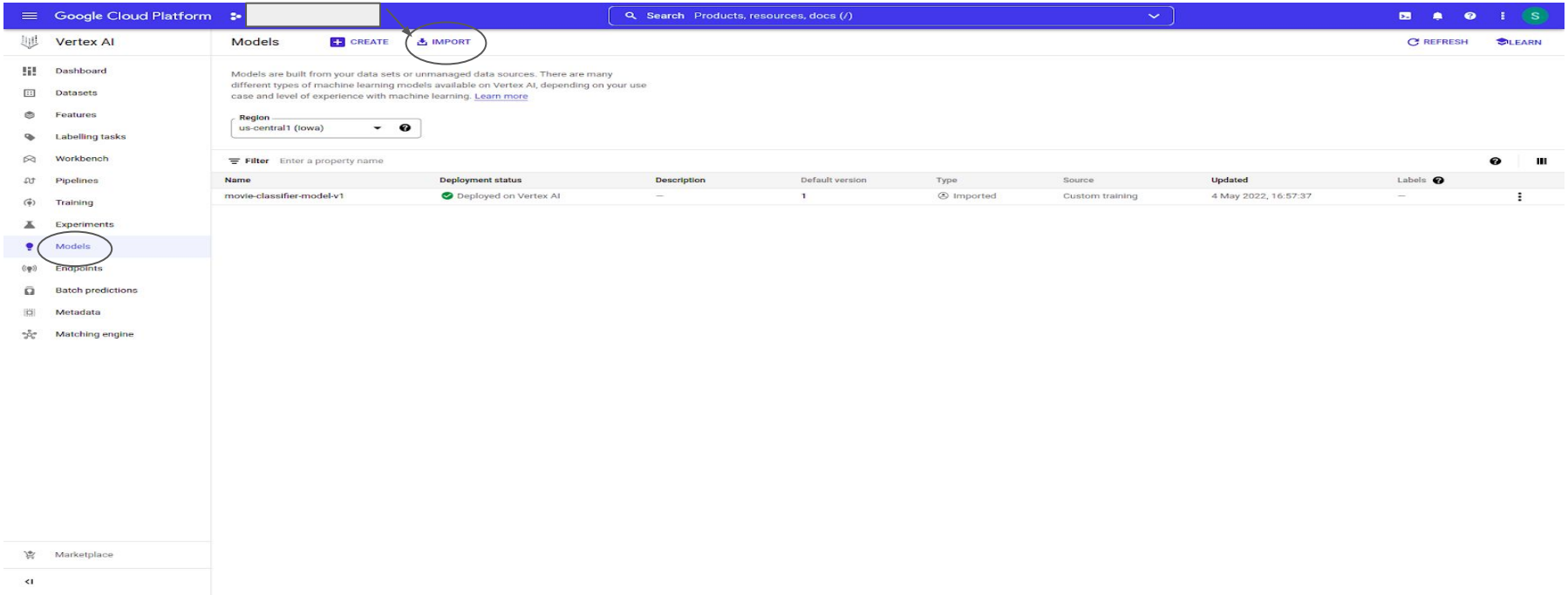


- The training will take around **15-20 minutes**. Once the training job changes from **Training** to **Finished** and the model saving is complete, you can deploy your trained model on Vertex AI.

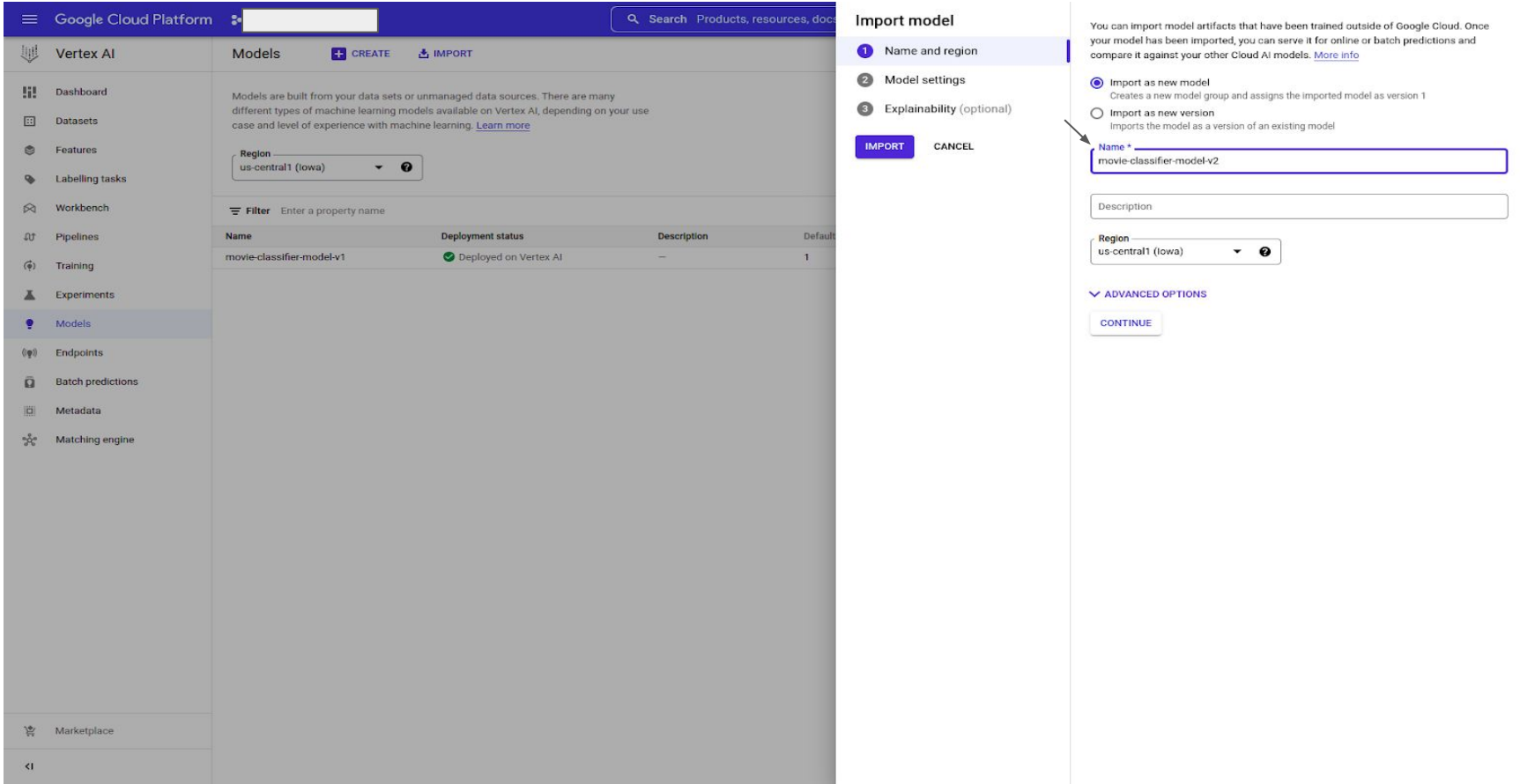


# STEP 8: Model Deployment

- To deploy the model on Vertex AI, go to the **MODEL** section of Vertex AI and click **IMPORT** button.



- Enter **Model name** (mandatory) and provide **Description** (optional) and click **Continue**.



- Next select **Import an existing custom container** option and under **Custom Container** field provide the following routes in **Prediction route** and **Health route** text field:

  **Prediction route=**/get_movie_genres

  **Health route=**/health

  (We have specified these routes in our **inference.py file** and we will use them when we invoke our model endpoint)

- Also replace port numbers from **8080** to **5000** in **Command** text field.
- Under **Explainability options**, leave the **None** option selected and hit **IMPORT** button.

Google Cloud Platform

Search Products, resources, docs

Vertex AI

- Dashboard
- Datasets
- Features
- Labelling tasks
- Workbench
- Pipelines
- Training
- Experiments
- Models
- Endpoints
- Batch predictions
- Metadata
- Matching engine
- Marketplace

## Models

CREATE IMPORT

Models are built from your data sets or unmanaged data sources. There are many different types of machine learning models available on Vertex AI, depending on your use case and level of experience with machine learning. Learn more

Region: us-central1 (Iowa)

Filter Enter a property name

| Name | Deployment status | Description | Default |
|---|---|---|---|
| movie-classifier-model-v1 | Deployed on Vertex AI | — | 1 |

## Import model

1. Name and region
2. Model settings
3. Explainability (optional)

IMPORT CANCEL

- Import model artifacts into a new pre-built container
  View the list of supported runtimes including TensorFlow, scikit-learn and XGBoost versions
- Import an existing custom container
  Build a custom Docker container. Must be stored in Container Registry or Artifact Registry

### Custom container settings

Container image
gcr.io/pronto-project/movie-classifier-image@sha256:ace98e9b96604e BROWSE

The URI for the Docker image in Container Registry or Artifact Registry. You must have permission to access the image. Learn more

Command

Model artifact location (Cloud storage path)
gs:// movie-classifier-model-bucket BROWSE

Path to the Cloud Storage directory where the exported model file is stored (not the path to the model file itself).

### Arguments

Optional. Add arguments for the command that runs when the container starts. Overrides the container's CMD instruction. Enter one parameter and its argument per line.

```
--flag_a=xxxx
-flag2
flag3
```

### Environment variables

Optional. Specifies the command that runs when the container starts. Overrides the container's ENTRYPOINT instructions. Enter one variable per line, separating the key and value with an equal (=) sign.

```
ENV_VAR_1=MY_ENV_VALUE
ENV_VAR_2=MY_SECOND_ENV_VALUE
```

Prediction route

An HTTP path to send prediction requests to. Vertex AI will forward requests to the entered path on the container's IP address and port. If not set, defaults to a value set when the model is deployed to an endpoint. Learn more about predict routes.

Google Cloud Platform

Search Products, resources, docs

Vertex AI

- Dashboard
- Datasets
- Features
- Labelling tasks
- Workbench
- Pipelines
- Training
- Experiments
- Models
- Endpoints
- Batch predictions
- Metadata
- Matching engine
- Marketplace

## Models

CREATE IMPORT

Models are built from your data sets or unmanaged data sources. There are many different types of machine learning models available on Vertex AI, depending on your use case and level of experience with machine learning. Learn more

Region: us-central1 (Iowa)

Filter Enter a property name

| Name | Deployment status | Description | Default |
|---|---|---|---|
| movie-classifier-model-v1 | Deployed on Vertex AI | — | 1 |

## Import model

1. Name and region
2. Model settings
3. Explainability (optional)

IMPORT CANCEL

### Environment variables

Optional. Specifies the command that runs when the container starts. Overrides the container's ENTRYPOINT instructions. Enter one variable per line, separating the key and value with an equal (=) sign.

```
ENV_VAR_1=MY_ENV_VALUE
ENV_VAR_2=MY_SECOND_ENV_VALUE
```

Prediction route: /get_movie_genres

An HTTP path to send prediction requests to. Vertex AI will forward requests to the entered path on the container's IP address and port. If not set, defaults to a value set when the model is deployed to an endpoint. Learn more about predict routes.

Health route: /health

An HTTP path to send health checks to. Vertex AI occasionally sends GET requests to this path on the container's IP address and port to check that the container is healthy. If not set, defaults to a value set when the model is deployed to an endpoint. Learn more about health routes.

Port: 5000

Port to expose from the container. Prediction requests and health checks will be sent to the port. If left blank, the default port is 8080.

### Predict schemata

Optional. Learn more about the predict schemata

gs:// Instances BROWSE

Cloud Storage location to a YAML file that defines the format of a single instance used in prediction and explanation requests.

gs:// Parameters BROWSE

Cloud Storage location to a YAML file that defines the prediction and explanation parameters.

gs:// Predictions BROWSE

Cloud Storage location to a YAML file that defines the format of a single prediction or explanation.

CONTINUE

Google Cloud Platform

Search Products, resources, docs

Vertex AI

- Dashboard
- Datasets
- Features
- Labelling tasks
- Workbench
- Pipelines
- Training
- Experiments
- Models
- Endpoints
- Batch predictions
- Metadata
- Matching engine
- Marketplace

## Models

CREATE IMPORT

Models are built from your data sets or unmanaged data sources. There are many different types of machine learning models available on Vertex AI, depending on your use case and level of experience with machine learning. Learn more

Region: us-central1 (Iowa)

Filter Enter a property name

| Name | Deployment status | Description | Default |
|---|---|---|---|
| movie-classifier-model-v1 | Deployed on Vertex AI | – | 1 |

## Import model

- Name and region
- Model settings
- 3 Explainability (optional)

IMPORT CANCEL

## Explainability options

LEARN

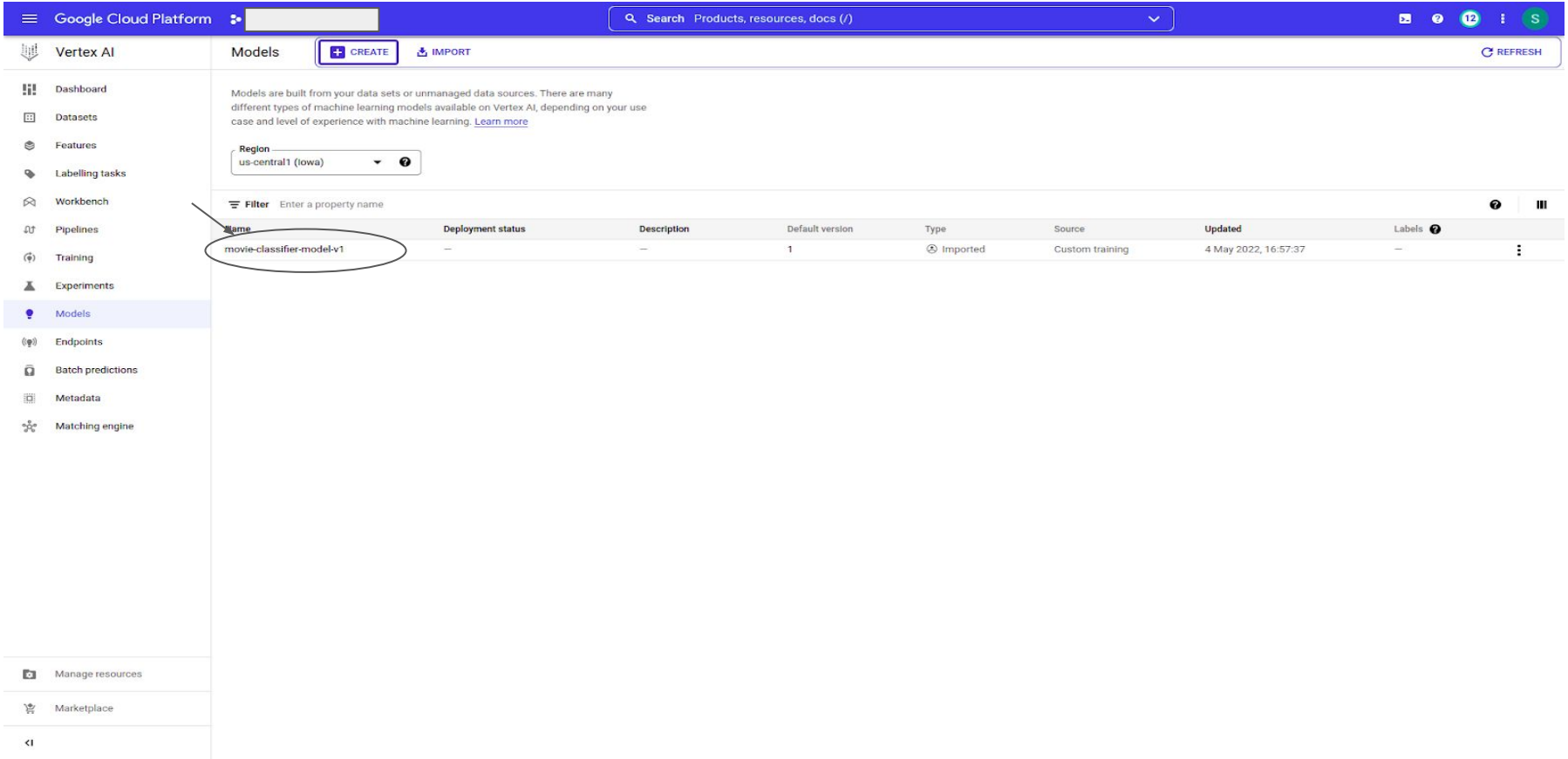
In Vertex AI, models are made explainable through feature attribution, which tells you how much each feature contributed to the predicted result. You can use this information to verify that the model is behaving as expected, recognise bias in your models and get ideas for ways to improve your model and your training data. Explainability will incur a minor additional cost. Learn more

### Select a feature attribution method

Your model's data type determines which attribution methods are available to use. Learn more about attribution methods

- None
- Sampled Shapley (for tabular models)
- Integrated gradients (for tabular models)
- Integrated gradients (for image classification models)
- XRAI (for image classification models)

- After few seconds, the model will appear in **MODEL** section of Vertex AI and now you can deploy this model to an endpoint.



# STEP 9: Endpoint Deployment

- When your training job ran, Vertex created a model resource for you. In order to use this model, you need to deploy a endpoint. You can have many endpoints per model.
- Go to the **ENDPOINT** section of Vertex AI and then click on **CREATE AN ENDPOINT** button.
- Give the **Endpoint name** like “movie-classifier-endpoint” and hit the **Continue** button**.**

Google Cloud Platform

Search Products, resources, docs

Vertex AI

- Dashboard
- Datasets
- Features
- Labelling tasks
- Workbench
- Pipelines
- Training
- Experiments
- Models
- Endpoints
- Batch predictions
- Metadata
- Matching engine

Manage resources

Marketplace

## Endpoints

CREATE ENDPOINT

Endpoints are machine learning models made available for online prediction requests. Endpoints are useful for timely predictions from many users (for example, in response to an application request). You can also request batch predictions if you don't need immediate results.

To create an endpoint, you need at least one machine-learning model. Learn more

Region: us-central1 (Iowa)

Filter Enter a property name

| Name | ID | Status | Models |
|---|---|---|---|
| | | | 0 |
| | | | 0 |
| | | | 0 |
| | | | 0 |
| | | | 0 |
| | | | 0 |
| | | | 0 |
| | | | 0 |
| | | | 0 |
| | | | 0 |
| | | | 0 |
| | | | 0 |
| | | | 0 |

## New endpoint

1. Define your endpoint
2. Model settings

CREATE CANCEL

Endpoint name *: movie-classifier-endpoint

### Location

Region: us-central1 (Iowa)

### Access

Determines how your endpoint can be accessed. By default, endpoints are available for prediction serving through a REST API. Endpoint access can't be changed after the endpoint is created.

- (●) Standard
  Makes the endpoint available for prediction serving through a REST API. AutoML and custom-trained models can be added to standard endpoints.
- ( ) Private
  Create a private connection to this endpoint using a VPC network and private services access. Only custom-trained and tabular models can be added to private endpoints. Learn more

ADVANCED OPTIONS

CONTINUE

- Under **Model settings**, there is a model name drop down where all your created model list appears.
- Select the model that you created in the previous step and select its version. Once you do that, some more options will appear.
- Leave **Traffic split** at 100 and enter 1 for **Minimum number of compute nodes**.
- Under **Machine type**, select **n1-standard-2** (or any machine type you'd like).
- From **Service account** drop-down, select a service account that has all the permissions that your code can access.
- Leave the rest of the defaults selected and then click **Continue**.

Google Cloud Platform

Search Products, resources, docs

Vertex AI

- Dashboard
- Datasets
- Features
- Labelling tasks
- Workbench
- Pipelines
- Training
- Experiments
- Models
- Endpoints
- Batch predictions
- Metadata
- Matching engine

Manage resources

Marketplace

## Endpoints

CREATE ENDPOINT

Endpoints are machine learning models made available for online prediction requests. Endpoints are useful for timely predictions from many users (for example, in response to an application request). You can also request batch predictions if you don't need immediate results.

To create an endpoint, you need at least one machine-learning model. Learn more

Region: us-central1 (Iowa)

Filter: Enter a property name

| Name | ID | Status | Models |
|---|---|---|---|
| | | | 0 |
| | | | 0 |
| | | | 0 |
| | | | 0 |
| | | | 0 |
| | | | 0 |
| | | | 0 |
| | | | 0 |
| | | | 0 |
| | | | 0 |
| | | | 0 |
| | | | 0 |
| | | | 0 |

## New endpoint

1. Define your endpoint
2. Model settings
3. Model monitoring

CREATE CANCEL

### Add model

Model name *: movie-classifier-model-v1

Version: Version 1

Traffic split *: 100 %

### Compute resources

Choose how compute resources will serve prediction traffic to your model

- **Autoscaling**: If you set a minimum and maximum, compute nodes will scale to meet traffic demand within those boundaries
- **No scaling**: If you only set a minimum, then that number of compute nodes will always run regardless of traffic demand (the maximum will be set to minimum)

Once scaling settings are set, they can't be changed unless you redeploy the model. Pricing guide

Minimum number of compute nodes *: 1

Default is 1. If set to 1 or more, then compute resources will continuously run even without traffic demand. This can increase cost but avoid dropped requests due to node initialisation.

Maximum number of compute nodes (optional)

Enter a number equal to or greater than the minimum nodes. Can reduce costs but may cause reliability issues for high traffic.

ADVANCED SCALING OPTIONS

Machine type *: n1-standard-2, 2 vCPUs, 7.5 GiB memory

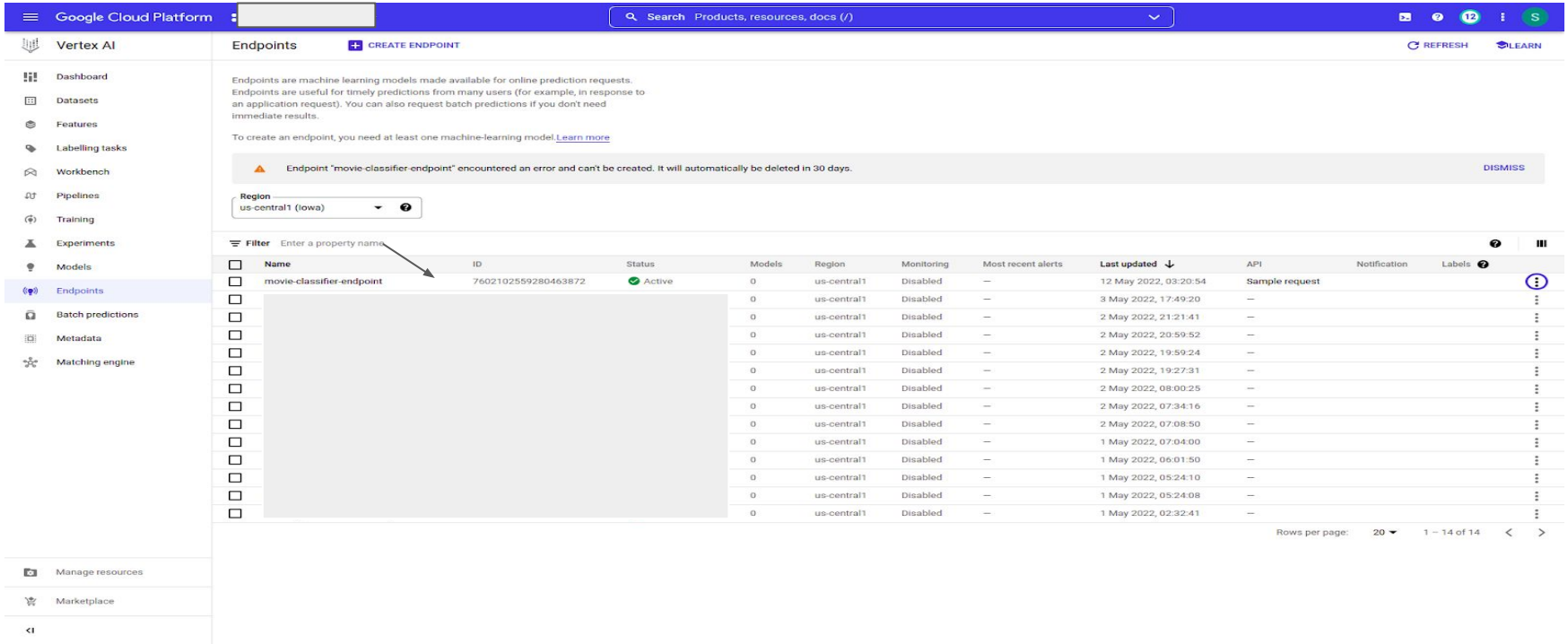
Accelerator type: NVIDIA_TESLA_K80

Accelerator count: 1

Service account: App Engine default service account

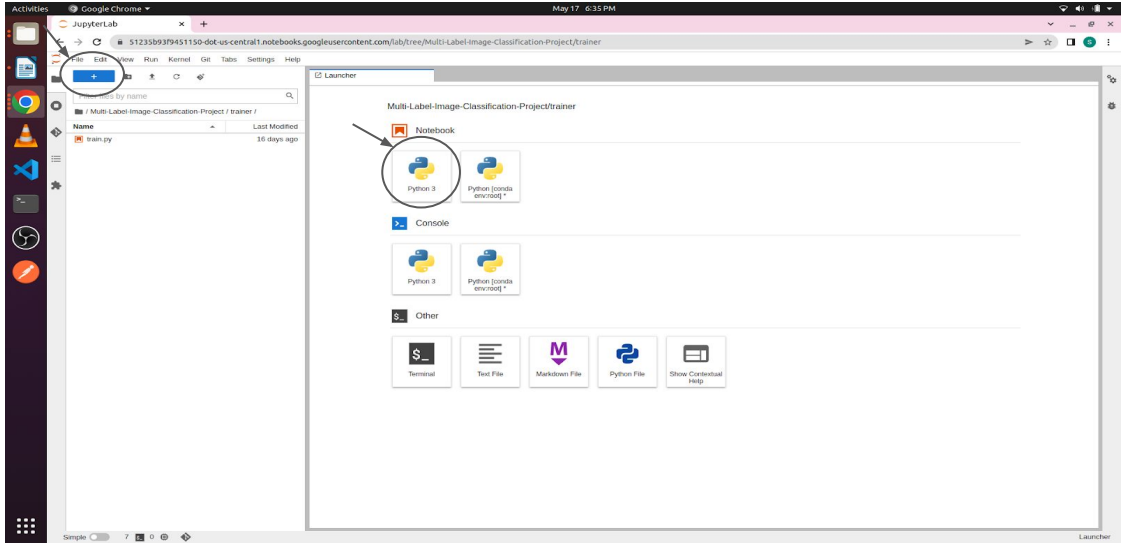
A service account determines what Google Cloud resources your service code can access. By default, a Google-managed service account is used with permissions appropriate for most models. You can also use a user-managed service account to customise permissions. Learn more.

= We won't enable model monitoring for this model, so next click **Create** to kick off the endpoint deployment. It will appear in Vertex AI **ENDPOINTS** section and you can then use it to do the predicting.



# STEP 10: Get predictions on the deployed model

- We'll get predictions on our trained model from a Python notebook, using the **Vertex Python API**.
- Go back to your notebook instance and create a Python 3 notebook from the Launcher. **Also upload the image that you want to use for prediction**.



- In your notebook, run the following in a cell to install the Vertex AI SDK:

```
!pip3 install google-cloud-aiplatform --upgrade --user
```

- Then add a cell in your notebook to import the SDK and create a reference to the endpoint you just deployed:

```
from google.cloud import aiplatform

PROJECT_ID = "XXXXXXXX"

ENDPOINT_ID = "4325213563224324234"

endpoint = aiplatform.Endpoint(
endpoint_name=f"projects/{PROJECT_ID}/locations/us-central1/endpoints/{ENDPOINT_ID}")
```

- Replace PROJECT_ID with your project id and ENDPOINT_ID with your active endpoint id.

- You can find your endpoint ID in the Endpoints section of the Vertex AI.

Google Cloud Platform

Search Products, resources, docs (/)

Vertex AI

- Dashboard
- Datasets
- Features
- Labelling tasks
- Workbench
- Pipelines
- Training
- Experiments
- Models
- Endpoints
- Batch predictions
- Metadata
- Matching engine
- Marketplace

Endpoints

CREATE ENDPOINT

REFRESH

LEARN

Endpoints are machine learning models made available for online prediction requests. Endpoints are useful for timely predictions from many users (for example, in response to an application request). You can also request batch predictions if you don't need immediate results.

To create an endpoint, you need at least one machine-learning model. Learn more

Region
us-central1 (Iowa)

Filter Enter a property name

| Name | ID | Status | Models | Region | Monitoring | Most recent alerts | Last updated | API | Notification | Labels |
|---|---|---|---|---|---|---|---|---|---|---|
| movie-classifier-endpoint | 7602102559280463872 | Active | 1 | us-central1 | Disabled | – | 12 May 2022, 03:35:02 | Sample request | | |

- Finally, make a prediction to your endpoint by copying and running the code below in a new cell:

```
import base64

encoded_string = ""

with open("movie-poster-image.jpg", "rb") as image_file:

    encoded_string = base64.b64encode(image_file.read())

instance = [{"b64_string": encoded_string.decode('utf-8') }]

prediction = endpoint.predict(instances=instance)

print(prediction.predictions[0])
```

- Run this cell, and you should see the following prediction output:

```
{'predicted_genres': ['Comedy', 'Drama', 'Romance']}
```

# 🎉 Congratulations! 🎉

You've learned how to use Vertex AI to:

- Train a model by providing the training code in a custom container. You used a Pytorch model in this example, but you can train a model built with any framework using custom containers.
- Deploy a Pytorch model using a custom container as part of the same workflow you used for training.
- Create a model endpoint and generate a prediction.