**AI 620 Emerging Topics in Artificial Intelligence**

**HOS10A Image Classification Using Amazon SageMaker**

03/21/2023 Developed by Yared Shewarade

09/23/2024 Updated by Anh Nguyen

12/3/2024 Reviewed by Jonathan Koerber

School of Technology and Computing (STC) @City University of Seattle (CityU)

**Before You Start**

* The directory path shown in screenshots may be different from yours.
* Some steps are not explained in the tutorial**.** If you are not sure what to do:
  1. Consult the resources listed below.
  2. If you cannot solve the problem after a few tries, ask a student worker for help.

**Learning Outcomes**

Students will be able to learn:

* Introduction to Image Classification
* Image Classification setup using Amazon SageMaker

**Resources**

* Tripuraneni, S., & Song, C. (2019). *Hands-on artificial intelligence on amazon web services: Decrease the time to market for AI and ML applications with the power of AWS* (1st ed.). Packt.

# Introduction to Neural Topic Model (NTM)

Image classification has been one of the leading research fields to successfully classify images and solve many business problems across a variety of industries. For example, the entire autonomous vehicle industry is dependent on the accuracy of these image classification and object detection models. Amazon SageMaker image classification algorithm is a supervised learning algorithm that supports multi-label classification. It takes an image as input and outputs one or more labels assigned to that image. It uses a convolutional neural network that can be trained from scratch or trained using transfer learning when a large number of training images are not available.

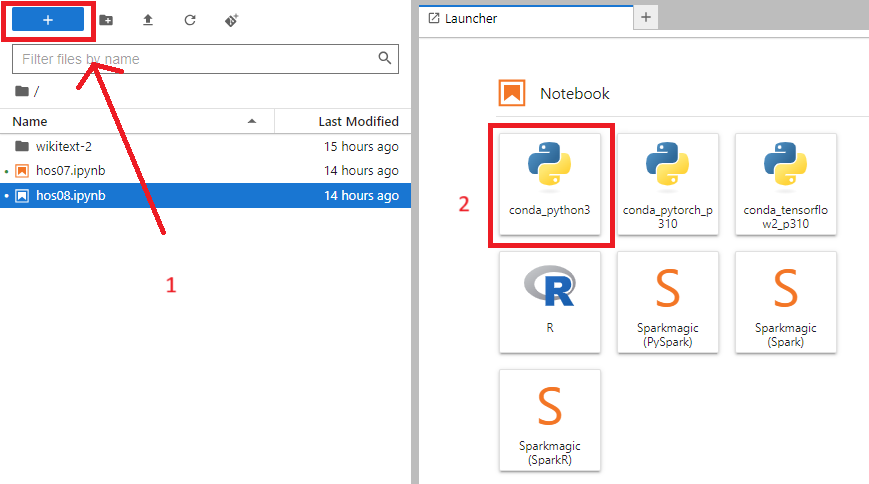
The recommended input format for the Amazon SageMaker image classification algorithms is Apache MXNet RecordIO. However, we can also use raw images in .jpg or .png format.

# Image classification using Amazon SageMaker

Note: For submission, take the screenshot for all steps and save it in your local repository along with your code.

## Install the required module and prepare the data

Open the notebook instance you created in HOS07 > Open JupyterLab > New Launcher (+) > conda\_python3



Text

Description automatically generated



Let’s download the data and transfer them to S3.

Text

Description automatically generated

## Training the model

Let’s train the model once we have the data available in the correct format for training.

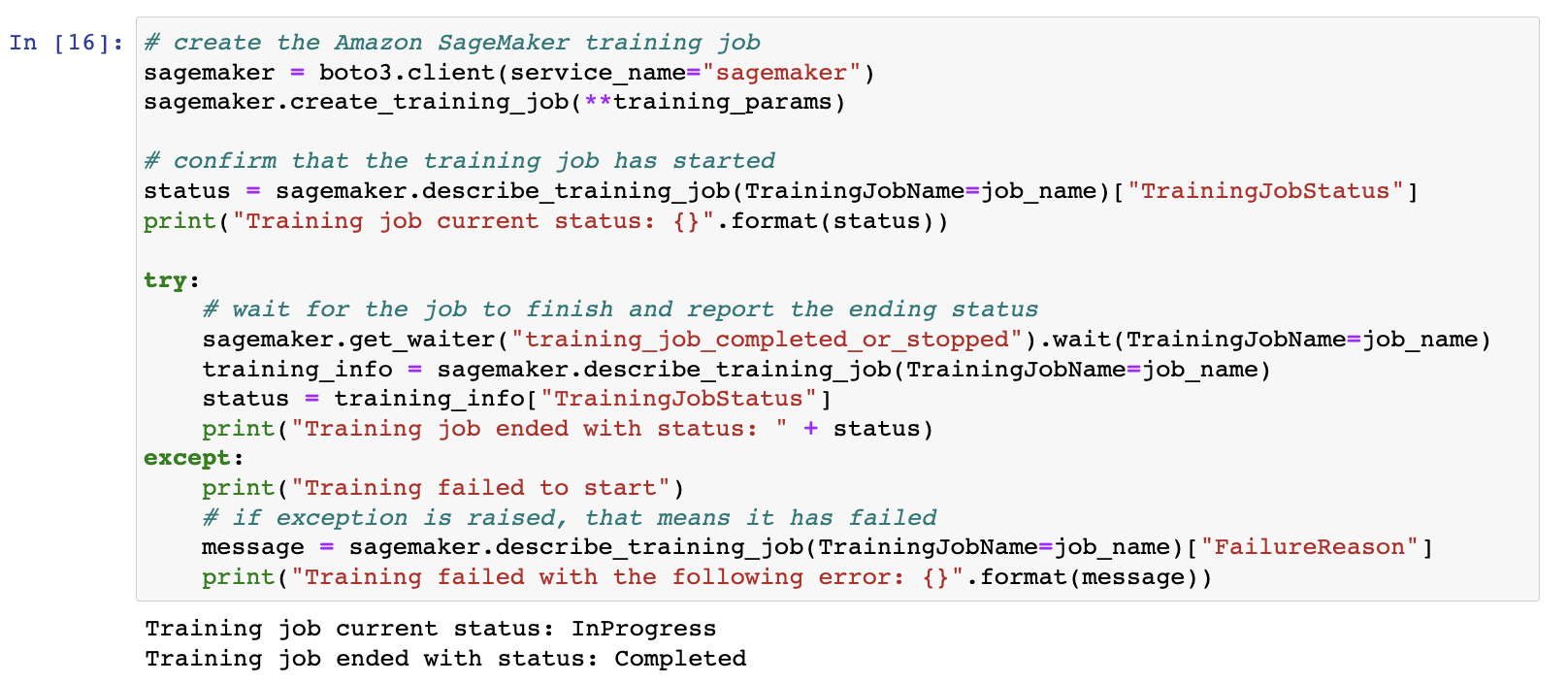


Graphical user interface, text, application, email

Description automatically generated

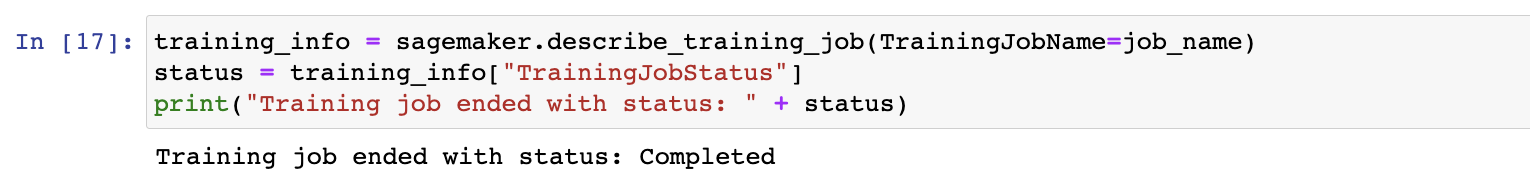
Timeline

Description automatically generated with medium confidence



If we see the message “Training job current status: Completed”, it indicates that the training is successfully completed.

Let’s print out the status of the training



**HOS submission instructions:**

1. Please install the GitHub Desktop: <https://cityuseattle.github.io/docs/git/github_desktop/>

2. Clone, organize, and submit your work through GitHub Desktop: <https://cityuseattle.github.io/docs/hoporhos>