IBM Project Name: Real-Time Communication System Powered by AI for Specially Abled

TEAM ID: PNT2022TMID19449

IBM WATSON STUDIO DEPLOYMENT CODE

1.]INSTALLING THE KERAS ,INSTALLING THE TENSORFLOW

In [97]:
```
!pip install Keras==2.2.4
!pip install tensorflow==2.7
```

```
Requirement already satisfied: Keras==2.2.4 in /opt/conda/envs/Python-3.9/l
ib/python3.9/site-packages (2.2.4)
Requirement already satisfied: h5py in /opt/conda/envs/Python-3.9/lib/pytho
n3.9/site-packages (from Keras==2.2.4) (3.2.1)
Requirement already satisfied: keras-preprocessing>=1.0.5 in /opt/conda/env
s/Python-3.9/lib/python3.9/site-packages (from Keras==2.2.4) (1.1.2)
Requirement already satisfied: numpy>=1.9.1 in /opt/conda/envs/Python-3.9/l
ib/python3.9/site-packages (from Keras==2.2.4) (1.20.3)
Requirement already satisfied: pyyaml in /opt/conda/envs/Python-3.9/lib/pyt
hon3.9/site-packages (from Keras==2.2.4) (5.4.1)
Requirement already satisfied: keras-applications>=1.0.6 in /opt/conda/envs
/Python-3.9/lib/python3.9/site-packages (from Keras==2.2.4) (1.0.8)
Requirement already satisfied: scipy>=0.14 in /opt/conda/envs/Python-3.9/li
b/python3.9/site-packages (from Keras==2.2.4) (1.7.3)
Requirement already satisfied: six>=1.9.0 in /opt/conda/envs/Python-3.9/lib
/python3.9/site-packages (from Keras==2.2.4) (1.15.0)
```

2.]IMPORTING LIBRARIES TO BUILD MODEL.

In [ ]:
```
#library to train the model
import keras
import tensorflow
```

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

3.]IMPORTING LIBRARIES FOR IMAGE AUGMENTATION.

In [99]:

```
#image augmentation
from tensorflow.keras.preprocessing.image import ImageDataGenerator
train_datagen=ImageDataGenerator(rescale=1./255,zoom_range=0.2,shear_range=
0.2,horizontal_flip=True,vertical_flip=False)
test_datagen=ImageDataGenerator(rescale=1./255)
```

4.]ADDING STREAMING_BODY_OBJECT FOR DATASET.ZIP

In [100]:

```
import os, types
import pandas as pd
from botocore.client import Config
import ibm_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.
cos_client = ibm_boto3.client(service_name='s3',
    ibm_api_key_id='aqprHZFuH38ECUn869hHk4qyvS_iKJfrZAWUJJQ-mQKx',
    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')

bucket = 'realtimecommunicationforspecially-donotdelete-pr-rfqndcvwgch6fu'
object_key = 'Dataset.zip'

streaming_body_4 = cos_client.get_object(Bucket=bucket,
Key=object_key)['Body']

# Your 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/
```

In[101]:

```
ls
```

```
Dataset/  test_set/  training_set/
```

5.]UNZIPPING THE DATASET

In [102]:

```
from io import BytesIO
import zipfile
unzip=zipfile.ZipFile(BytesIO(streaming_body_4.read()),'r')
file_paths=unzip.namelist()
for path in file_paths:
    unzip.extract(path)
```

In [103]:

```
pwd
```

Out[103]:

```
'/home/wsuser/work/Dataset'
```

In [104]:

```
#checking that the dataset is there are not
import os
filenamer = os.listdir('/home/wsuser/work/Dataset/training_set')
```

6.]TRAINING AND TESTING IMAGES UNDER CLASSES

In [105]:

```
x_train=train_datagen.flow_from_directory("/home/wsuser/work/Dataset/trainin
ng_set",target_size=(64,64),class_mode="categorical",batch_size=25)
```

```
Found 15750 images belonging to 9 classes.
```

In [106]:

```
x_test=test_datagen.flow_from_directory("/home/wsuser/work/Dataset/test_set
",target_size=(64,64),
class_mode='categorical' , batch_size=25)
```

```
Found 2250 images belonging to 9 classes.
```

7.]TOTAL CLASSES UNDER TRAINING AND TESTING.

In [107]:

```
x_train.class_indices
```

Out[107]:

```
{'A': 0, 'B': 1, 'C': 2, 'D': 3, 'E': 4, 'F': 5, 'G': 6, 'H': 7, 'I': 8}
```

In [108]:

```
x_test.class_indices
```

Out[108]:
```
{'A': 0, 'B': 1, 'C': 2, 'D': 3, 'E': 4, 'F': 5, 'G': 6, 'H': 7, 'I': 8}
```
In [109]:
```
train_datagen=ImageDataGenerator(rescale=1./255,zoom_range=0.2,horizontal_f
lip=True,vertical_flip=False)
```
In [110]:
```
test_datagen=ImageDataGenerator(rescale=1./255)
```

8.]MODEL BUILDING USING CNN

In [111]:
```
model=Sequential()
```
In [112]:
```
model.add(Convolution2D(32,(3,3),input_shape=(64,64,3),activation='relu'))
```
In [113]:
```
model.add(MaxPooling2D(pool_size=(2,2)))
```
In [114]:
```
model.add(Flatten())
```
In [115]:
```
model.summary()
```
```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_1 (Conv2D)            (None, 62, 62, 32)        896
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 31, 31, 32)        0
_________________________________________________________________
flatten_1 (Flatten)          (None, 30752)             0
=================================================================
Total params: 896
Trainable params: 896
Non-trainable params: 0
_________________________________________________________________
```

9.]ADDING LAYERS FOR MODEL TRAINING.

HIDDEN LAYERS

In [117]:
```
model.add(Dense(units = 300, activation='relu'))
#model.add(Dense(unit = 150,init = "uniform" activation='softmax'))
```

OUTPUT LAYERS

In [118]:
```
model.add(Dense(units = 5, activation='softmax'))
```

10.]OPTIMIZING THE MODEL

In [119]:
```
model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['ac
curacy'])
```

In [120]:
```
len(x_train)
```

Out[120]:
```
630
```

In [121]:
```
len(x_test)
```

Out[121]:
```
90
```

11.]FITTING THE MODEL

In [125]:
```
#model.fit_generator(x_train,steps_per_epoch=len(x_train),validation_data=x
_test,validation_steps=len(x_test),epochs=10)
# Fitting the Model Generator
model.fit_generator(x_train,steps_per_epoch=630,epochs=1,validation_data=x_
test,validation_steps=90)
#model.fit(x_train, epochs=100, verbose=1)
```

```
---------------------------------------------------------------------------
InvalidArgumentError                      Traceback (most recent call last)
/tmp/wsuser/ipykernel_164/1479672656.py in
      1 #model.fit_generator(x_train,steps_per_epoch=len(x_train),validatio
n_data=x_test,validation_steps=len(x_test),epochs=10)
      2 # Fitting the Model Generator
----> 3 model.fit_generator(x_train,steps_per_epoch=630,epochs=1,validation
_data=x_test,validation_steps=90)
      4 #model.fit(x_train, epochs=100, verbose=1)

/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/tensorflow/python/ke
ras/engine/training.py in fit_generator(self, generator, steps_per_epoch, e
pochs, verbose, callbacks, validation_data, validation_steps, validation_fr
eq, class_weight, max_queue_size, workers, use_multiprocessing, shuffle, in
itial_epoch)
   1964                    'will be removed in a future version. '
   1965                    'Please use `Model.fit`, which supports generator
s.')
-> 1966     return self.fit(
   1967         generator,
   1968         steps_per_epoch=steps_per_epoch,

/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/tensorflow/python/ke
ras/engine/training.py in fit(self, x, y, batch_size, epochs, verbose, call
backs, validation_split, validation_data, shuffle, class_weight, sample_wei
```

```
ght, initial_epoch, steps_per_epoch, validation_steps, validation_batch_siz
e, validation_freq, max_queue_size, workers, use_multiprocessing)
   1187                 _r=1):
   1188               callbacks.on_train_batch_begin(step)
-> 1189               tmp_logs = self.train_function(iterator)
   1190               if data_handler.should_sync:
   1191                 context.async_wait()

/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/tensorflow/python/ut
il/traceback_utils.py in error_handler(*args, **kwargs)
    151     except Exception as e:
    152       filtered_tb = _process_traceback_frames(e.__traceback__)
--> 153       raise e.with_traceback(filtered_tb) from None
    154     finally:
    155       del filtered_tb

/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/tensorflow/python/ea
ger/execute.py in quick_execute(op_name, num_outputs, inputs, attrs, ctx, n
ame)
     56   try:
     57     ctx.ensure_initialized()
---> 58     tensors = pywrap_tfe.TFE_Py_Execute(ctx._handle, device_name, o
p_name,
     59                                         inputs, attrs, num_outputs)
     60   except core._NotOkStatusException as e:
InvalidArgumentError:  logits and labels must be broadcastable: logits_size
=[25,5] labels_size=[25,9]
         [[node categorical_crossentropy/softmax_cross_entropy_with_logits
 (defined at /opt/conda/envs/Python-3.9/lib/python3.9/site-packages/tensorf
low/python/keras/backend.py:4889)
]] [Op:__inference_train_function_2383]

Errors may have originated from an input operation.
Input Source operations connected to node categorical_crossentropy/softmax_
cross_entropy_with_logits:
In[0] categorical_crossentropy/softmax_cross_entropy_with_logits/Reshape:

In[1] categorical_crossentropy/softmax_cross_entropy_with_logits/Reshape_1:

Operation defined at: (most recent call last)
>>>   File "/opt/conda/envs/Python-3.9/lib/python3.9/runpy.py", line 197, i
n _run_module_as_main
>>>     return _run_code(code, main_globals, None,
>>>
>>>   File "/opt/conda/envs/Python-3.9/lib/python3.9/runpy.py", line 87, in
_run_code
>>>     exec(code, run_globals)
>>>
>>>   File "/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/ipykerne
l/__main__.py", line 3, in
>>>     app.launch_new_instance()
>>>
>>>   File "/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/traitlet
s/config/application.py", line 846, in launch_instance
>>>     app.start()
>>>
```

```
>>>   File "/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/ipykerne
l/kernelapp.py", line 677, in start
>>>     self.io_loop.start()
>>>
>>>   File "/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/tornado/
platform/asyncio.py", line 199, in start
>>>     self.asyncio_loop.run_forever()
>>>
>>>   File "/opt/conda/envs/Python-3.9/lib/python3.9/asyncio/base_events.py
", line 601, in run_forever
>>>     self._run_once()
>>>
>>>   File "/opt/conda/envs/Python-3.9/lib/python3.9/asyncio/base_events.py
", line 1905, in _run_once
>>>     handle._run()
>>>
>>>   File "/opt/conda/envs/Python-3.9/lib/python3.9/asyncio/events.py", li
ne 80, in _run
>>>     self._context.run(self._callback, *self._args)
>>>
>>>   File "/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/ipykerne
l/kernelbase.py", line 457, in dispatch_queue
>>>     await self.process_one()
>>>
>>>   File "/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/ipykerne
l/kernelbase.py", line 446, in process_one
>>>     await dispatch(*args)
>>>
>>>   File "/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/ipykerne
l/kernelbase.py", line 353, in dispatch_shell
>>>     await result
>>>
>>>   File "/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/ipykerne
l/kernelbase.py", line 648, in execute_request
>>>     reply_content = await reply_content
>>>
>>>   File "/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/ipykerne
l/ipkernel.py", line 353, in do_execute
>>>     res = shell.run_cell(code, store_history=store_history, silent=sile
nt)
>>>
>>>   File "/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/ipykerne
l/zmqshell.py", line 533, in run_cell
>>>     return super(ZMQInteractiveShell, self).run_cell(*args, **kwargs)
>>>
>>>   File "/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/IPython/
core/interactiveshell.py", line 2914, in run_cell
>>>     result = self._run_cell(
>>>
>>>   File "/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/IPython/
core/interactiveshell.py", line 2960, in _run_cell
>>>     return runner(coro)
>>>
>>>   File "/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/IPython/
core/async_helpers.py", line 78, in _pseudo_sync_runner
>>>     coro.send(None)
>>>
```

```
>>>   File "/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/IPython/
core/interactiveshell.py", line 3185, in run_cell_async
>>>     has_raised = await self.run_ast_nodes(code_ast.body, cell_name,
>>>
>>>   File "/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/IPython/
core/interactiveshell.py", line 3377, in run_ast_nodes
>>>     if (await self.run_code(code, result,  async_=asy)):
>>>
>>>   File "/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/IPython/
core/interactiveshell.py", line 3457, in run_code
>>>     exec(code_obj, self.user_global_ns, self.user_ns)
>>>
>>>   File "/tmp/wsuser/ipykernel_164/3808038373.py", line 3, in
>>>     model.fit_generator(x_train,steps_per_epoch=630,epochs=1,validation
_data=x_test,validation_steps=90)
>>>
>>>   File "/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/tensorfl
ow/python/keras/engine/training.py", line 1966, in fit_generator
>>>     return self.fit(
>>>
>>>   File "/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/tensorfl
ow/python/keras/engine/training.py", line 1189, in fit
>>>     tmp_logs = self.train_function(iterator)
>>>
>>>   File "/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/tensorfl
ow/python/keras/engine/training.py", line 859, in train_function
>>>     return step_function(self, iterator)
>>>
>>>   File "/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/tensorfl
ow/python/keras/engine/training.py", line 849, in step_function
>>>     outputs = model.distribute_strategy.run(run_step, args=(data,))
>>>
>>>   File "/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/tensorfl
ow/python/keras/engine/training.py", line 842, in run_step
>>>     outputs = model.train_step(data)
>>>
>>>   File "/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/tensorfl
ow/python/keras/engine/training.py", line 800, in train_step
>>>     loss = self.compiled_loss(
>>>
>>>   File "/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/tensorfl
ow/python/keras/engine/compile_utils.py", line 204, in __call__
>>>     loss_value = loss_obj(y_t, y_p, sample_weight=sw)
>>>
>>>   File "/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/tensorfl
ow/python/keras/losses.py", line 155, in __call__
>>>     losses = call_fn(y_true, y_pred)
>>>
>>>   File "/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/tensorfl
ow/python/keras/losses.py", line 259, in call
>>>     return ag_fn(y_true, y_pred, **self._fn_kwargs)
>>>
>>>   File "/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/tensorfl
ow/python/keras/losses.py", line 1679, in categorical_crossentropy
>>>     return backend.categorical_crossentropy(
>>>
```

```
>>>   File "/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/tensorfl
ow/python/keras/backend.py", line 4889, in categorical_crossentropy
>>>     return nn.softmax_cross_entropy_with_logits_v2(
>>>
```

12.]SAVING THE MODEL

In [126]:
```
ls
```

```
Dataset/  test_set/  training_set/
```

In [127]:
```
pwd
```

Out[127]:
```
'/home/wsuser/work/Dataset'
```

In [128]:
```
model.save('Dataset.h5')
```

13.]CONVERTING ZIP FILE TO TAR FILE FOR LOCAL USE.

In [134]:
```
#converting the model to tar
!tar -zcvf image.Classification.model_new.tgz Dataset.h5
```

```
Dataset.h5
```

In [135]:
```
ls -1
```

```
Dataset/
Dataset.h5
image.Classification.model_new.tgz
test_set/
training_set/
```

14.]INSTALLING WATSON MACHINE LEARNING CLIENT SOFTWARE

In [137]:
```
#installing the machine learning repository
!pip install watson_machine_learning_client --upgrade
```

```
Collecting watson_machine_learning_client
  Downloading watson_machine_learning_client-1.0.391-py3-none-any.whl (538
kB)
     |████████████████████████████████| 538 kB 23.9 MB/s eta 0:00:01
Requirement already satisfied: pandas in /opt/conda/envs/Python-3.9/lib/pyt
hon3.9/site-packages (from watson_machine_learning_client) (1.3.4)
Requirement already satisfied: lomond in /opt/conda/envs/Python-3.9/lib/pyt
hon3.9/site-packages (from watson_machine_learning_client) (0.3.3)
```

```
Requirement already satisfied: urllib3 in /opt/conda/envs/Python-3.9/lib/py
thon3.9/site-packages (from watson_machine_learning_client) (1.26.7)
Requirement already satisfied: requests in /opt/conda/envs/Python-3.9/lib/p
ython3.9/site-packages (from watson_machine_learning_client) (2.26.0)
Requirement already satisfied: certifi in /opt/conda/envs/Python-3.9/lib/py
thon3.9/site-packages (from watson_machine_learning_client) (2022.9.24)
Requirement already satisfied: tqdm in /opt/conda/envs/Python-3.9/lib/pytho
n3.9/site-packages (from watson_machine_learning_client) (4.62.3)
Requirement already satisfied: boto3 in /opt/conda/envs/Python-3.9/lib/pyth
on3.9/site-packages (from watson_machine_learning_client) (1.18.21)
Requirement already satisfied: tabulate in /opt/conda/envs/Python-3.9/lib/p
ython3.9/site-packages (from watson_machine_learning_client) (0.8.9)
Requirement already satisfied: ibm-cos-sdk in /opt/conda/envs/Python-3.9/li
b/python3.9/site-packages (from watson_machine_learning_client) (2.11.0)
Requirement already satisfied: s3transfer<0.6.0,>=0.5.0 in /opt/conda/envs/
Python-3.9/lib/python3.9/site-packages (from boto3->watson_machine_learning
_client) (0.5.0)
Requirement already satisfied: botocore<1.22.0,>=1.21.21 in /opt/conda/envs
/Python-3.9/lib/python3.9/site-packages (from boto3->watson_machine_learnin
g_client) (1.21.41)
Requirement already satisfied: jmespath<1.0.0,>=0.7.1 in /opt/conda/envs/Py
thon-3.9/lib/python3.9/site-packages (from boto3->watson_machine_learning_c
lient) (0.10.0)
Requirement already satisfied: python-dateutil<3.0.0,>=2.1 in /opt/conda/en
vs/Python-3.9/lib/python3.9/site-packages (from botocore<1.22.0,>=1.21.21->
boto3->watson_machine_learning_client) (2.8.2)
Requirement already satisfied: six>=1.5 in /opt/conda/envs/Python-3.9/lib/p
ython3.9/site-packages (from python-dateutil<3.0.0,>=2.1->botocore<1.22.0,>
=1.21.21->boto3->watson_machine_learning_client) (1.15.0)
Requirement already satisfied: ibm-cos-sdk-core==2.11.0 in /opt/conda/envs/
Python-3.9/lib/python3.9/site-packages (from ibm-cos-sdk->watson_machine_le
arning_client) (2.11.0)
Requirement already satisfied: ibm-cos-sdk-s3transfer==2.11.0 in /opt/conda
/envs/Python-3.9/lib/python3.9/site-packages (from ibm-cos-sdk->watson_mach
ine_learning_client) (2.11.0)
Requirement already satisfied: charset-normalizer~=2.0.0 in /opt/conda/envs
/Python-3.9/lib/python3.9/site-packages (from requests->watson_machine_lear
ning_client) (2.0.4)
Requirement already satisfied: idna<4,>=2.5 in /opt/conda/envs/Python-3.9/l
ib/python3.9/site-packages (from requests->watson_machine_learning_client)
(3.3)
Requirement already satisfied: pytz>=2017.3 in /opt/conda/envs/Python-3.9/l
ib/python3.9/site-packages (from pandas->watson_machine_learning_client) (2
021.3)
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) (
1.20.3)
Installing collected packages: watson-machine-learning-client
Successfully installed watson-machine-learning-client-1.0.391
```

15.]IMPORTING APICLIENT FOR DEPLOYING.

In [138]:
```
from ibm_watson_machine_learning import APIClient
url_credentials = {
    "url": "https://us-south.ml.cloud.ibm.com",
    "apikey": "sqLVTXSP3nnAKfzJ1rKRKCpNzS_XZ8_HXa9FRwV7BvOP"
}
client = APIClient(url_credentials)
```

In [139]:
```
client = APIClient(url_credentials)
```

16.]CREATING API_CLIENT SPACE ID.

In [140]:
```
def guid_from_space_name(client, space_name):
    space = client.spaces.get_details()
    return(next(item for item in space['resources'] if
item['entity']['name'] == space_name)['metadata']['id'])
```

In [143]:
```
space_uid = guid_from_space_name(client, 'Image Classification')
print("space UID = " + space_uid)
```

```
space UID = d90f421e-9169-47e7-a58c-0e7bb0e65685
```

In [145]:
```
client.set.default_space(space_uid)
```

Out[145]:
```
'SUCCESS'
```

In [147]:
```
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
autoai-kb_rt22.2-py3.10         125b6d9a-5b1f-5e8d-972a-b251688ccf40  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
pytorch_1.2-py3.6               2c8ef57d-2687-4b7d-acce-01f94976dac1  base
spark-mllib_2.3                 2e51f700-bca0-4b0d-88dc-5c6791338875  base
pytorch-onnx_1.1-py3.6-edt      32983cea-3f32-4400-8965-dde874a8d67e  base
spark-mllib_3.0-py37            36507ebe-8770-55ba-ab2a-eafe787600e9  base
spark-mllib_2.4                 390d21f8-e58b-4fac-9c55-d7ceda621326  base
autoai-ts_rt22.2-py3.10         396b2e83-0953-5b86-9a55-7ce1628a406f  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
pytorch-onnx_rt22.2-py3.10      40e73f55-783a-5535-b3fa-0c8b94291431  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
------------------------------  ------------------------------------  ----
Note: Only first 50 records were displayed. To display more use 'limit' par
ameter.
```

In [ ]:

```
software_spec_uid =
client.software_specifications.get_uid_by_name("tensorflow")
software_spec_uid
```

17.]STORING THE MODEL_ID FOR DATASET.H5

In [ ]:

```
#store the model
model_details = client.repository.store_model(model='Image-classification-
model_new.tgz',meta_props={
    client.repository.ModelMetaNames.NAME:'CNN',
    client.repository.ModelMetaNames.TYPE:"keras_2.2.4",
    client.repository.ModelMetaNames.SOFTWARE_SPEC_UID:software_spec_uid}
                                          )
model_id = client.repository.get_model_uid(model_details)
```

In [ ]:

```
model_id
```

In [171]:

```
model.save('Dataset.h5')
```

18.]DOWNLOADING THE TAR FILE ON CLIENT REPOSITORY

In [ ]:
```
client.repository.download(model_id, 'my_model.tar.gz')
```

19.]TEST THE MODEL

In [186]:
```
import numpy as np
from tensorflow.keras.models import load_model
from keras.preprocessing import image
```

20.]LOADING THE DATASET

In [187]:
```
#Load the model
model=load_model('Dataset.h5')
```

21.]ADDING STREAMING_BODY FOR TEST IMAGE.

In [188]:
```
import os, types
import pandas as pd
from botocore.client import Config
import ibm_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.
cos_client = ibm_boto3.client(service_name='s3',
    ibm_api_key_id='aqprHZFuH38ECUn869hHk4qyvS_iKJfrZAWUJJQ-mQKx',
    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')

bucket = 'realtimecommunicationforspecially-donotdelete-pr-rfqndcvwgch6fu'
object_key = '1.png'

streaming_body_5 = cos_client.get_object(Bucket=bucket,
Key=object_key)['Body']
```

```
# Your 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/
```

22.]TESTING ON SEVERAL TESTING IMAGES

In [189]:

```
img = image.load_img(streaming_body_5,target_size=(64, 64))
#img=image.load_img("/home/wsuser/work/1",target_size=(64,64))
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
/tmp/wsuser/ipykernel_164/365554034.py in
----> 1 img = image.load_img(streaming_body_5,target_size=(64, 64))
      2 #img=image.load_img("/home/wsuser/work/1",target_size=(64,64))

/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/keras/preprocessing/
image.py in load_img(path, grayscale, color_mode, target_size, interpolatio
n)
    311     ```python
    312     (x_train, y_train), (x_test, y_test) = cifar10.load_data()
--> 313     y_train = np_utils.to_categorical(y_train, num_classes)
    314     y_test = np_utils.to_categorical(y_test, num_classes)
    315

/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/keras_preprocessing/
image/utils.py in load_img(path, grayscale, color_mode, target_size, interp
olation)
    111         raise ImportError('Could not import PIL.Image. '
    112                           'The use of `load_img` requires PIL.')
--> 113     with open(path, 'rb') as f:
    114         img = pil_image.open(io.BytesIO(f.read()))
    115         if color_mode == 'grayscale':

TypeError: expected str, bytes or os.PathLike object, not StreamingBody
```

In [ ]:

```
ls
```

In [181]:

```
img=image.load_img(r"/home/wsuser/work/Dataset/test_set/A/1.png")
```

```
---------------------------------------------------------------------------
FileNotFoundError                         Traceback (most recent call last)
/tmp/wsuser/ipykernel_164/1035932264.py in
----> 1 img=image.load_img(r"/content/drive/MyDrive/IBM_PROJECT/Dataset/tra
ining_set/A/1.png")

/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/tensorflow/python/ke
ras/preprocessing/image.py in load_img(path, grayscale, color_mode, target_
size, interpolation)
    311       ValueError: if interpolation method is not supported.
    312   """
```

```
--> 313   return image.load_img(path, grayscale=grayscale, color_mode=color
_mode,
    314                           target_size=target_size, interpolation=inte
rpolation)
    315

/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/keras_preprocessing/
image/utils.py in load_img(path, grayscale, color_mode, target_size, interp
olation)
    111         raise ImportError('Could not import PIL.Image. '
    112                           'The use of `load_img` requires PIL.')
--> 113     with open(path, 'rb') as f:
    114         img = pil_image.open(io.BytesIO(f.read()))
    115         if color_mode == 'grayscale':

FileNotFoundError: [Errno 2] No such file or directory: '/content/drive/MyD
rive/IBM_PROJECT/Dataset/training_set/A/1.png'
```

In [ ]:
```
img
```

In [ ]:
```
img1=image.load_ing(r"/home/wsuser/work/Dataset/test_set/C/1.png")
```

In [ ]:
```
img1
```

In [ ]:
```
x=image.img_to_array(img)
```

In [ ]:
```
x
```

In [ ]:
```
x1=np.expand_dims(x,axis=1)
```

In [ ]:
```
x1
```

In [ ]:
```
y=np.argmax(model.predoct(x),axis=1)
```

In [ ]:
```
y
```

In [ ]:
```
x_train.class_indices
```

In [ ]:
```
index=['A','B','C','D','E','F','G','H','I']
```

In [ ]:
```
index[y[0]]
```

In [ ]:
```
img=image.load_img(r"/home/wsuser/work/Dataset/test_set/A/90.png",target_si
ze=(64,64))
x=image.ing_to_array(img)
x=np.expand_dims(x,axis=0)
y=fnp.argmax(model.predict(x),axis=1)
index=['A','B','C','D','E','F','G','H','I']
index[y[0]]]
```

In [ ]:

```
img=image.load_img(
"/home/wsuser/work/Dataset/test_set/D/1.png",target_size=(64,64))
x=image.ing_to_array(img)
x=np.expand_dims(x,axis=0)
y=np.argmax(model.predict(x)
index=['A','B','C','D','E','F','G','H','I']
index[y[0]]
```

In [ ]:

```
img=image.load_img(r"/content/drive/MyDrive/IBM_PROJECT/Dataset/test_set/G/
1.png",target_size=(64,64))
x=image.ing_to_array(img)
x=np.expand_dims(x,axisme)
y=np.argmax(model.predict(x), axis=1)
index=['A','B','C','D','E','F','G','H','I']
index[y[0]]
```

In[]:

```
img=image.load_img(r"/content/drive/MyDrive/IBM_PROJECT/Dataset/test_set/D/
1.png",target_size=(64,64))
x-image.ing_to_array(img)
x=np.expand_dims(x,axisme)
y=np.argmax(model.predict(x), axis=1)
index=['A','B','C','D','E','F','G','H','I']
index[y[0]]
```

In [ ]:

```
!tar -zcvf Dataset-classification-model.tgz specially.h5
```

In [ ]:

```
import tensorflow as tf
tf .__ _version_
```

In [ ]:

```
!pip install keras == 2.2.4
```

23.]IBM DEPLOYMENT

In [ ]:

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

In [ ]:

```
from ibm_watson_machine learning import APIClient
wml_credentials={
"url":"https://us-south.ml.cloud.ibm.com",
"apikey":"x91CJTUTrrIfLvrXsKf8yLyI1KHb3JV0Y7Qrwy1zilb2"
}
client=APIClient(wml_credentials)
```

CLIENT

In [ ]:

```
def guid_space_name(client,animal_deploy):
space-client.spaces.get_details()
return(next(item for item in space[' resources'] if iten['entity']['name']=
animal_deploy)["metadata']['id'])
```

In [ ]:
```
space_uid-guid_space_name(client,'animal_deploy")
print("Space UID "+space_uid)
```

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

In [ ]:
```
client,software specifications.list(200)
```

In [ ]:
```
software_space_uid=client.software_specifications.get_uid_by_name('tensorfl
ow_rt22.1-py3.9¹)
```

In [ ]:
```
software_space_uid
```

In [ ]:
```
model_details=client.repository.store_model(model='Dataset.tgz',meta_props=
{
client.repository.ModelMetaNames.NAME: "CNN Model Building",
client.repository.ModelMetaNames.TYPE: 'tensorflow_2.7',
client.repository.ModelMetaNames.SOFTWARE_SPEC_UID: software_space_uid
})
```

In [ ]:
```
model_id=client.repository.get_model_id(model_details)
```

In [ ]:
```
model_id
```