In [ ]:

```
import numpy as np
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras import layers
import h5py
import warnings
warnings.filterwarnings('ignore')
```

In [ ]:

```
file_electron = "SingleElectronPt50_IMGCROPS_n249k_RHv1.hdf5"
file_photon = "SinglePhotonPt50_IMGCROPS_n249k_RHv1.hdf5"

with h5py.File(file_electron, "r") as f1:
    X_elec = np.array(f1['X'][:])
    y_elec = np.array(f1['y'][:])
with h5py.File(file_photon, "r") as f2:
    X_phot = np.array(f2['X'][:])
    y_phot = np.array(f2['y'][:])
print(X_elec.shape)
print(X_phot.shape)
```

```
(249000, 32, 32, 2)
(249000, 32, 32, 2)
```

In [ ]:

```
num_classes = 2
input_shape = (32, 32, 2)
X = np.append(X_elec, X_phot, axis=0)
y = np.append(y_elec, y_phot)
X.shape
```

Out[ ]:

```
(498000, 32, 32, 2)
```

In [ ]:

```
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(X, y, test_size=.2, random_state=4, stratify=y)
```

In [ ]:

```
y_train = keras.utils.to_categorical(y_train, num_classes=2)
y_test = keras.utils.to_categorical(y_test, num_classes=2)
```

In [ ]:

```
learning_rate = 0.0001
batch_size = 256
num_epochs = 50
patch_size = 2
num_patches = (input_shape[0]//patch_size)**2
projection_dim = 64
num_heads = 2
transformer_units = [
    projection_dim * 2,
    projection_dim,
]
transformer_layers = 2
mlp_head_units = [512, 256]
```

In [ ]:

```
def mlp(x, hidden_units):
    for units in hidden_units:
        x = layers.Dense(units, activation=tf.nn.gelu)(x)
    return x
```

In [ ]:

```
class Patches(layers.Layer):
    def __init__(self, patch_size):
        super(Patches, self).__init__()
        self.patch_size = patch_size

    def call(self, images):
        batch_size = tf.shape(images)[0]
        patches = tf.image.extract_patches(
            images=images,
            sizes=[1, self.patch_size, self.patch_size, 1],
            strides=[1, self.patch_size, self.patch_size, 1],
            rates=[1, 1, 1, 1],
            padding="VALID",
        )
        patch_dims = patches.shape[-1]
        patches = tf.reshape(patches, [batch_size, -1, patch_dims])
        return patches
```

In [ ]:

```
class PatchEncoder(layers.Layer):
    def __init__(self, num_patches, projection_dim):
        super(PatchEncoder, self).__init__()
        self.num_patches = num_patches
        self.projection = layers.Dense(units=projection_dim)
        self.position_embedding = layers.Embedding(
            input_dim=num_patches, output_dim=projection_dim
        )

    def call(self, patch):
        positions = tf.range(start=0, limit=self.num_patches, delta=1)
        encoded = self.projection(patch) + self.position_embedding(positions)
        return encoded
```

In [ ]:

```
def create_vit_classifier():
    inputs = layers.Input(shape=input_shape)
    patches = Patches(patch_size)(inputs)
    encoded_patches = PatchEncoder(num_patches, projection_dim)(patches)

    for _ in range(transformer_layers):
        x1 = layers.LayerNormalization(epsilon=1e-6)(encoded_patches)
        attention_output = layers.MultiHeadAttention(
            num_heads=num_heads, key_dim=projection_dim, dropout=0
        )(x1, x1)
        x2 = layers.Add()([attention_output, encoded_patches])
        x3 = layers.LayerNormalization(epsilon=1e-6)(x2)
        x3 = mlp(x3, hidden_units=transformer_units)
        encoded_patches = layers.Add()([x3, x2])
    representation = layers.LayerNormalization(epsilon=1e-6)(encoded_patches)
    representation = layers.Flatten()(representation)
    features = mlp(representation, hidden_units=mlp_head_units)
    outputs = layers.Dense(num_classes, activation='softmax')(features)
    model = keras.Model(inputs=inputs, outputs=outputs)
    return model
```

In [ ]:

```
def run_experiment(model):
    model.compile(optimizer=tf.optimizers.Adam(learning_rate=learning_rate), loss='categorical_crossentropy', metrics=[tf.keras.me

    reduce_lr = tf.keras.callbacks.ReduceLROnPlateau(monitor='val_auc', factor=0.2,
                              patience=10, min_lr=1e-10    , verbose=1)
    history = model.fit(
        x=x_train,
        y=y_train,
        batch_size=batch_size,
        epochs=num_epochs,
        validation_split=0.1,
        callbacks=[reduce_lr]
    )
    return model, history
```

In [ ]:

```
vit_classifier = create_vit_classifier()
with tf.device('/gpu:0'):
    model, history = run_experiment(vit_classifier)
```

```
2022-03-23 01:37:27.243773: I tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:936] successful NUMA node read from SysFS had n
egative value (-1), but there must be at least one NUMA node, so returning NUMA node zero
2022-03-23 01:37:27.261132: I tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:936] successful NUMA node read from SysFS had n
egative value (-1), but there must be at least one NUMA node, so returning NUMA node zero
2022-03-23 01:37:27.263336: I tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:936] successful NUMA node read from SysFS had n
egative value (-1), but there must be at least one NUMA node, so returning NUMA node zero
2022-03-23 01:37:27.266516: I tensorflow/core/platform/cpu_feature_guard.cc:151] This TensorFlow binary is optimized with oneAPI D
eep Neural Network Library (oneDNN) to use the following CPU instructions in performance-critical operations:  AVX2 FMA
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
2022-03-23 01:37:27.268711: I tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:936] successful NUMA node read from SysFS had n
egative value (-1), but there must be at least one NUMA node, so returning NUMA node zero
2022-03-23 01:37:27.270947: I tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:936] successful NUMA node read from SysFS had n
egative value (-1), but there must be at least one NUMA node, so returning NUMA node zero
2022-03-23 01:37:27.273037: I tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:936] successful NUMA node read from SysFS had n
egative value (-1), but there must be at least one NUMA node, so returning NUMA node zero
2022-03-23 01:37:28.160767: I tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:936] successful NUMA node read from SysFS had n
egative value (-1), but there must be at least one NUMA node, so returning NUMA node zero
2022-03-23 01:37:28.163021: I tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:936] successful NUMA node read from SysFS had n
egative value (-1), but there must be at least one NUMA node, so returning NUMA node zero
2022-03-23 01:37:28.165119: I tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:936] successful NUMA node read from SysFS had n
egative value (-1), but there must be at least one NUMA node, so returning NUMA node zero
2022-03-23 01:37:28.167292: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1525] Created device /job:localhost/replica:0/task:
0/device:GPU:0 with 47216 MB memory:  -> device: 0, name: Quadro RTX 8000, pci bus id: 0000:04:00.0, compute capability: 7.5
Epoch 1/50
1401/1401 [==============================] - 91s 62ms/step - loss: 0.6781 - auc: 0.6304 - val_loss: 0.6397 - val_auc: 0.6838 - lr:
1.0000e-04
Epoch 2/50
1401/1401 [==============================] - 87s 62ms/step - loss: 0.6399 - auc: 0.6851 - val_loss: 0.6214 - val_auc: 0.7124 - lr:
1.0000e-04
Epoch 3/50
1401/1401 [==============================] - 86s 62ms/step - loss: 0.6218 - auc: 0.7124 - val_loss: 0.6009 - val_auc: 0.7418 - lr:
1.0000e-04
Epoch 4/50
1401/1401 [==============================] - 87s 62ms/step - loss: 0.6093 - auc: 0.7294 - val_loss: 0.6072 - val_auc: 0.7382 - lr:
1.0000e-04
Epoch 5/50
1401/1401 [==============================] - 87s 62ms/step - loss: 0.6012 - auc: 0.7394 - val_loss: 0.5904 - val_auc: 0.7528 - lr:
1.0000e-04
Epoch 6/50
1401/1401 [==============================] - 87s 62ms/step - loss: 0.5950 - auc: 0.7467 - val_loss: 0.5902 - val_auc: 0.7538 - lr:
1.0000e-04
```

```
Epoch 7/50
1401/1401 [==============================] - 87s 62ms/step - loss: 0.5884 - auc: 0.7541 - val_loss: 0.5798 - val_auc: 0.7634 - lr:
1.0000e-04
Epoch 8/50
1401/1401 [==============================] - 87s 62ms/step - loss: 0.5868 - auc: 0.7558 - val_loss: 0.5848 - val_auc: 0.7603 - lr:
1.0000e-04
Epoch 9/50
1401/1401 [==============================] - 87s 62ms/step - loss: 0.5817 - auc: 0.7616 - val_loss: 0.5818 - val_auc: 0.7627 - lr:
1.0000e-04
Epoch 10/50
1401/1401 [==============================] - 87s 62ms/step - loss: 0.5776 - auc: 0.7660 - val_loss: 0.5830 - val_auc: 0.7610 - lr:
1.0000e-04
Epoch 11/50
1400/1401 [============================>.] - ETA: 0s - loss: 0.5741 - auc: 0.7698
Epoch 11: ReduceLROnPlateau reducing learning rate to 1.9999999494757503e-05.
1401/1401 [==============================] - 87s 62ms/step - loss: 0.5741 - auc: 0.7698 - val_loss: 0.5729 - val_auc: 0.7709 - lr:
1.0000e-04
Epoch 12/50
1401/1401 [==============================] - 87s 62ms/step - loss: 0.5610 - auc: 0.7831 - val_loss: 0.5704 - val_auc: 0.7742 - lr:
2.0000e-05
Epoch 13/50
1401/1401 [==============================] - 86s 62ms/step - loss: 0.5590 - auc: 0.7850 - val_loss: 0.5650 - val_auc: 0.7792 - lr:
2.0000e-05
Epoch 14/50
1401/1401 [==============================] - 86s 62ms/step - loss: 0.5575 - auc: 0.7865 - val_loss: 0.5682 - val_auc: 0.7761 - lr:
2.0000e-05
Epoch 15/50
1401/1401 [==============================] - 87s 62ms/step - loss: 0.5561 - auc: 0.7879 - val_loss: 0.5643 - val_auc: 0.7799 - lr:
2.0000e-05
Epoch 16/50
1401/1401 [==============================] - 86s 62ms/step - loss: 0.5548 - auc: 0.7891 - val_loss: 0.5666 - val_auc: 0.7777 - lr:
2.0000e-05
Epoch 17/50
1401/1401 [==============================] - 87s 62ms/step - loss: 0.5539 - auc: 0.7900 - val_loss: 0.5686 - val_auc: 0.7763 - lr:
2.0000e-05
Epoch 18/50
1401/1401 [==============================] - 87s 62ms/step - loss: 0.5525 - auc: 0.7913 - val_loss: 0.5664 - val_auc: 0.7776 - lr:
2.0000e-05
Epoch 19/50
1401/1401 [==============================] - 87s 62ms/step - loss: 0.5514 - auc: 0.7924 - val_loss: 0.5647 - val_auc: 0.7795 - lr:
2.0000e-05
Epoch 20/50
1401/1401 [==============================] - 87s 62ms/step - loss: 0.5500 - auc: 0.7937 - val_loss: 0.5664 - val_auc: 0.7795 - lr:
2.0000e-05
Epoch 21/50
1400/1401 [============================>.] - ETA: 0s - loss: 0.5486 - auc: 0.7950
Epoch 21: ReduceLROnPlateau reducing learning rate to 3.999999898951501e-06.
1401/1401 [==============================] - 87s 62ms/step - loss: 0.5486 - auc: 0.7950 - val_loss: 0.5658 - val_auc: 0.7795 - lr:
2.0000e-05
Epoch 22/50
1401/1401 [==============================] - 86s 62ms/step - loss: 0.5435 - auc: 0.7999 - val_loss: 0.5634 - val_auc: 0.7808 - lr:
4.0000e-06
Epoch 23/50
1401/1401 [==============================] - 86s 62ms/step - loss: 0.5428 - auc: 0.8006 - val_loss: 0.5632 - val_auc: 0.7818 - lr:
4.0000e-06
Epoch 24/50
1401/1401 [==============================] - 87s 62ms/step - loss: 0.5422 - auc: 0.8011 - val_loss: 0.5638 - val_auc: 0.7810 - lr:
4.0000e-06
Epoch 25/50
1401/1401 [==============================] - 87s 62ms/step - loss: 0.5418 - auc: 0.8014 - val_loss: 0.5624 - val_auc: 0.7820 - lr:
4.0000e-06
Epoch 26/50
1401/1401 [==============================] - 87s 62ms/step - loss: 0.5414 - auc: 0.8018 - val_loss: 0.5633 - val_auc: 0.7813 - lr:
4.0000e-06
Epoch 27/50
1401/1401 [==============================] - 87s 62ms/step - loss: 0.5409 - auc: 0.8023 - val_loss: 0.5632 - val_auc: 0.7823 - lr:
4.0000e-06
Epoch 28/50
1401/1401 [==============================] - 87s 62ms/step - loss: 0.5404 - auc: 0.8027 - val_loss: 0.5630 - val_auc: 0.7815 - lr:
4.0000e-06
Epoch 29/50
1401/1401 [==============================] - 86s 62ms/step - loss: 0.5399 - auc: 0.8031 - val_loss: 0.5627 - val_auc: 0.7817 - lr:
4.0000e-06
Epoch 30/50
1401/1401 [==============================] - 86s 62ms/step - loss: 0.5394 - auc: 0.8037 - val_loss: 0.5635 - val_auc: 0.7813 - lr:
4.0000e-06
Epoch 31/50
1400/1401 [============================>.] - ETA: 0s - loss: 0.5389 - auc: 0.8041
Epoch 31: ReduceLROnPlateau reducing learning rate to 7.999999979801942e-07.
1401/1401 [==============================] - 87s 62ms/step - loss: 0.5389 - auc: 0.8041 - val_loss: 0.5635 - val_auc: 0.7809 - lr:
4.0000e-06
Epoch 32/50
1401/1401 [==============================] - 86s 62ms/step - loss: 0.5375 - auc: 0.8054 - val_loss: 0.5631 - val_auc: 0.7816 - lr:
8.0000e-07
Epoch 33/50
1401/1401 [==============================] - 87s 62ms/step - loss: 0.5374 - auc: 0.8055 - val_loss: 0.5630 - val_auc: 0.7818 - lr:
8.0000e-07
Epoch 34/50
1401/1401 [==============================] - 87s 62ms/step - loss: 0.5372 - auc: 0.8056 - val_loss: 0.5630 - val_auc: 0.7818 - lr:
8.0000e-07
Epoch 35/50
1401/1401 [==============================] - 86s 62ms/step - loss: 0.5371 - auc: 0.8057 - val_loss: 0.5630 - val_auc: 0.7818 - lr:
```

```
8.0000e-07
Epoch 36/50
1401/1401 [==============================] - 87s 62ms/step - loss: 0.5370 - auc: 0.8058 - val_loss: 0.5629 - val_auc: 0.7818 - lr:
8.0000e-07
Epoch 37/50
1401/1401 [==============================] - 87s 62ms/step - loss: 0.5369 - auc: 0.8059 - val_loss: 0.5629 - val_auc: 0.7818 - lr:
8.0000e-07
Epoch 38/50
1401/1401 [==============================] - 87s 62ms/step - loss: 0.5368 - auc: 0.8060 - val_loss: 0.5630 - val_auc: 0.7819 - lr:
8.0000e-07
Epoch 39/50
1401/1401 [==============================] - 86s 62ms/step - loss: 0.5367 - auc: 0.8061 - val_loss: 0.5629 - val_auc: 0.7822 - lr:
8.0000e-07
Epoch 40/50
1401/1401 [==============================] - 87s 62ms/step - loss: 0.5366 - auc: 0.8062 - val_loss: 0.5634 - val_auc: 0.7817 - lr:
8.0000e-07
Epoch 41/50
1400/1401 [============================>.] - ETA: 0s - loss: 0.5365 - auc: 0.8063
Epoch 41: ReduceLROnPlateau reducing learning rate to 1.600000018697756e-07.
1401/1401 [==============================] - 86s 62ms/step - loss: 0.5365 - auc: 0.8062 - val_loss: 0.5633 - val_auc: 0.7816 - lr:
8.0000e-07
Epoch 42/50
1401/1401 [==============================] - 86s 62ms/step - loss: 0.5362 - auc: 0.8066 - val_loss: 0.5629 - val_auc: 0.7821 - lr:
1.6000e-07
Epoch 43/50
1401/1401 [==============================] - 86s 62ms/step - loss: 0.5361 - auc: 0.8066 - val_loss: 0.5629 - val_auc: 0.7819 - lr:
1.6000e-07
Epoch 44/50
1401/1401 [==============================] - 86s 62ms/step - loss: 0.5361 - auc: 0.8067 - val_loss: 0.5629 - val_auc: 0.7820 - lr:
1.6000e-07
Epoch 45/50
1401/1401 [==============================] - 86s 62ms/step - loss: 0.5361 - auc: 0.8067 - val_loss: 0.5629 - val_auc: 0.7820 - lr:
1.6000e-07
Epoch 46/50
1401/1401 [==============================] - 87s 62ms/step - loss: 0.5360 - auc: 0.8067 - val_loss: 0.5629 - val_auc: 0.7821 - lr:
1.6000e-07
Epoch 47/50
1401/1401 [==============================] - 86s 62ms/step - loss: 0.5360 - auc: 0.8067 - val_loss: 0.5628 - val_auc: 0.7821 - lr:
1.6000e-07
Epoch 48/50
1401/1401 [==============================] - 86s 62ms/step - loss: 0.5360 - auc: 0.8067 - val_loss: 0.5629 - val_auc: 0.7819 - lr:
1.6000e-07
Epoch 49/50
1401/1401 [==============================] - 86s 62ms/step - loss: 0.5360 - auc: 0.8068 - val_loss: 0.5629 - val_auc: 0.7820 - lr:
1.6000e-07
Epoch 50/50
1401/1401 [==============================] - 86s 62ms/step - loss: 0.5359 - auc: 0.8068 - val_loss: 0.5629 - val_auc: 0.7819 - lr:
1.6000e-07
```

In [ ]:

```
model.evaluate(x_test,y_test)
```

```
3113/3113 [==============================] - 42s 14ms/step - loss: 0.5674 - auc: 0.7775
```

Out[ ]:

```
[0.5674085021018982, 0.7774555683135986]
```

Results

Train AUC Score: 0.806

Validation AUC Score: 0.782

Test AUC Score: 0.777