Team ID : PNT2022TMID42525

Date : 29 Oct 2022

Sprint - 4

```
from google.colab import drive
drive.mount('/content/drive')
```

```
    Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remo
```

```
#Extracting Data
!unzip "/content/drive/MyDrive/IBM Project Development/Classification of Arrhythmia by Using Deep Learning with 2-D ECG Spec
```

```
      inflating: data/train/Ventricular Fibrillation/VFEfig_468.png
      inflating: data/train/Ventricular Fibrillation/VFEfig_469.png
      inflating: data/train/Ventricular Fibrillation/VFEfig_47.png
      inflating: data/train/Ventricular Fibrillation/VFEfig_470.png
      inflating: data/train/Ventricular Fibrillation/VFEfig_471.png
      inflating: data/train/Ventricular Fibrillation/VFEfig_472.png
      inflating: data/train/Ventricular Fibrillation/VFEfig_48.png
      inflating: data/train/Ventricular Fibrillation/VFEfig_49.png
      inflating: data/train/Ventricular Fibrillation/VFEfig_50.png
      inflating: data/train/Ventricular Fibrillation/VFEfig_51.png
      inflating: data/train/Ventricular Fibrillation/VFEfig_52.png
      inflating: data/train/Ventricular Fibrillation/VFEfig_53.png
      inflating: data/train/Ventricular Fibrillation/VFEfig_54.png
      inflating: data/train/Ventricular Fibrillation/VFEfig_55.png
      inflating: data/train/Ventricular Fibrillation/VFEfig_56.png
      inflating: data/train/Ventricular Fibrillation/VFEfig_57.png
      inflating: data/train/Ventricular Fibrillation/VFEfig_58.png
      inflating: data/train/Ventricular Fibrillation/VFEfig_59.png
      inflating: data/train/Ventricular Fibrillation/VFEfig_60.png
                                  cular Fibrillation/VFEfig_61.png
                                  cular Fibrillation/VFEfig_62.png
                                  cular Fibrillation/VFEfig_63.png
```

Saved successfully!

```
  inflating: data/train/Ventricular Fibrillation/VFEfig_64.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_65.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_66.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_67.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_68.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_69.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_70.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_71.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_72.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_73.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_74.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_75.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_76.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_77.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_78.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_79.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_80.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_81.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_82.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_83.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_84.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_85.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_86.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_87.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_88.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_89.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_90.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_91.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_92.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_93.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_94.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_95.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_96.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_97.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_98.png
  inflating: data/train/Ventricular Fibrillation/VFEfig_99.png
```

Image Augmentation / Preprocessing :

Saved successfully!

```
#Import req. Lib.
from tensorflow.keras.preprocessing.image import ImageDataGenerator


#Augmentation On Training Variable
train_datagen = ImageDataGenerator(rescale= 1./255,
                 zoom_range=0.2,
                 horizontal_flip =True)


#Augmentation On Testing Variable
test_datagen = ImageDataGenerator(rescale= 1./255)


#Augmentation On Training Variable
ftrain = train_datagen.flow_from_directory('/content/data/train',
                                           target_size=(64,64),
                                           class_mode='categorical',
                                           batch_size=100)
```

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

```
#Augmentation On Testing Variable
ftest = test_datagen.flow_from_directory('/content/data/test',
                                          target_size=(64,64),
                                          class_mode='categorical',
                                          batch_size=100)
```

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

Model Building :

Adding Layers :

Saved successfully! ×

```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Convolution2D, MaxPooling2D, Flatten, Dense


# Build a CNN Block:
model = Sequential() #intializing sequential model
model.add(Convolution2D(32,(3,3),activation='relu', input_shape=(64,64,3))) #convolution layer
model.add(MaxPooling2D(pool_size=(2, 2))) #Maxpooling layer
model.add(Flatten()) #Flatten layer
model.add(Dense(400,activation='relu')) #Hidden Layer 1
model.add(Dense(200,activation='relu')) #Hidden Layer 2
model.add(Dense(6,activation='softmax')) #Output Layer
```

Compiling :

```
# Compiling The Model...
model.compile(optimizer='adam',loss='categorical_crossentropy',metrics=['accuracy'])
```

Fit / Train The Model :

```
#Train Model:
model.fit_generator(ftrain,
                    steps_per_epoch=len(ftrain),
                    epochs=10,
                    validation_data=ftest,
                    validation_steps=len(ftest))
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:6: UserWarning: `Model.fit_generator` is deprecated and w

Epoch 1/10
154/154 [==============================] - 35s 176ms/step - loss: 1.5479 - accuracy: 0.5302 - val_loss: 1.3259 - val_a
Epoch 2/10
154/154 [==============================] - 27s 174ms/step - loss: 0.6284 - accuracy: 0.7950 - val_loss: 0.7936 - val_a
Epoch 3/10
                          =========] - 30s 193ms/step - loss: 0.3648 - accuracy: 0.8883 - val_loss: 0.5895 - val_a
```

Saved successfully!

```
154/154 [==============================] - 27s 174ms/step - loss: 0.2853 - accuracy: 0.9141 - val_loss: 0.4523 - val_a
Epoch 5/10
154/154 [==============================] - 27s 174ms/step - loss: 0.2442 - accuracy: 0.9243 - val_loss: 0.4683 - val_a
Epoch 6/10
154/154 [==============================] - 26s 171ms/step - loss: 0.2142 - accuracy: 0.9339 - val_loss: 0.5106 - val_a
Epoch 7/10
154/154 [==============================] - 27s 172ms/step - loss: 0.1898 - accuracy: 0.9400 - val_loss: 0.4912 - val_a
Epoch 8/10
154/154 [==============================] - 27s 173ms/step - loss: 0.1736 - accuracy: 0.9469 - val_loss: 0.4330 - val_a
Epoch 9/10
154/154 [==============================] - 27s 173ms/step - loss: 0.1513 - accuracy: 0.9538 - val_loss: 0.5544 - val_a
Epoch 10/10
154/154 [==============================] - 27s 173ms/step - loss: 0.1351 - accuracy: 0.9572 - val_loss: 0.6018 - val_a
<keras.callbacks.History at 0x7fc46048ea10>
```

Saving The Model :

```
#Save Model
model.save('CAUDL.h5')
```

Testing The Model :

```
#Import req. Lib.
from tensorflow.keras.preprocessing import image
import numpy as np
```

```
#Testing No 1 :-
img = image.load_img('/content/data/test/Left Bundle Branch Block/fig_5910.png',target_size=(64,64)) #Reading image
f = image.img_to_array(img) #Convertinng image to array
f = np.expand_dims(f,axis=0) #Expanding dimensions
pred = np.argmax(model.predict(f)) #predicting higher propability index
                                   ormal','Premature Atrial Contraction','Premature Ventricular Contractions','Right Bundle
```

Saved successfully! ×

```
     1/1 [==============================] - 0s 112ms/step
     'Left Bundle Branch Block'
```

```
#Testing No 2 :-
img = image.load_img('/content/data/test/Normal/fig_2203.png',target_size=(64,64)) #Reading image
f = image.img_to_array(img) #Convertinng image to array
f = np.expand_dims(f,axis=0) #Expanding dimensions
pred = np.argmax(model.predict(f)) #predicting higher propability index
op = ['Left Bundle Branch Block','Normal','Premature Atrial Contraction','Premature Ventricular Contractions','Right Bundle
op[pred] #List indexing with output
```

```
     1/1 [==============================] - 0s 27ms/step
     'Normal'
```

```
#Testing No 3 :-
img = image.load_img('/content/data/test/Premature Atrial Contraction/fig_1383.png',target_size=(64,64)) #Reading image
f = image.img_to_array(img) #Convertinng image to array
f = np.expand_dims(f,axis=0) #Expanding dimensions
pred = np.argmax(model.predict(f)) #predicting higher propability index
op = ['Left Bundle Branch Block','Normal','Premature Atrial Contraction','Premature Ventricular Contractions','Right Bundle
op[pred] #List indexing with output
```

```
     1/1 [==============================] - 0s 15ms/step
     'Premature Atrial Contraction'
```

```
#Testing No 4 :-
img = image.load_img('/content/data/test/Premature Ventricular Contractions/VEBfig_1.png',target_size=(64,64)) #Reading imag
f = image.img_to_array(img) #Convertinng image to array
f = np.expand_dims(f,axis=0) #Expanding dimensions
pred = np.argmax(model.predict(f)) #predicting higher propability index
op = ['Left Bundle Branch Block','Normal','Premature Atrial Contraction','Premature Ventricular Contractions','Right Bundle
op[pred] #List indexing with output
```

```
     1/1 [==============================] - 0s 15ms/step
     'Ventricular Fibrillation'
```

Saved successfully! ×

...Testing No 4 showing a wrng result ❗

```
#Testing No 5 :-
img = image.load_img('/content/data/test/Right Bundle Branch Block/fig_100.png',target_size=(64,64)) #Reading image
f = image.img_to_array(img) #Convertinng image to array
f = np.expand_dims(f,axis=0) #Expanding dimensions
pred = np.argmax(model.predict(f)) #predicting higher propability index
op = ['Left Bundle Branch Block','Normal','Premature Atrial Contraction','Premature Ventricular Contractions','Right Bundle
op[pred] #List indexing with output
```

```
1/1 [==============================] - 0s 16ms/step
'Right Bundle Branch Block'
```

```
#Testing No 6 :-
img = image.load_img('/content/data/test/Ventricular Fibrillation/VFEfig_122.png',target_size=(64,64)) #Reading image
f = image.img_to_array(img) #Convertinng image to array
f = np.expand_dims(f,axis=0) #Expanding dimensions
pred = np.argmax(model.predict(f)) #predicting higher propability index
op = ['Left Bundle Branch Block','Normal','Premature Atrial Contraction','Premature Ventricular Contractions','Right Bundle
op[pred] #List indexing with output
```

```
1/1 [==============================] - 0s 15ms/step
'Ventricular Fibrillation'
```

Model Tuning:

```
from tensorflow.keras.callbacks import EarlyStopping, ReduceLROnPlateau
```

```
early_stop = EarlyStopping(monitor='val_accuracy',
                           patience=5)
```

```
lr = ReduceLROnPlateau(monitor='val_accuaracy',
                       factor=0.5,
                                    01)
```

Saved successfully!

```
callback = [early_stop,lr]

#Train model
model.fit_generator(ftrain,
                    steps_per_epoch=len(ftrain),
                    epochs=100,
                    callbacks=callback,
                    validation_data=ftest,
                    validation_steps=len(ftest))
```

```
Epoch 1/100
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:7: UserWarning: `Model.fit_generator` is deprecated and w
  import sys
154/154 [==============================] - ETA: 0s - loss: 0.1191 - accuracy: 0.9630WARNING:tensorflow:Learning rate r
154/154 [==============================] - 26s 172ms/step - loss: 0.1191 - accuracy: 0.9630 - val_loss: 0.6480 - val_a
Epoch 2/100
154/154 [==============================] - ETA: 0s - loss: 0.1066 - accuracy: 0.9679WARNING:tensorflow:Learning rate r
154/154 [==============================] - 26s 171ms/step - loss: 0.1066 - accuracy: 0.9679 - val_loss: 0.6173 - val_a
Epoch 3/100
154/154 [==============================] - ETA: 0s - loss: 0.0951 - accuracy: 0.9719WARNING:tensorflow:Learning rate r
154/154 [==============================] - 26s 172ms/step - loss: 0.0951 - accuracy: 0.9719 - val_loss: 0.8147 - val_a
Epoch 4/100
154/154 [==============================] - ETA: 0s - loss: 0.0756 - accuracy: 0.9774WARNING:tensorflow:Learning rate r
154/154 [==============================] - 26s 171ms/step - loss: 0.0756 - accuracy: 0.9774 - val_loss: 0.7134 - val_a
Epoch 5/100
154/154 [==============================] - ETA: 0s - loss: 0.0703 - accuracy: 0.9785WARNING:tensorflow:Learning rate r
154/154 [==============================] - 27s 173ms/step - loss: 0.0703 - accuracy: 0.9785 - val_loss: 0.8673 - val_a
Epoch 6/100
154/154 [==============================] - ETA: 0s - loss: 0.0583 - accuracy: 0.9814WARNING:tensorflow:Learning rate r
154/154 [==============================] - 26s 172ms/step - loss: 0.0583 - accuracy: 0.9814 - val_loss: 0.7834 - val_a
Epoch 7/100
154/154 [==============================] - ETA: 0s - loss: 0.0537 - accuracy: 0.9829WARNING:tensorflow:Learning rate r
154/154 [==============================] - 27s 172ms/step - loss: 0.0537 - accuracy: 0.9829 - val_loss: 0.8721 - val_a
Epoch 8/100
154/154 [==============================] - ETA: 0s - loss: 0.0474 - accuracy: 0.9865WARNING:tensorflow:Learning rate r
154/154 [==============================] - 28s 181ms/step - loss: 0.0474 - accuracy: 0.9865 - val_loss: 0.8234 - val_a
Epoch 9/100
154/154 [==============================] - ETA: 0s - loss: 0.0458 - accuracy: 0.9855WARNING:tensorflow:Learning rate r
                     =========] - 27s 173ms/step - loss: 0.0458 - accuracy: 0.9855 - val_loss: 0.8935 - val_a

                     =========] - ETA: 0s - loss: 0.0397 - accuracy: 0.9877WARNING:tensorflow:Learning rate r
```

Saved successfully! ×

```
154/154 [==============================] - 26s 171ms/step - loss: 0.0397 - accuracy: 0.9877 - val_loss: 0.7478 - val_a
Epoch 11/100
154/154 [==============================] - ETA: 0s - loss: 0.0367 - accuracy: 0.9892WARNING:tensorflow:Learning rate r
154/154 [==============================] - 26s 171ms/step - loss: 0.0367 - accuracy: 0.9892 - val_loss: 1.0439 - val_a
Epoch 12/100
154/154 [==============================] - ETA: 0s - loss: 0.0338 - accuracy: 0.9896WARNING:tensorflow:Learning rate r
154/154 [==============================] - 26s 170ms/step - loss: 0.0338 - accuracy: 0.9896 - val_loss: 0.9812 - val_a
Epoch 13/100
154/154 [==============================] - ETA: 0s - loss: 0.0299 - accuracy: 0.9902WARNING:tensorflow:Learning rate r
154/154 [==============================] - 26s 170ms/step - loss: 0.0299 - accuracy: 0.9902 - val_loss: 1.0120 - val_a
Epoch 14/100
154/154 [==============================] - ETA: 0s - loss: 0.0247 - accuracy: 0.9920WARNING:tensorflow:Learning rate r
154/154 [==============================] - 26s 171ms/step - loss: 0.0247 - accuracy: 0.9920 - val_loss: 0.9792 - val_a
Epoch 15/100
154/154 [==============================] - ETA: 0s - loss: 0.0326 - accuracy: 0.9896WARNING:tensorflow:Learning rate r
154/154 [==============================] - 26s 171ms/step - loss: 0.0326 - accuracy: 0.9896 - val_loss: 0.9333 - val_a
Epoch 16/100
154/154 [==============================] - ETA: 0s - loss: 0.0285 - accuracy: 0.9904WARNING:tensorflow:Learning rate r
154/154 [==============================] - 26s 172ms/step - loss: 0.0285 - accuracy: 0.9904 - val_loss: 1.0015 - val_a
Epoch 17/100
154/154 [==============================] - ETA: 0s - loss: 0.0273 - accuracy: 0.9916WARNING:tensorflow:Learning rate r
154/154 [==============================] - 28s 180ms/step - loss: 0.0273 - accuracy: 0.9916 - val_loss: 0.9774 - val_a
<keras.callbacks.History at 0x7fc460159c90>
```

```
#Testing No 1 :-
img = image.load_img('/content/data/test/Left Bundle Branch Block/fig_5898.png',target_size=(64,64)) #Reading image
f = image.img_to_array(img) #Convertinng image to array
f = np.expand_dims(f,axis=0) #Expanding dimensions
pred = np.argmax(model.predict(f)) #predicting higher propability index
op = ['Left Bundle Branch Block','Normal','Premature Atrial Contraction','Premature Ventricular Contractions','Right Bundle
op[pred] #List indexing with output
```

```
1/1 [==============================] - 0s 16ms/step
'Left Bundle Branch Block'
```

```
#Testing No 2 :-
/test/Normal/fig_2113.png',target_size=(64,64)) #Reading image
tinng image to array
```

Saved successfully! ×

```
f = np.expand_dims(f,axis=0) #Expanding dimensions
pred = np.argmax(model.predict(f)) #predicting higher propability index
op = ['Left Bundle Branch Block','Normal','Premature Atrial Contraction','Premature Ventricular Contractions','Right Bundle
op[pred] #List indexing with output
```

```
    1/1 [==============================] - 0s 21ms/step
    'Normal'
```

```
#Testing No 3 :-
img = image.load_img('/content/data/test/Premature Atrial Contraction/fig_100.png',target_size=(64,64)) #Reading image
f = image.img_to_array(img) #Convertinng image to array
f = np.expand_dims(f,axis=0) #Expanding dimensions
pred = np.argmax(model.predict(f)) #predicting higher propability index
op = ['Left Bundle Branch Block','Normal','Premature Atrial Contraction','Premature Ventricular Contractions','Right Bundle
op[pred] #List indexing with output
```

```
    1/1 [==============================] - 0s 17ms/step
    'Premature Atrial Contraction'
```

```
#Testing No 4 :-
img = image.load_img('/content/data/test/Premature Ventricular Contractions/fig_6090.png',target_size=(64,64)) #Reading imag
f = image.img_to_array(img) #Convertinng image to array
f = np.expand_dims(f,axis=0) #Expanding dimensions
pred = np.argmax(model.predict(f)) #predicting higher propability index
op = ['Left Bundle Branch Block','Normal','Premature Atrial Contraction','Premature Ventricular Contractions','Right Bundle
op[pred] #List indexing with output
```

```
    1/1 [==============================] - 0s 15ms/step
    'Premature Ventricular Contractions'
```

...Testing No 4 now shows the correct result ✅

```
#Testing No 5 :-
img = image.load_img('/content/data/test/Right Bundle Branch Block/fig_100.png',target_size=(64,64)) #Reading image
                                    tinng image to array
                                    ding dimensions
                                    #predicting higher propability index
```

Saved successfully! ×

```
op = ['Left Bundle Branch Block','Normal','Premature Atrial Contraction','Premature Ventricular Contractions','Right Bundle
op[pred] #List indexing with output

     1/1 [==============================] - 0s 17ms/step
     'Right Bundle Branch Block'
```

```
#Testing No 6 :-
img = image.load_img('/content/data/test/Ventricular Fibrillation/VFEfig_198.png',target_size=(64,64)) #Reading image
f = image.img_to_array(img) #Convertinng image to array
f = np.expand_dims(f,axis=0) #Expanding dimensions
pred = np.argmax(model.predict(f)) #predicting higher propability index
op = ['Left Bundle Branch Block','Normal','Premature Atrial Contraction','Premature Ventricular Contractions','Right Bundle
op[pred] #List indexing with output

     1/1 [==============================] - 0s 18ms/step
     'Ventricular Fibrillation'
```

Saving The Model :

```
#Save Model
model.save('CAUDL.h5')
```

```
!tar -zcvf CAUDL.tgz CAUDL.h5

     CAUDL.h5
```

IBM Cloud Deployment (Watson) :

```
!pip install watson-machine-learning-client
```

Saved successfully!

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Collecting watson-machine-learning-client
  Downloading watson_machine_learning_client-1.0.391-py3-none-any.whl (538 kB)
     |████████████████████████████████| 538 kB 14.7 MB/s
Requirement already satisfied: tabulate in /usr/local/lib/python3.7/dist-packages (from watson-machine-learning-client
Requirement already satisfied: tqdm in /usr/local/lib/python3.7/dist-packages (from watson-machine-learning-client) (4
Collecting boto3
  Downloading boto3-1.26.6-py3-none-any.whl (132 kB)
     |████████████████████████████████| 132 kB 72.6 MB/s
Collecting ibm-cos-sdk
  Downloading ibm-cos-sdk-2.12.0.tar.gz (55 kB)
     |████████████████████████████████| 55 kB 4.9 MB/s
Requirement already satisfied: certifi in /usr/local/lib/python3.7/dist-packages (from watson-machine-learning-client)
Requirement already satisfied: pandas in /usr/local/lib/python3.7/dist-packages (from watson-machine-learning-client)
Collecting lomond
  Downloading lomond-0.3.3-py2.py3-none-any.whl (35 kB)
Requirement already satisfied: requests in /usr/local/lib/python3.7/dist-packages (from watson-machine-learning-client
Requirement already satisfied: urllib3 in /usr/local/lib/python3.7/dist-packages (from watson-machine-learning-client)
Collecting jmespath<2.0.0,>=0.7.1
  Downloading jmespath-1.0.1-py3-none-any.whl (20 kB)
Collecting botocore<1.30.0,>=1.29.6
  Downloading botocore-1.29.6-py3-none-any.whl (9.9 MB)
     |████████████████████████████████| 9.9 MB 54.9 MB/s
Collecting s3transfer<0.7.0,>=0.6.0
  Downloading s3transfer-0.6.0-py3-none-any.whl (79 kB)
     |████████████████████████████████| 79 kB 9.4 MB/s
Requirement already satisfied: python-dateutil<3.0.0,>=2.1 in /usr/local/lib/python3.7/dist-packages (from botocore<1.
Collecting urllib3
  Downloading urllib3-1.26.12-py2.py3-none-any.whl (140 kB)
     |████████████████████████████████| 140 kB 73.9 MB/s
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.7/dist-packages (from python-dateutil<3.0.0,>=2.1->b
Collecting ibm-cos-sdk-core==2.12.0
  Downloading ibm-cos-sdk-core-2.12.0.tar.gz (956 kB)
     |████████████████████████████████| 956 kB 65.7 MB/s
Collecting ibm-cos-sdk-s3transfer==2.12.0
  Downloading ibm-cos-sdk-s3transfer-2.12.0.tar.gz (135 kB)
     |████████████████████████████████| 135 kB 64.7 MB/s
Collecting jmespath<2.0.0,>=0.7.1
  Downloading jmespath-0.10.0-py2.py3-none-any.whl (24 kB)

                              py3-none-any.whl (62 kB)
                              ████| 62 kB 2.0 MB/s
```

Saved successfully!

```
Requirement already satisfied: charset-normalizer<3,>=2 in /usr/local/lib/python3.7/dist-packages (from requests->wats
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.7/dist-packages (from requests->watson-machine-l
Requirement already satisfied: numpy>=1.17.3 in /usr/local/lib/python3.7/dist-packages (from pandas->watson-machine-le
Requirement already satisfied: pytz>=2017.3 in /usr/local/lib/python3.7/dist-packages (from pandas->watson-machine-lea
Building wheels for collected packages: ibm-cos-sdk, ibm-cos-sdk-core, ibm-cos-sdk-s3transfer
  Building wheel for ibm-cos-sdk (setup.py) ... done
  Created wheel for ibm-cos-sdk: filename=ibm_cos_sdk-2.12.0-py3-none-any.whl size=73929 sha256=336a5fe3828bb71fda1e6d
  Stored in directory: /root/.cache/pip/wheels/ec/94/29/2b57327cf00664b6614304f7958abd29d77ea0e5bbece2ea57
  Building wheel for ibm-cos-sdk-core (setup.py) ... done
  Created wheel for ibm-cos-sdk-core: filename=ibm_cos_sdk_core-2.12.0-py3-none-any.whl size=562962 sha256=5076e19324b
  Stored in directory: /root/.cache/pip/wheels/64/56/fb/5cd6f4f40406c828a5289b95b2752a4d142a9afb359244ed8d
  Building wheel for ibm-cos-sdk-s3transfer (setup.py) ... done
  Created wheel for ibm-cos-sdk-s3transfer: filename=ibm_cos_sdk_s3transfer-2.12.0-py3-none-any.whl size=89779 sha256=
  Stored in directory: /root/.cache/pip/wheels/57/79/6a/ffe3370ed7ebc00604f9f76766e1e0348dcdcad2b2e32df9e1
Successfully built ibm-cos-sdk ibm-cos-sdk-core ibm-cos-sdk-s3transfer
Installing collected packages: urllib3, requests, jmespath, ibm-cos-sdk-core, botocore, s3transfer, ibm-cos-sdk-s3tran
  Attempting uninstall: urllib3
    Found existing installation: urllib3 1.24.3
    Uninstalling urllib3-1.24.3:
      Successfully uninstalled urllib3-1.24.3
  Attempting uninstall: requests
    Found existing installation: requests 2.23.0
    Uninstalling requests-2.23.0:
      Successfully uninstalled requests-2.23.0
Successfully installed boto3-1.26.6 botocore-1.29.6 ibm-cos-sdk-2.12.0 ibm-cos-sdk-core-2.12.0 ibm-cos-sdk-s3transfer-
WARNING: The following packages were previously imported in this runtime:
  [requests,urllib3]
```

```
!pip install ibm_watson_machine_learning
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: ibm_watson_machine_learning in /usr/local/lib/python3.7/dist-packages (1.0.257)
Requirement already satisfied: pandas<1.5.0,>=0.24.2 in /usr/local/lib/python3.7/dist-packages (from ibm_watson_machin
Requirement already satisfied: tabulate in /usr/local/lib/python3.7/dist-packages (from ibm_watson_machine_learning) (
Requirement already satisfied: requests in /usr/local/lib/python3.7/dist-packages (from ibm_watson_machine_learning) (
Requirement already satisfied: ibm-cos-sdk==2.7.* in /usr/local/lib/python3.7/dist-packages (from ibm_watson_machine_l
Requirement already satisfied: urllib3 in /usr/local/lib/python3.7/dist-packages (from ibm_watson_machine_learning) (1
Requirement already satisfied: lomond in /usr/local/lib/python3.7/dist-packages (from ibm_watson_machine_learning) (0.
Requirement already satisfied: packaging in /usr/local/lib/python3.7/dist-packages (from ibm_watson_machine_learning)
Requirement already satisfied: importlib-metadata in /usr/local/lib/python3.7/dist-packages (from ibm_watson_machine_l
                               certifi in /usr/local/lib/python3.7/dist-packages (from ibm_watson_machine_learning) (2
                               ibm-cos-sdk-s3transfer==2.7.0 in /usr/local/lib/python3.7/dist-packages (from ibm-cos-s
                               ibm-cos-sdk-core==2.7.0 in /usr/local/lib/python3.7/dist-packages (from ibm-cos-sdk==2.
```

Saved successfully!

```
Requirement already satisfied: jmespath<1.0.0,>=0.7.1 in /usr/local/lib/python3.7/dist-packages (from ibm-cos-sdk==2.7
Requirement already satisfied: python-dateutil<3.0.0,>=2.1 in /usr/local/lib/python3.7/dist-packages (from ibm-cos-sdk
Requirement already satisfied: docutils<0.16,>=0.10 in /usr/local/lib/python3.7/dist-packages (from ibm-cos-sdk-core==
Requirement already satisfied: pytz>=2017.3 in /usr/local/lib/python3.7/dist-packages (from pandas<1.5.0,>=0.24.2->ibm
Requirement already satisfied: numpy>=1.17.3 in /usr/local/lib/python3.7/dist-packages (from pandas<1.5.0,>=0.24.2->ib
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.7/dist-packages (from python-dateutil<3.0.0,>=2.1->i
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.7/dist-packages (from requests->ibm_watson_machi
Requirement already satisfied: charset-normalizer<3,>=2 in /usr/local/lib/python3.7/dist-packages (from requests->ibm_
Requirement already satisfied: typing-extensions>=3.6.4 in /usr/local/lib/python3.7/dist-packages (from importlib-meta
Requirement already satisfied: zipp>=0.5 in /usr/local/lib/python3.7/dist-packages (from importlib-metadata->ibm_watso
Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in /usr/local/lib/python3.7/dist-packages (from packaging->ibm
```

```
from ibm_watson_machine_learning import APIClient

wml_credentials = {
    "url":"https://us-south.ml.cloud.ibm.com",
    "apikey":"t_STJIQGvNZW_S4q3DieWdLKEkRxR0X5jPjEx_n1Tij2"
}
client = APIClient(wml_credentials)
```

```
Python 3.7 and 3.8 frameworks are deprecated and will be removed in a future release. Use Python 3.9 framework instead
```

```
client
```

```
<ibm_watson_machine_learning.client.APIClient at 0x7fc40c6411d0>
```

```
client.spaces.get_details()
```

```
{'resources': [{'entity': {'compute': [{'crn': 'crn:v1:bluemix:public:pm-20:us-
south:a/7faba0a0dde544c0ab273d89d887bacd:a2c465c1-7b97-4131-89fe-21a4b627fad7::',
      'guid': 'a2c465c1-7b97-4131-89fe-21a4b627fad7',
      'name': 'Watson Machine Learning-pc',
      'type': 'machine_learning'}],
    'description': 'Classification of Arrhythmia by Using Deep Learning with 2-D ECG Spectral Image Representation
```

Saved successfully!

```
    'scope': {'bss_account_id': '7faba0a0dde544c0ab273d89d887bacd'},
    'stage': {'production': False},
    'status': {'state': 'active'},
    'storage': {'properties': {'bucket_name': '022b2e9c-7b1d-4678-a64d-584020cdb695',
      'bucket_region': 'us-south',
      'credentials': {'admin': {'access_key_id': '6e856b4f7c244ccab51cde1937a14da7',
        'api_key': 'AVX57CJHwAmF0029IDK1fO2ajpnC70iQIWALAibh4zDb',
        'secret_access_key': '1b92b72d1b058d78c91860cd9dac0481400f3899c583b4b3',
        'service_id': 'ServiceId-341f55fa-2766-4df1-9c9c-e195a2112bd3'},
       'editor': {'access_key_id': '580c845ecf864d2a953adb87504b62fb',
        'api_key': 'uZeAQxclP0ylFI72t9YWNkCpm-Rsbp0K6N89QEDMghfC',
        'resource_key_crn': 'crn:v1:bluemix:public:cloud-object-
storage:global:a/7faba0a0dde544c0ab273d89d887bacd:e3b0ff82-767f-4473-8dce-03d66a3b920b::',
        'secret_access_key': '4531008e70466ec4c8ddf279f3eb656754679cf8e96140a6',
        'service_id': 'ServiceId-c11a71d7-178e-4eb2-9258-124bd43a7ff0'},
       'viewer': {'access_key_id': '0855d6c3d9014464a0d9e7a3d78f337e',
        'api_key': 'G9MIIj3yNY7ihX9l9BxCoo5ZNfbPER_TywJeunW_zV9d',
        'resource_key_crn': 'crn:v1:bluemix:public:cloud-object-
storage:global:a/7faba0a0dde544c0ab273d89d887bacd:e3b0ff82-767f-4473-8dce-03d66a3b920b::',
        'secret_access_key': '055c7863c1ae98232ebb39eaeb1f158382e36f6f757cc7c0',
        'service_id': 'ServiceId-3fd62183-8ca9-418f-bfe2-c048de1e6d45'}},
      'endpoint_url': 'https://s3.us-south.cloud-object-storage.appdomain.cloud',
      'guid': 'e3b0ff82-767f-4473-8dce-03d66a3b920b',
      'resource_crn': 'crn:v1:bluemix:public:cloud-object-storage:global:a/7faba0a0dde544c0ab273d89d887bacd:e3b0ff82-
767f-4473-8dce-03d66a3b920b::'},
     'type': 'bmcos_object_storage'}},
   'metadata': {'created_at': '2022-11-10T04:37:44.706Z',
    'creator_id': 'IBMid-662003X5JS',
    'id': '8cb20b68-2b1b-4080-b28b-d2e165f03ac8',
    'updated_at': '2022-11-10T04:38:07.733Z',
    'url': '/v2/spaces/8cb20b68-2b1b-4080-b28b-d2e165f03ac8'}}]}
```

```
client.spaces.list()
```

```
Note: 'limit' is not provided. Only first 50 records will be displayed if the number of records exceed 50
------------------------------------  ------------  ------------------------
ID                                    NAME          CREATED
8cb20b68-2b1b-4080-b28b-d2e165f03ac8  CAUDL_Deploy  2022-11-10T04:37:44.706Z
                              ------  ------------  ------------------------
```

Saved successfully! ✕

```
space_uid ="8cb20b68-2b1b-4080-b28b-d2e165f03ac8"
space_uid
```

```
'8cb20b68-2b1b-4080-b28b-d2e165f03ac8'
```

```
client.set.default_space(space_uid )
```

```
'SUCCESS'
```

```
client.software_specifications.list()
```

```
-----------------------------  ------------------------------------  ----
NAME                           ASSET_ID                              TYPE
default_py3.6                  0062b8c9-8b7d-44a0-a9b9-46c416adcbd9  base
kernel-spark3.2-scala2.12      020d69ce-7ac1-5e68-ac1a-31189867356a  base
pytorch-onnx_1.3-py3.7-edt     069ea134-3346-5748-b513-49120e15d288  base
scikit-learn_0.20-py3.6        09c5a1d0-9c1e-4473-a344-eb7b665ff687  base
spark-mllib_3.0-scala_2.12     09f4cff0-90a7-5899-b9ed-1ef348aebdee  base
pytorch-onnx_rt22.1-py3.9      0b848dd4-e681-5599-be41-b5f6fccc6471  base
ai-function_0.1-py3.6          0cdb0f1e-5376-4f4d-92dd-da3b69aa9bda  base
shiny-r3.6                     0e6e79df-875e-4f24-8ae9-62dcc2148306  base
tensorflow_2.4-py3.7-horovod   1092590a-307d-563d-9b62-4eb7d64b3f22  base
pytorch_1.1-py3.6              10ac12d6-6b30-4ccd-8392-3e922c096a92  base
tensorflow_1.15-py3.6-ddl      111e41b3-de2d-5422-a4d6-bf776828c4b7  base
runtime-22.1-py3.9             12b83a17-24d8-5082-900f-0ab31fbfd3cb  base
scikit-learn_0.22-py3.6        154010fa-5b3b-4ac1-82af-4d5ee5abbc85  base
default_r3.6                   1b70aec3-ab34-4b87-8aa0-a4a3c8296a36  base
pytorch-onnx_1.3-py3.6         1bc6029a-cc97-56da-b8e0-39c3880dbbe7  base
kernel-spark3.3-r3.6           1c9e5454-f216-59dd-a20e-474a5cdf5988  base
pytorch-onnx_rt22.1-py3.9-edt  1d362186-7ad5-5b59-8b6c-9d0880bde37f  base
tensorflow_2.1-py3.6           1eb25b84-d6ed-5dde-b6a5-3fbdf1665666  base
spark-mllib_3.2                20047f72-0a98-58c7-9ff5-a77b012eb8f5  base
tensorflow_2.4-py3.8-horovod   217c16f6-178f-56bf-824a-b19f20564c49  base
runtime-22.1-py3.9-cuda        26215f05-08c3-5a41-a1b0-da66306ce658  base
do_py3.8                       295addb5-9ef9-547e-9bf4-92ae3563e720  base
autoai-ts_3.8-py3.8            2aa0c932-798f-5ae9-abd6-15e0c2402fb5  base
tensorflow_1.15-py3.6          2b73a275-7cbf-420b-a912-eae7f436e0bc  base
kernel-spark3.3-py3.9          2b7961e2-e3b1-5a8c-a491-482c8368839a  base
                               2c8ef57d-2687-4b7d-acce-01f94976dac1  base
                               2e51f700-bca0-4b0d-88dc-5c6791338875  base
                               32983cea-3f32-4400-8965-dde874a8d67e  base
```

Saved successfully! ×

```
spark-mllib_3.0-py37           36507ebe-8770-55ba-ab2a-eafe787600e9  base
spark-mllib_2.4                390d21f8-e58b-4fac-9c55-d7ceda621326  base
xgboost_0.82-py3.6             39e31acd-5f30-41dc-ae44-60233c80306e  base
pytorch-onnx_1.2-py3.6-edt     40589d0e-7019-4e28-8daa-fb03b6f4fe12  base
default_r36py38                41c247d3-45f8-5a71-b065-8580229facf0  base
autoai-ts_rt22.1-py3.9         4269d26e-07ba-5d40-8f66-2d495b0c71f7  base
autoai-obm_3.0                 42b92e18-d9ab-567f-988a-4240ba1ed5f7  base
pmml-3.0_4.3                   493bcb95-16f1-5bc5-bee8-81b8af80e9c7  base
spark-mllib_2.4-r_3.6          49403dff-92e9-4c87-a3d7-a42d0021c095  base
xgboost_0.90-py3.6             4ff8d6c2-1343-4c18-85e1-689c965304d3  base
pytorch-onnx_1.1-py3.6         50f95b2a-bc16-43bb-bc94-b0bed208c60b  base
autoai-ts_3.9-py3.8            52c57136-80fa-572e-8728-a5e7cbb42cde  base
spark-mllib_2.4-scala_2.11     55a70f99-7320-4be5-9fb9-9edb5a443af5  base
spark-mllib_3.0                5c1b0ca2-4977-5c2e-9439-ffd44ea8ffe9  base
autoai-obm_2.0                 5c2e37fa-80b8-5e77-840f-d912469614ee  base
spss-modeler_18.1              5c3cad7e-507f-4b2a-a9a3-ab53a21dee8b  base
cuda-py3.8                     5d3232bf-c86b-5df4-a2cd-7bb870a1cd4e  base
autoai-kb_3.1-py3.7            632d4b22-10aa-5180-88f0-f52dfb6444d7  base
pytorch-onnx_1.7-py3.8         634d3cdc-b562-5bf9-a2d4-ea90a478456b  base
spark-mllib_2.3-r_3.6          6586b9e3-ccd6-4f92-900f-0f8cb2bd6f0c  base
tensorflow_2.4-py3.7           65e171d7-72d1-55d9-8ebb-f813d620c9bb  base
spss-modeler_18.2              687eddc9-028a-4117-b9dd-e57b36f1efa5  base
-----------------------------  ------------------------------------  ----
Note: Only first 50 records were displayed. To display more use 'limit' parameter.
```

```
software_space_uid = client.software_specifications.get_uid_by_name("tensorflow_rt22.1-py3.9")
software_space_uid
```

```
'acd9c798-6974-5d2f-a657-ce06e986df4d'
```

```
model_details = client.repository.store_model(model="/content/CAUDL.tgz", meta_props={
    client.repository.ModelMetaNames.NAME:"CAUDL Model",
    client.repository.ModelMetaNames.TYPE:"tensorflow_2.7",
    client.repository.ModelMetaNames.SOFTWARE_SPEC_UID:software_space_uid
})
```

Saved successfully! ✕

```
oftware_specs': [],
          'software_spec': {'id': 'acd9c798-6974-5d2f-a657-ce06e986df4d',
           'name': 'tensorflow_rt22.1-py3.9'},
          'type': 'tensorflow_2.7'},
         'metadata': {'created_at': '2022-11-10T07:57:58.856Z',
          'id': '09cf5e5e-0210-4ba0-a675-9e899b2a62c2',
          'modified_at': '2022-11-10T07:58:15.333Z',
          'name': 'CAUDL Model',
          'owner': 'IBMid-662003X5JS',
          'resource_key': 'f907a1c2-19cc-43c4-ba52-a676e90d1034',
          'space_id': '8cb20b68-2b1b-4080-b28b-d2e165f03ac8'},
         'system': {'warnings': []}}
```

If Want To Get Model After Sometime / Days :

```
model_id = client.repository.get_model_id(model_details)
model_id
```

```
     '09cf5e5e-0210-4ba0-a675-9e899b2a62c2'
```

Downloading Model Again :

```
client.repository.download(model_id,"CAUDL_IBM_Model.tgz")
```

```
     Successfully saved model content to file: 'CAUDL_IBM_Model.tgz'
     '/content/CAUDL_IBM_Model.tgz'
```

Saved successfully! ×

Colab paid products - Cancel contracts here

✓ 22s completed at 1:34 PM ● ✕

Could not connect to the reCAPTCHA service. Please check your internet connection and reload to get a reCAPTCHA challenge.

Saved successfully! ✕