roduct is in a pre-release state and might change or have limited support. For more information, see the product launch stages (/products/#product-launch-stages).

This quickstart walks you through the process of:

- Copying a set of videos into Google Cloud Storage.
- Creating CSV files that list videos and their labels.
- Using AutoML Video Classification to create your dataset, and train and use your model.

1. Sign in (https://accounts.google.com/Login) to your Google Account.

   If you don't already have one, sign up for a new account (https://accounts.google.com/SignUp).

2. In the Cloud Console, on the project selector page, select or create a Cloud project.

★ **Note**: If you don't plan to keep the resources that you create in this procedure, create a project instead of selecting an existing project. After you finish these steps, you can delete the project, removing all resources associated with the project.

   Go to the project selector page (https://console.cloud.google.com/projectselector2/home/dashboard)

3. Make sure that billing is enabled for your Google Cloud project. Learn how to confirm billing is enabled for your project (/billing/docs/how-to/modify-project).

4. Enable the AutoML and Cloud Storage APIs.

   Enable the APIs (https://console.cloud.google.com/flows/enableapi?apiid=storage-component.googleapis.com,automl.googleapis.com,storage-api.google

5. Install the `gcloud` command line tool (/sdk/downloads#interactive).

6. Follow the instructions to create a service account and download a key file (/iam/docs/creating-managing-service-accounts#creating_a_service_account) for that account.

★ Service accounts are the only authentication option available with the AutoML API.

7. Set the `GOOGLE_APPLICATION_CREDENTIALS` environment variable to the path to the service account key file that you downloaded when you created the service account.

8. Set the `PROJECT_ID` environment variable to your Project ID (/resource-manager/docs/creating-managing-projects#identifying_projects).

   The AutoML API calls and resource names include your Project ID in them. The `PROJECT_ID` environment variable provides a convenient way to specify the ID.

9. If you are an owner for your project, add your service account to the **AutoML Editor** IAM role, replacing ***service-account-name*** with the name of your new service account. For example, `service-account1@myproject.iam.gserviceaccount.com`.

10. Otherwise (if you are not a project owner), ask a project owner to add both your user ID and your service account to the **AutoML Editor** IAM role.

Decide on a name for your dataset and use the following `curl` command to create a new dataset with that name.

The `curl` command uses the `gcloud auth application-default print-access-token` command to obtain an access token for a service account that you set up earlier in the topic and. The path to the service account key file is stored in the `GOOGLE_APPLICATION_CREDENTIALS` environment variable.

You should receive a response that contains the dataset id for your new dataset. For example: `VCN4798585402963263488`.

Import the sample training data into your dataset. The `importData` command takes, as input, the path to the CSV that contains paths to the training and test data CSV files. These files are made available in the "automl-video-demo-data" bucket on Google Cloud Storage.

- Replace ***your-dataset-id*** with the dataset identifier for your dataset (not the display name). For example: `VCN4798585402963263488`.

You should receive an operation id for your import data operation. For example: `VCN7506374678919774208`.

You can query the status of your import data operation by using the following `curl` command.

- Replace ***your-operation-id*** with the operation id for your import data operation.

The import operation can take some time to complete. When the import task is finished, you will see `done: true` in the status of the operation with no errors listed, as shown in the following example.

You can get your list of datasets and the number of sample videos that were imported into the dataset by using the following command.

You should see output similar to the following:

After you have created your dataset and imported your training data into your dataset, you can train your custom model.

Train your model by using the following `curl` command.

- Replace ***your-dataset-id*** with the dataset identifier for your dataset (not the display name).
- Replace ***your-model-name*** with a name that you choose for your model.

You should receive an operation id for your import data operation. For example: `VCN1741767155885539328`.

You can query the status of your model training operation by using the following `curl` command.

- Replace ***your-operation-id*** with the operation id for your training operation.

You should see output similar to the following. When the operation is complete, you will see `done: true` with no errors listed.

After your model training operation successfully completes, you can verify that your model is available by using the following command to list the models for your project.

You should see output similar to the following:

You can request annotations (predictions) for videos by using the `batchPredict` command. The `batchPredict` command takes, as input, a CSV file stored in your Google Cloud Storage bucket that contains the paths to the videos to annotate, and the start and end times that identify the segment of video to annotate. For this quickstart, this CSV file is named `hmdb_split1_train_gs.csv`.

Run the following command to make a batch (asynchronous) predict request.

- Replace ***your-model-id*** with the identifier for your model.
- Replace ***your-output-bucket*** with a Google Cloud Storage bucket that will contain the results of your prediction.
- Replace ***your-object-id*** with an object name that identifies where to store the output of your prediction request in your Google Cloud Storage bucket. You must have write permissions to the Google Cloud Storage bucket.

You should receive an operation id for your batch predict request. For example: `VCN926615623331479552`.

You can query the status of your batch predict operation by using the following `curl` command.

- Replace ***your-operation-id*** with the operation id for your batch predict operation.

Depending on the number of videos that you specified in your CSV file, the batch predict task can take some time to complete. When the task is finished, you will see `done: true` in the status of the operation with no errors listed, as shown in the following example.

When the batch predict task is complete, the output of the prediction is stored in the Google Cloud Storage bucket that you specified in your command. There is a JSON file for each video segment. For example:

my-video-01.avi.json

If you no longer need your custom model and the related dataset, you can delete them.

You can list the models for your project, along with their identifiers, by using the following command:

You can delete a model by using the following command.

- Replace ***your-model-id*** with the identifier for your model.

You can list the datasets for your project, along with their identifiers, by using the following command:

You can delete a dataset by using the following command.

- Replace ***your-dataset-id*** with the identifier for your model.