**SAP Leonardo Machine Learning Foundation**

Version 1.0

**Part 3- Retraining a model in SAP Leonardo**

**Objective**:

In this Hands On exercise we shall see how to retrain the in-built model of SAP Leonardo to suite our requirement. By the end of this exercise we will know:

1. How to use Cloud Foundry Command Line Interface (CF CLI),
2. Using MINIO client to access AWS S3 storage and copy our training data to AWS S3
3. Finally using the retraining API from SAP Leonardo service key to retrain our model.

**Pre-requisite:**

* Intended audience are expected to complete part 1 and part 2 of the document.
* The audience should have the following files provided as training material:
* Tech\_brands -> It contains the images for training, testing and validation

Installation of CF CLI, Minio Client and ML plugin for CF CLI are included in this document.

# Retraining a Machine Learning API

* 1. Configuring cloud foundry CLI (Command line Interface)

|  |  |  |
| --- | --- | --- |
| Log on to cloud foundry account.  Click on “**Useful Links**” -> “**Tools**” |  |  |
| Click on “**CLOUD**” tab.  Click on “**Cloud Foundry CLI**”, to install the CLI. |  |  |
| Go down to find the download link for windows 64bit installer.  Click on the link. |  |  |
| Extract the zip file.  Double click on the “**cf-cli-installer\_6.42.0\_winx64**”.  Follow the installation wizard and click on “**Install**”. |  |  |
| Open the command prompt.  Press “**windows**” + “**r**” and enter “**cmd**” to open command prompt. |  |  |
| Enter the API endpoint from the cloud foundry subaccount  Enter the following commands:  **cf api <API Endpoint>**  Enter the following command to login:  **cf login**  Enter the SAP cloud platform “Email” and “Password” |  |  |
| If you are using a trial account you should have only one Organization and few spaces. Select the space “ML”. |  |  |

* 1. Install ML Plugin and configure SAP ML in cloud foundry

|  |  |  |
| --- | --- | --- |
| Click on “**Useful Links**” -> “**Tools**” and go the “**ML Foundation**” tab.  Click on the ML foundation plugin for windows.  Extract the ZIP folder. |  |  |
| In the command prompt navigate to the “sapmlcli-win-1.0.0” folder type the following command:  **cf install-plugin -f sapmlcli.exe** |  |  |
| To see the configurations of ML type :  **cf sapml config get** |  |  |
| If the “ml\_foundation\_service\_name” is not the same as your ML service name (for trial account it will be – “ml-foundation-trial-beta”)change it using the command :  **cf sapml config set ml\_foundation\_service\_name <your ML service name>**  similarly change the “retraining\_image\_api” to your retraining image api, from service key:  **cf sapml config set retraining\_image\_api <your retraining\_image\_api>** |  |  |

* 1. Initialize AWS S3 file system and upload data to AWS S3 (using minio client)

|  |  |  |
| --- | --- | --- |
| Initialize the file system  By executing this command:  **cf sapml fs init** |  |  |
| To get the <endpoint> <access key> <secret key> of the AWS file service execute the following command:  **cf sapml fs config**  {This will be used to connect to AWS S3} |  |  |
| Download minio client from the below URL:  minio.io  save it in the downloads folder and navigate to downloads in cmd.  Type:-  **mc**  to check minio client is working properly.  We will use minio client to copy our training data to AWS S3. |  |  |
| Execute the following command to connect to AWS S3 using the <endpoint> <access key> <secret key>  **mc config host add saps3 https://<endpoint> <access key> <secret key>**  {We got this information from “cf sapml fs config” command} |  |  |
| Download the retraining data from :  <https://github.com/saphanaacademy/MLF>  {You can also create your own training data}  Take the “brands” folder and put in the “downloads” . For convenience we renamed it to “tech\_brands” |  |  |
| Copy the “tech\_brands” folder to cloud (Provided as training material):  **mc cp tech\_brands saps3/data --recursive** |  |  |
| Check the “tech\_brands” folder in cloud:  **cf sapml fs list tech\_brands/** |  |  |

* 1. Run the Retraining Job

|  |  |  |
| --- | --- | --- |
| To run the Retraining Job We need a retraining.JSON file which specifies the configuration.  You can find it in the training content or you can create this file on your own.  Keep this file in “downloads”. |  |  |
| Submit the retraining job:  **cf sapml retraining job\_submit retrain.json -m image**  Check the job status using:  **cf sapml retraining jobs -m image** |  |  |
| Check the job created using:  **cf sapml fs list** |  |  |
| Get the log file of the job:  **Cf sapml fs get <job-ID>/retraining.log retraining.log**  The log file can be found on the same path where we are currently executing the commands (Downloads) |  |  |
| See the available retraining models:  **cf sapml retraining models -m image** |  |  |

* 1. Deploy the Retrained model

|  |  |  |
| --- | --- | --- |
| To deploy the retrained model (tech\_brands) execute the following command:  **cf sapml retraining model\_deploy tech\_brands 1 -m image**  To check the deployment status run the following command:  **cf sapml retraining model\_deployments -m image** |  |  |

* 1. Test the retrained model

|  |  |
| --- | --- |
| Open Postman from google chrome extensions/ Postman App.  Generate the access token as shown previously.  The retrained API will be:  **<IMAGE\_CLASSIFICATION\_API>/models/<model\_name>/versions/<version>** |  |
| Eg.  https://mlftrial-retrain-image-api.cfapps.eu10.hana.ondemand.com/api/v2/image/retraining/tech\_brands/versions/1 | |
| Sent the HTTP request using POST and same configurations as done previously. |  |

We executed the same image in the original IMAGE\_CLASSIFICATION\_API and the retrained API :

|  |  |  |
| --- | --- | --- |
| Image | IMAGE\_CLASSIFICATION\_API | Retrained API |
|  |  |  |

Note: In Trial account we can deploy only one model at a time. In the next exercise we shall deploy another model, so we should un-deploy the existing model first

Un-Deploy the Model

|  |  |  |
| --- | --- | --- |
| To check the deployed model type the following command:  “**cf sapml modelserver list”** |  |  |
| To Check for a specific deployed model details with model ID:  “**cf sapml modelserver get <Model ID>”** |  |  |
| To Un-deploy/delete the model from server:  **“cf sapml modelserver delete <Model ID>”** |  |  |

Thus we have deleted the model and we are ready for the next part.