This tutorial demonstrates how to create a custom translation model using AutoML Translation. The application trains a custom model using an English to Spanish dataset of technology-oriented sentence pairs from software localization.

The tutorial covers training the custom model, evaluating its performance, and translating new content.

ıg Code Samples: Most of the code samples in this tutorial are taken from larger code files located in GitHub (https://github.com/). You can view and download the ete file from which a code sample is taken by clicking the "View on GitHub" button provided above a sample.

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 Translation 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).

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. For example:

8. Add your new service account to the **AutoML Editor** IAM role with the following commands. Replace ***project-id*** with the name of your GCP project and replace ***service-account-name*** with the name of your new service account, for example `service-account1@myproject.iam.gserviceaccount.com`.

9. Allow the AutoML Translation service accounts to access your Google Cloud project resources:

10. Install the client library (/translate/automl/docs/client-libraries).
11. Set the PROJECT_ID and REGION_NAME environment variables.

    Replace ***project-id*** with the Project ID (/resource-manager/docs/creating-managing-projects#identifying_projects) of your Google Cloud Platform project. AutoML Translation currently requires the location `us-central1`.

12. Create a Google Cloud Storage bucket to store the documents that you will use to train your custom model.

    The bucket name must be in the format: `$PROJECT_ID-vcm`. The following command creates a storage bucket in the `us-central1` region named `$PROJECT_ID-vcm`.

13. Download (/translate/automl/docs/sample/automl-translation-data.zip) the archive file containing the sample data for training the model, extract its contents, and upload the files to your Google Cloud Storage bucket.

    See Preparing your training data (/translate/automl/docs/prepare) for details about the formats.

★ The sample code in this tutorial uses the English to Spanish dataset. Datasets with target languages German, French, Russian, and Chinese are also available. If you use one of these alternate datasets, replace the language code **es** in the samples with the appropriate language code.

14. In the `en-es.csv` file from the previous step, replace `{project_id}` with the Project ID for your project.

You can download the source code from the location provided below. After downloading, you can copy the source code into your Google Cloud project folder.

The first step in creating a custom model is to create an empty dataset that will eventually hold the training data for the model. When you create a dataset, you specify the source and target languages for the translation.

Run the `create_dataset` function to create an empty dataset. You must modify the following lines of code:

- Set the `project_id` to your PROJECT_ID
- Set the `display_name` for the dataset (`en_es_dataset`)
- Modify the `target_language_code` field from `ja` to `es`

The response includes the details of the newly created dataset, including the Dataset ID that you'll use to reference the dataset in future requests. We recommend that you set an environment variable `DATASET_ID` to the returned Dataset ID value.

The next step is to populate the dataset with a list of training sentence pairs.

The `import_dataset` function interface takes as input a .csv file that lists the locations of all training documents and the proper label for each training document. (See Prepare your data (/translate/automl/docs/prepare) for details about the required format.) For this tutorial, we will be using `en-es.csv`, which you uploaded to Google Cloud Storage above.

Run the `import_data` function to import the training content. You must modify the following lines of code:

- Set the `project_id` to your PROJECT_ID
- Set the `dataset_id` for the dataset (from the output of the previous step)
- Set the `path` which is the URI of the (`gs://YOUR_PROJECT_ID-vcm/en-es.csv`)

Now that you have a dataset of labeled training documents, you can train a new model.

model training can take several hours to complete.

To run `create_model`, you must modify the following lines of code:

- Set the `project_id` to your PROJECT_ID
- Set the `dataset_id` for the dataset (from the output of the previous step)
- Set the `display_name` for the new model (en_es_test_model)

The `create_model` function kicks off a training operation and prints the operation name. Training happens asynchronously and can take a while to complete, so you can use the operation ID to check training status (/translate/automl/docs/models#get-operation). When training is complete, `create_model` returns the Model ID. As with the Dataset ID, you might want to set an environment variable `MODEL_ID` to the returned Model ID value.

After training, you can evaluate your model's readiness by reviewing its BLEU score (/translate/automl/docs/evaluating).

The `list_model_evaluations` function takes the Model ID as a parameter.

Make a request to display the overall evaluation performance of the model by executing the following request. You must modify the following lines of code:

- Set the `project_id` to your PROJECT_ID
- Set the `model_id` to your model's id

BLEU score is too low, you can strengthen the training dataset and re-train your model. For more information, see Evaluating models slate/automl/docs/evaluating).

When your custom model meets your quality standards, you can use it to translate novel content.

For the `predict` function you must modify the following lines of code:

- Set the `project_id` to your PROJECT_ID
- Set the `model_id` to your model's id
- Set the `file_path` to the downloaded file ("resources/input.txt")

The function returns the translated content.

Above is the Spanish translation for the English sentence: “View and manage your Google Tag Manager accounts.” Contrast this custom translation with the translation from the base Google model:

When you are done using this sample model, you can delete it permanently. You will no longer be able to use the model for prediction.

Make a request with operation type `delete_model` to delete a model you created you must modify the following lines of code:

- Set the `project_id` to your PROJECT_ID
- Set the `model_id` to your model's id