

Sprint-4

Train The Model On IBM

Train The Model on IBM Watson

Date	15 Nov 2022
TeamID	PNT2022TMID15718
ProjectName	Classification of Arrhythmia by Using Deep Learning with 2-D

TASK:

Train The Model on IBM Watson:

GO ON WATSON SERVICES(SCREEN SHOT):

The screenshot displays the IBM Cloud Watson Studio service page. The 'Create' tab is active, showing a location dropdown set to 'Frankfurt (eu-de)'. Below this, a pricing plan table is visible. The 'Lite' plan is highlighted, showing it is free and includes 1 authorized user and 10 capacity unit-hours monthly limit. A warning message indicates that only one Lite plan instance can exist per resource group. The 'Create' button is visible at the bottom right.

Plan	Features	Pricing
Lite	<ul style="list-style-type: none">1 authorized user10 capacity unit-hours monthly limitEnvironment = # of capacity units required per hour<ul style="list-style-type: none">• 1 vCPU + 4 GB RAM = 0.5• 2 vCPU + 8 GB RAM = 1• 4 vCPU + 16 GB RAM = 2Decision Optimization + Watson NLP = Environment + 5	Free
Professional	Unlimited collaborators	\$0.98 USD/Capacity Unit-Hour

GO ON NEW PROJECT (SCREEN SHOT):

Welcome, Hemanth suresh!

Take a tutorial
Step through implementing a Data fabric use case in a sample project.
→

Work with data
Create a project for your team to prepare data, find insights, or build models.
→

Learn what's new
Stay current with new features, enhancements, and other changes.
→

Quick start

- Create data pipelines with DataStage
- Build customer profiles with IBM Match 360 with Watson
- Catalog and govern data with Watson Knowledge Catalog
- Build and manage ML models with Watson Studio
- Query data anywhere with Watson Query

What's new
Spark 3.3 replaces Spark 3.2 for Watson

Projects

Project Name	Last Modified
ECG-classification	Today at 12:27 PM
Ecg_classification	Today at 10:42 AM
machine-learning	Today at 12:36 AM

New in gallery

- NOTEBOOK

Use AutoAI and Lale to predict

Notifications

① No notifications
You will see your most recent notifications here.

Deployments

Deployment Name	Last Modified
ecg_classification	Today at 01:50 PM

Feedback

GO ON IBM_PROJECT(SCREEN SHOT):

Projects / ECG-classification

Overview Assets Jobs Manage

Find assets Import assets New asset

4 assets

Asset types

- Data (3)
- Notebooks (1)

All assets

Name	Last modified
ECG_Classification Notebook	3 hours ago Modified by you
fig_23.png PNG	3 hours ago Modified by you
fig_16.png PNG	3 hours ago Modified by you
Dataset.zip ZIP	5 hours ago Modified by you

Items per page: 20 1-4 of 4 items 1 of 1 pages

Data in this project

Drop data files here or browse for files to upload

JUPYTER NOTEBOOK(SCREEN SHOT):

The screenshot displays the IBM Watson Studio interface. At the top, there's a navigation bar with 'Projects / ECG-classification / ECG_Classification'. Below this is a toolbar with icons for file operations and a menu. The main area shows a Jupyter notebook with the following code:

```

In [3]: keras.__version__
Out[3]: '2.7.0'

In [4]: import tensorflow as tf
        tf.__version__
Out[4]: '2.7.2'

In [5]: from tensorflow.keras.models import Sequential
        from tensorflow.keras.layers import Dense
        from tensorflow.keras.layers import Convolution2D
        from tensorflow.keras.layers import MaxPooling2D
        from tensorflow.keras.layers import Flatten, BatchNormalization, Dropout

In [6]: from tensorflow.keras.preprocessing.image import ImageDataGenerator

In [7]: train_datagen = ImageDataGenerator(rescale = 1./255, shear_range = 0.2, zoom_range = 0.2, horizontal_flip = True)
        test_datagen = ImageDataGenerator(rescale = 1./255)

In [8]:
import sys, types
import pandas as pd
from core.client import Config
from ibm_boto3

def __init__(self):
    return 0

def main_cell
Following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.
If you want to remove those credentials before you share the notebook.
client = ibm_boto3.client(service_name='s3',
                           api_key_id='IgnIGX_aPuKu4DHFTUABCGgx0UHZG6JoBB0lueiypNX0',
                           auth_endpoint='https://iam.cloud.ibm.com/oidc/token',
                           sig=Config(signature_version='oauth'),
                           endpoint_url='https://s3.private.us.cloud-object-storage.appdomain.cloud')

# 'ecgclassification-donotdelete-pr-fhxyoyjwxtmly'

```

On the right side, there's a 'Data' panel with a table showing uploaded files:

Files	Connections
Dataset.zip	Insert to code
fig_16.png	Insert to code
fig_23.png	Insert to code

The screenshot displays the IBM Watson Studio environment. At the top, there's a navigation bar with 'Projects / ECG-classification / ECG_Classification'. Below it, a toolbar contains icons for File, Edit, View, Insert, Cell, Kernel, and Help. The main area is a Jupyter Notebook with a code cell containing Python code. The code defines a boto3 client for S3, sets up credentials, and downloads a file named 'Dataset.zip'. It then uses pandas to load the data into a DataFrame. The right sidebar shows a 'Data' panel with a 'Files' tab, listing 'Dataset.zip', 'fig_16.png', and 'fig_23.png' with options to 'Insert to code'.

```

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.

client = ibm_boto3.client(service_name='s3',
                           ibm_api_key_id='IgnIGX_aPuKu4DFTUABCGgx0UHZG6JoBB0LueiypNX0',
                           ibm_auth_endpoint='https://iam.cloud.ibm.com/oidc/token',
                           config=Config(signature_version='oauth'),
                           endpoint_url='https://s3.private.us.cloud-object-storage.appdomain.cloud')

key = 'ecgclassification-donotdelete-pr-fhxyoyjwxmly'
object_key = 'Dataset.zip'

streaming_body_3 = cos_client.get_object(Bucket=bucket, Key=object_key)['Body']

# The data file was loaded into a botocore.response.StreamingBody object.
# Please read the documentation of ibm_boto3 and pandas to learn more about the possibilities to load the data.
# ibm_boto3 documentation: https://ibm.github.io/ibm-cos-sdk-python/
# pandas documentation: http://pandas.pydata.org/
# streaming_body_3 = BytesIO(streaming_body_3.read())
# streaming_body_3 = BytesIO(streaming_body_3.read())

In [9]: from io import BytesIO
import zipfile
unzip=zipfile.ZipFile(BytesIO(streaming_body_3.read()), 'r')
file_paths=unzip.namelist()
for path in file_paths:
    unzip.extract(path)

In [10]: import os
filename=os.listdir('/home/wsuser/work/Dataset/Test')

In [11]: x_train = train_datagen.flow_from_directory(directory=r'/home/wsuser/work/Dataset/Train',target_size = (64,64),batch_size=32)
x_test = test_datagen.flow_from_directory(directory=r'/home/wsuser/work/Dataset/Test',target_size = (64,64),batch_size=32)

Found 15341 images belonging to 6 classes.
Found 6825 images belonging to 6 classes.

In [12]: x_train.class_indices

```

