

Sprint-4

Train The Model On IBM

Train The Model on IBM Watson

Date	15 Nov 2022
TeamID	PNT2022TMID02236
ProjectName	Classification of Arrhythmia by Using Deep Learning with 2-D ECG Spectral Image Representation

TASK:

Train The Model on IBM Watson:

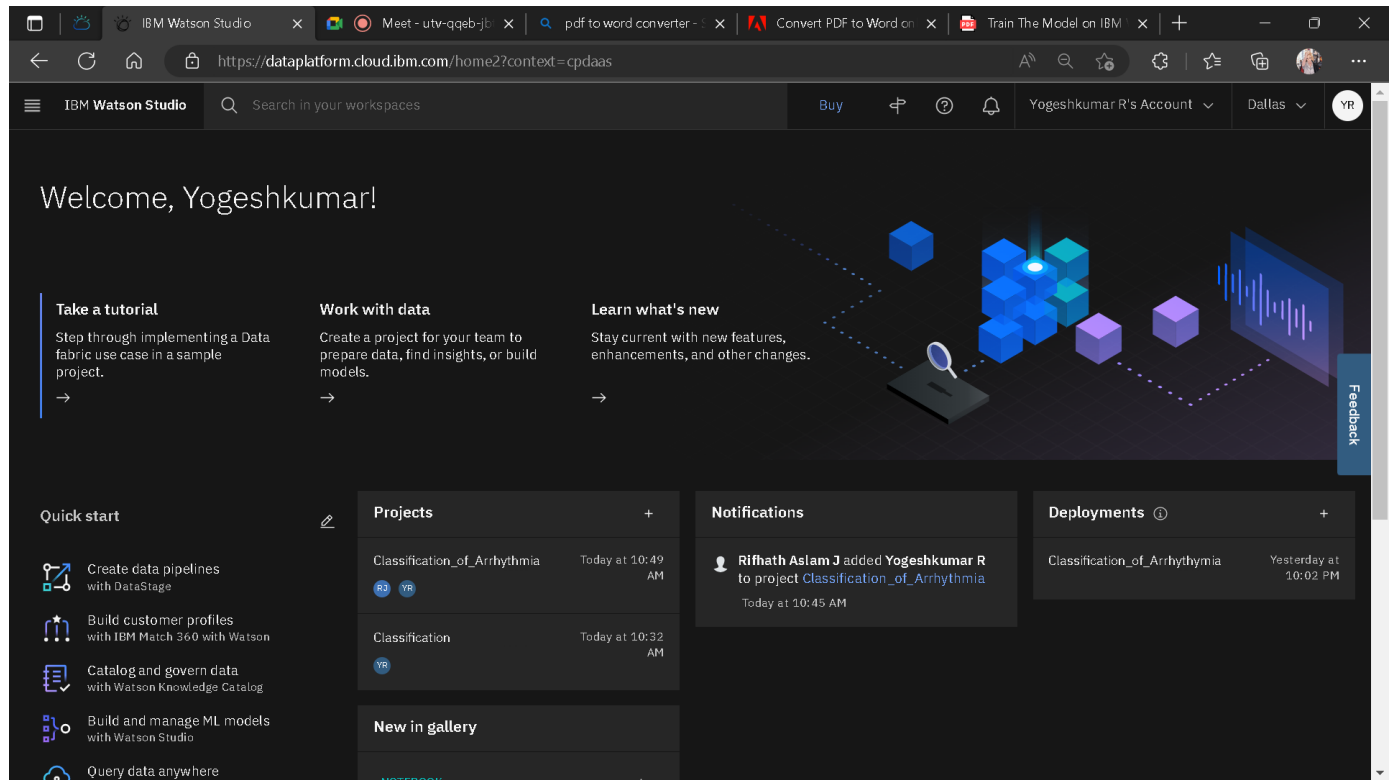
GO ON WATSON SERVICES(SCREEN SHOT):

Resource list /

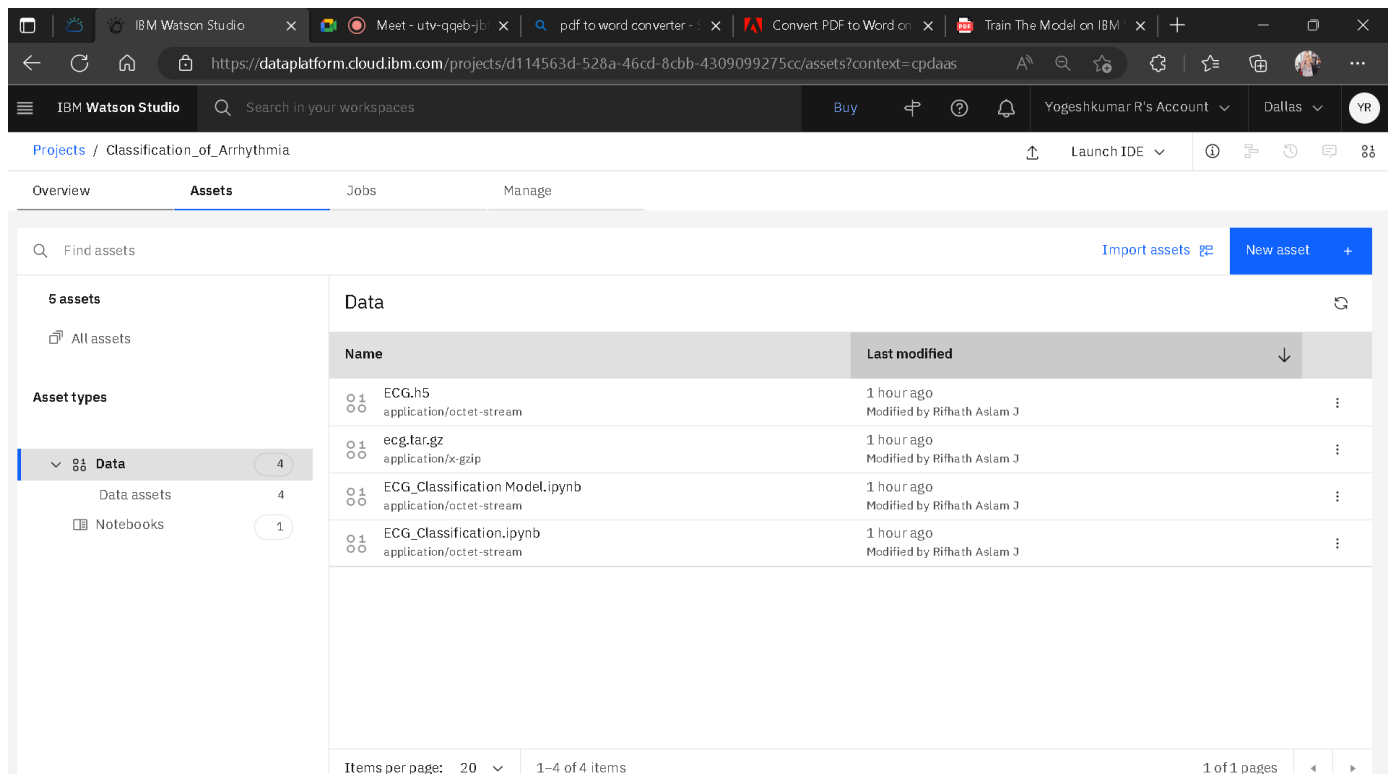
Watson Studio-11

Active [Add tags](#) [Details](#) [Actions...](#)

GO ON NEW PROJECT (SCREEN SHOT):



GO ON ASSEST(SCREEN SHOT):



GO ON IBM_PROJECT(SCREEN SHOT):

The screenshot shows the IBM Watson Studio interface. The top navigation bar includes the IBM Watson Studio logo, a search bar, and user account information (Yogeshkumar R's Account, Dallas). The main content area is titled "Projects / Classification_of_Arrhythmia" and has tabs for Overview, Assets, Jobs, and Manage. The "Assets" tab is active, displaying a list of assets. On the left, there's a sidebar with "Find assets" and "Asset types" (Data: 4, Notebooks: 1). The main table lists the following assets:

Name	Last modified
ECG.h5 application/octet-stream	1 hour ago Modified by Rithath Aslam J
ecg.tar.gz application/x-gzip	1 hour ago Modified by Rithath Aslam J
ECG_Classification Model.ipynb application/octet-stream	1 hour ago Modified by Rithath Aslam J
ECG_Classification.ipynb application/octet-stream	1 hour ago Modified by Rithath Aslam J

At the bottom, it shows "Items per page: 20" and "1-4 of 4 items".

JUPYTER NOTEBOOK(SCREEN SHOT):

The screenshot shows a Jupyter Notebook in IBM Watson Studio. The top navigation bar is the same as the previous screenshot. The notebook is titled "Ibm_NTP" and is part of the "Classification_of_Arrhythmia" project. The code in the notebook is as follows:

```
In [22]:
import os, types
import pandas as pd
from boto3.client import Config
import boto3

def __iter__(self): return 0

#@hidden_cell
# The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.
# You might want to remove those credentials before you share the notebook.

if os.environ.get('RUNTIME_ENV_LOCATION_TYPE') == 'external':
    endpoint_69c05974e5c84795a978662af2736fc1 = 'https://s3.us.cloud-object-storage.appdomain.cloud'
else:
    endpoint_69c05974e5c84795a978662af2736fc1 = 'https://s3.private.us.cloud-object-storage.appdomain.cloud'

client_69c05974e5c84795a978662af2736fc1 = boto3.client(service_name='s3',
    iam_api_key_id='KlInyzYsEAMBFEgZMg_6j5CyAcJpa01CPz3gd9KEdr_',
    iam_auth_endpoint='https://iam.cloud.ibm.com/oidc/token',
    config=Config(signature_version='oauth'),
    endpoint_url=endpoint_69c05974e5c84795a978662af2736fc1)

streaming_body_1 = client_69c05974e5c84795a978662af2736fc1.get_object(Bucket='ecgimagebasedheartbeatclassification-donotdelete-pr-l2ugdciyflqayf', Key='data.zip')['Body']

# Your data file was loaded into a boto3.response.StreamingBody object.
# Please read the documentation of boto3 and pandas to learn more about the possibilities to load the data.
# boto3 documentation: https://boto3.amazonaws.com/v1/documentation/api/latest/guide/quickstart.html#python
# pandas documentation: http://pandas.pydata.org/
```

Projects / Classification_of_Arrhythmia / IBM_NTP

```
File Edit View Insert Cell Kernel Help Trusted | Python 3.9

requirement already satisfied: pytz>=2017.3 in /opt/conda/envs/python-3.9/lib/python3.9/site-packages (from pandas-watson-machine-learning-client==2021.3)
Requirement already satisfied: numpy>=1.17.3 in /opt/conda/envs/python-3.9/lib/python3.9/site-packages (from pandas-watson-machine-learning-client==2021.3)
Installing collected packages: watson-machine-learning-client
Successfully installed watson-machine-learning-client-1.0.391

In [44]: # Replace the credentials that you got from watson machine learning service
from ibm.watson_machine_learning import APIClient
wml_credentials = {
    "url": "https://us-south.ml.cloud.ibm.com",
    "apikey": "MpwfHtY_VXAXAgD4uzxBAb00C2F3stDEVyb3oXk1UaRn"
}
client = APIClient(wml_credentials)

In [45]: client = APIClient(wml_credentials)

In [46]: def guid_from_space_name(client, space_name):
space = client.spaces.get_details()
#print(space)
return(next(item for item in space['resources'] if item['entity']['name'] == space_name)['metadata']['id'])

In [47]: space_uid = guid_from_space_name(client, 'image_classification')
print("Space UID = " + space_uid)

Space UID = 26b6a24d-f745-4d09-b456-8f6dfd7d9ca6

In [48]: client.set.default_space(space_uid)

Out[48]: 'SUCCESS'
```