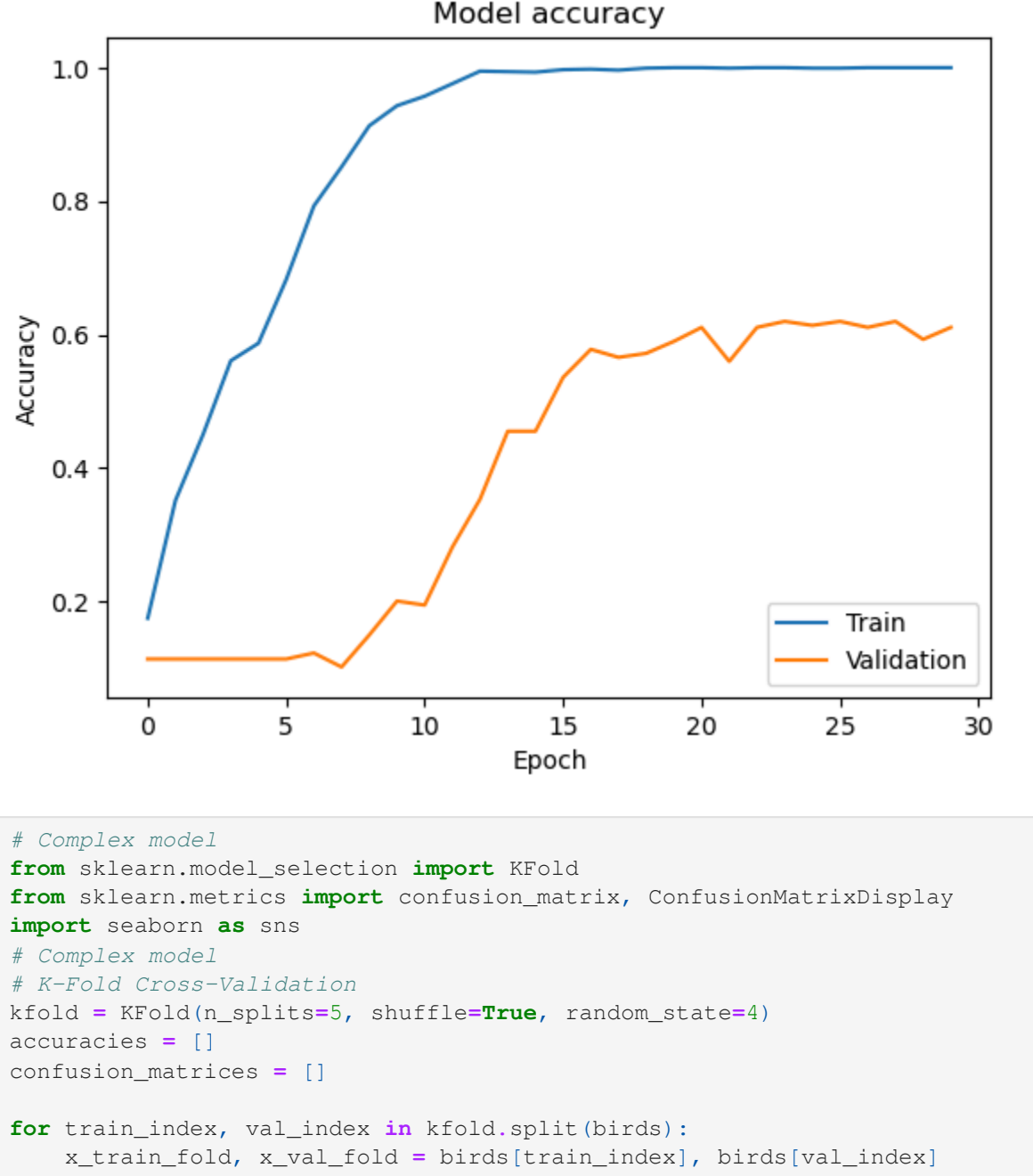
In [1]:
```python
import librosa
import librosa.display
import numpy as np
import scipy.signal
import matplotlib.pyplot as plt
%matplotlib inline
from IPython.display import Audio, IFrame, display
import glob
from scipy.ndimage import zoom
desired_shape = (128, 128)

def resize_spectrogram(spectrogram, desired_shape):
    zoom_factors = [desired_shape[0] / spectrogram.shape[0], desired_shape[1] / spectrogram.shape[1]]
    return zoom(spectrogram, zoom_factors)

birds=[]
labels=[]
sr = 16000
file_path=glob.glob('./birdSounds/*')
#Creating spectograms of bird sounds and append them to bird list and their labels to label list.
number=0
for file in file_path:
    a, sr = librosa.load(file, mono=True, sr=sr, offset=0)
    S = librosa.feature.melspectrogram(y=a, sr=sr, n_mels=128)
    S_dB = librosa.power_to_db(S, ref=np.max)
    S_dB = resize_spectrogram(S_dB, desired_shape)
    birds.append(S_dB)
    if 'Bent-Beak-Riffraff' in file:
        labels.append('Bent Beak Riffraff')
    elif 'Blue-collared-Zipper' in file:
        labels.append('Blue collared Zipper')
    elif 'Bombadil' in file:
        labels.append('Bombadil')
    elif 'Broad-winged-Jojo' in file:
        labels.append('Broad winged Jojo')
    elif 'Canadian-Cootamum' in file:
        labels.append('Canadian Cootamum')
    elif "Carries-Champagne-Pipit" in file:
        labels.append('Carries Champagne Pipit')
    elif 'Darkwing-Sparrow'in file:
        labels.append('Darkwing Sparrow')
    elif 'Eastern-Corn-Skeet' in file:
        labels.append('Eastern Corn Skeet')
    elif 'Green-Tipped-Scarlet-Pipit' in file:
        labels.append('Green tipped Scarlet Pipit')
    elif 'Lesser-Birchbeere' in file:
        labels.append('Lesser Birchbeere')
    elif 'Orange-Pine-Plover' in file:
        labels.append('Orange Pine Plover')
    elif 'Ordinary-Snape' in file:
        labels.append('Ordinary Snape')
    elif 'Pinkfinch' in file:
        labels.append('Pinkfinch')
    elif 'Purple-Tooting-Tout' in file:
        labels.append('Purple Tooting Tout')
    elif 'Qax' in file:
        labels.append('Qax')
    elif 'Queenscoat' in file:
        labels.append('Queenscoat')
    elif 'Rose-Crested-Blue-Pipit' in file:
        labels.append('Rose crested Blue Pipit')
    elif 'Scrawny-Jay' in file:
        labels.append('Scrawny Jay')
    elif 'Vermillion-Trillian'in file:
        labels.append('Vermillion Trillian')
    else:
        labels.append('Unknow')
    number+=1
```

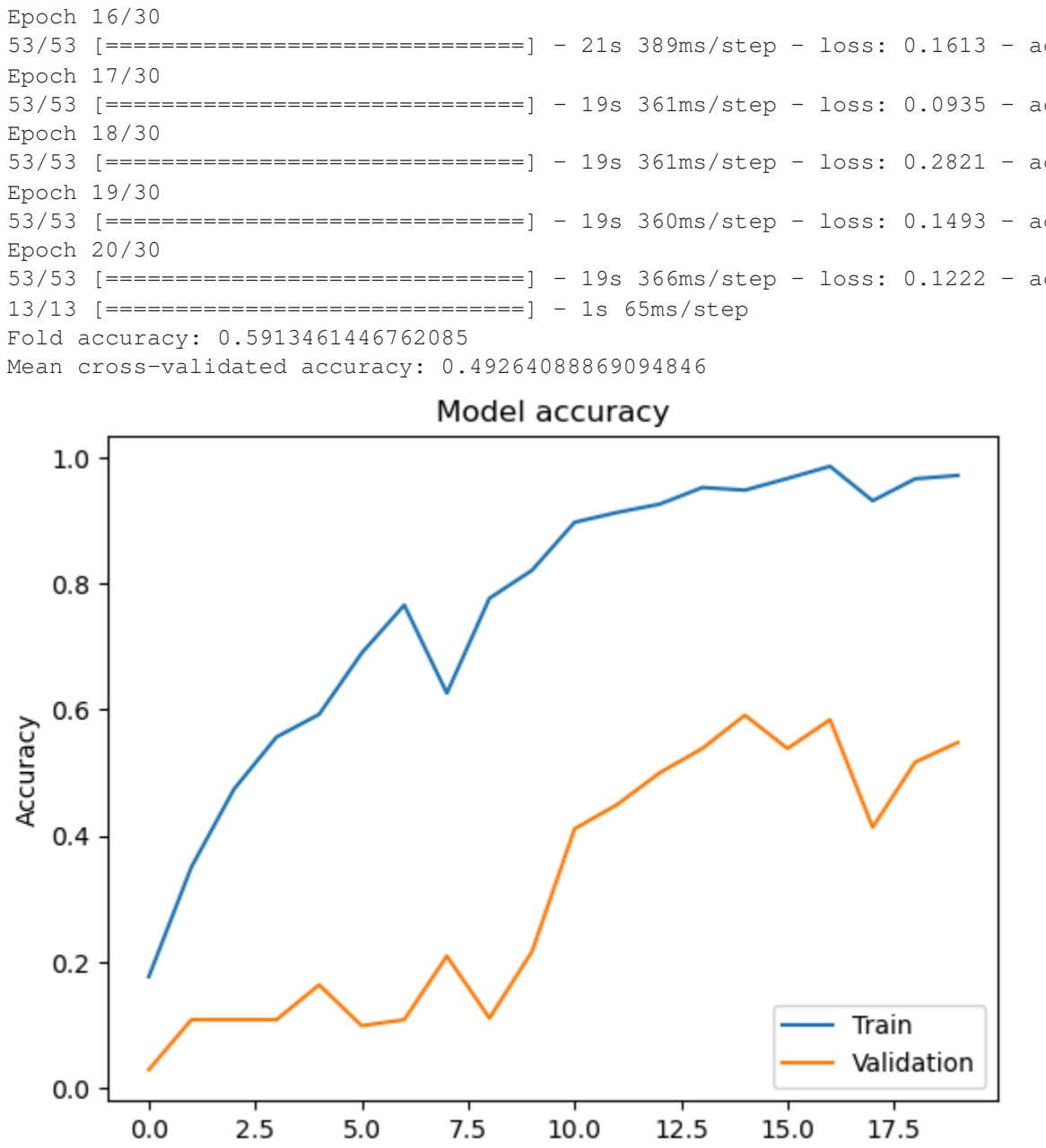
In [2]:
```python
from tensorflow.keras.callbacks import EarlyStopping
from tensorflow import keras
from tensorflow.keras import layers
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
early_stopping = EarlyStopping(monitor='val_loss', patience=5, restore_best_weights=True)
# Converting lists to Numpy array
birds=np.array(birds)
labels=np.array(labels)
birds=birds[..., np.newaxis]
birds = np.squeeze(birds)

num_classes = 20
input_shape = (128, 128,1)
# Simple model
labelencoder = LabelEncoder()
labels = labelencoder.fit_transform(labels)
x_train, x_test, y_train, y_test = train_test_split(birds,labels, test_size=0.2, random_state=4)
x_train = x_train.astype("float32") / 255
x_test = x_test.astype("float32") / 255

y_train = keras.utils.to_categorical(y_train, num_classes)
model = keras.Sequential([
    keras.Input(shape=input_shape),
    layers.Conv2D(32, kernel_size=(3, 3), activation="relu"),
    layers.BatchNormalization(),
    layers.MaxPooling2D(pool_size=(2, 2)),
    layers.Conv2D(64, kernel_size=(3, 3), activation="relu"),
    layers.BatchNormalization(),
    layers.MaxPooling2D(pool_size=(2, 2)),
    layers.Conv2D(128, kernel_size=(3, 3), activation="relu"),
    layers.BatchNormalization(),
    layers.MaxPooling2D(pool_size=(2, 2)),
    layers.Flatten(),
    layers.Dense(128, activation="relu"),
    layers.BatchNormalization(),
    layers.Dropout(0.5),
    layers.Dense(num_classes, activation="softmax"),
    ]
    )

model.summary()
batch_size = 32
epochs = 30
model.compile(loss="categorical_crossentropy", optimizer="adam", metrics=["accuracy"])
history = model.fit(x_train, y_train, batch_size=batch_size, epochs=epochs, validation_split=0.2,callbacks=[early_stopping])
print(history)
plt.plot(history.history['accuracy'])
plt.plot(history.history['val_accuracy'])
plt.title('Model accuracy')
plt.ylabel('Accuracy')
plt.xlabel('Epoch')
plt.legend(['Train', 'Validation'], loc='lower right')
plt.show()
```

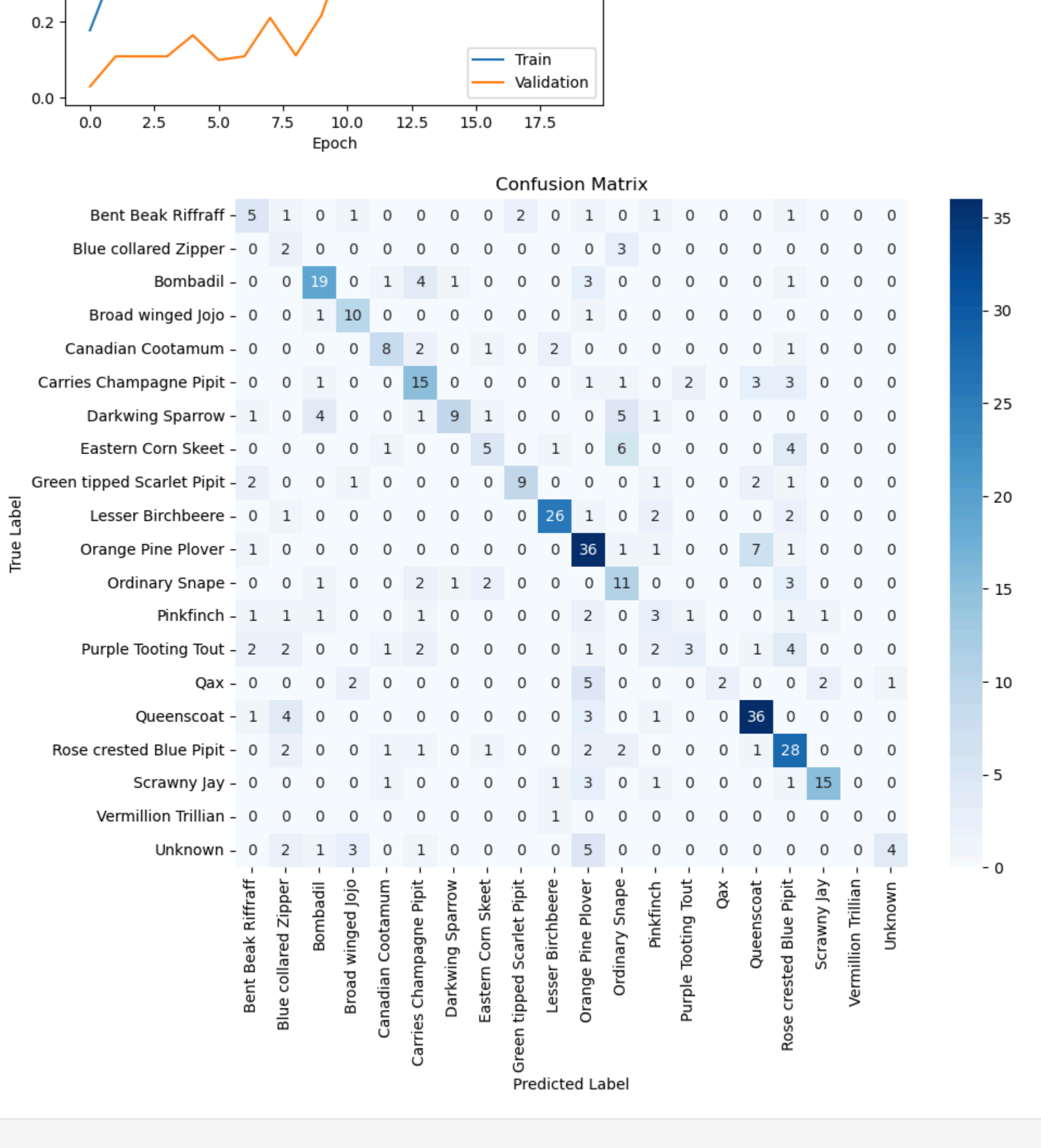
```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 126, 126, 32)      320

 batch_normalization (Batch  (None, 126, 126, 32)      128
 Normalization)

 max_pooling2d (MaxPooling2  (None, 63, 63, 32)        0
 D)

 conv2d_1 (Conv2D)           (None, 61, 61, 64)        18496

 batch_normalization_1 (Bat  (None, 61, 61, 64)        256
 chNormalization)

 max_pooling2d_1 (MaxPoolin  (None, 30, 30, 64)        0
 g2D)

 conv2d_2 (Conv2D)           (None, 28, 28, 128)       73856

 batch_normalization_2 (Bat  (None, 28, 28, 128)       512
 chNormalization)

 max_pooling2d_2 (MaxPoolin  (None, 14, 14, 128)       0
 g2D)

 flatten (Flatten)           (None, 25088)             0

 dense (Dense)               (None, 128)               3211392

 batch_normalization_3 (Bat  (None, 128)               512
 chNormalization)

 dropout (Dropout)           (None, 128)               0

 dense_1 (Dense)             (None, 20)                2580

=================================================================
Total params: 3308052 (12.62 MB)
Trainable params: 3307348 (12.62 MB)
Non-trainable params: 704 (2.75 KB)
_________________________________________________________________
Epoch 1/30
42/42 [==============================] - 18s 390ms/step - loss: 3.2704 - accuracy: 0.1749 - val_loss: 3.9951 - val_accuracy: 0.1138
Epoch 2/30
42/42 [==============================] - 16s 378ms/step - loss: 2.3110 - accuracy: 0.3514 - val_loss: 4.0660 - val_accuracy: 0.1138
Epoch 3/30
42/42 [==============================] - 16s 376ms/step - loss: 1.8655 - accuracy: 0.4505 - val_loss: 4.2967 - val_accuracy: 0.1138
Epoch 4/30
42/42 [==============================] - 16s 387ms/step - loss: 1.4917 - accuracy: 0.5608 - val_loss: 4.7359 - val_accuracy: 0.1138
Epoch 5/30
42/42 [==============================] - 16s 377ms/step - loss: 1.3778 - accuracy: 0.5871 - val_loss: 4.0181 - val_accuracy: 0.1138
Epoch 6/30
42/42 [==============================] - 16s 380ms/step - loss: 1.0486 - accuracy: 0.6824 - val_loss: 3.6105 - val_accuracy: 0.1138
Epoch 7/30
42/42 [==============================] - 16s 379ms/step - loss: 0.7108 - accuracy: 0.7928 - val_loss: 3.5313 - val_accuracy: 0.1228
Epoch 8/30
42/42 [==============================] - 16s 383ms/step - loss: 0.5380 - accuracy: 0.8514 - val_loss: 3.4002 - val_accuracy: 0.1018
Epoch 9/30
42/42 [==============================] - 16s 377ms/step - loss: 0.3869 - accuracy: 0.9129 - val_loss: 3.1789 - val_accuracy: 0.1497
Epoch 10/30
42/42 [==============================] - 16s 380ms/step - loss: 0.2642 - accuracy: 0.9429 - val_loss: 2.9772 - val_accuracy: 0.2006
Epoch 11/30
42/42 [==============================] - 16s 370ms/step - loss: 0.1914 - accuracy: 0.9572 - val_loss: 2.9771 - val_accuracy: 0.1946
Epoch 12/30
42/42 [==============================] - 15s 367ms/step - loss: 0.1280 - accuracy: 0.9760 - val_loss: 2.5955 - val_accuracy: 0.2814
Epoch 13/30
42/42 [==============================] - 16s 373ms/step - loss: 0.0784 - accuracy: 0.9947 - val_loss: 2.2554 - val_accuracy: 0.3533
Epoch 14/30
42/42 [==============================] - 16s 371ms/step - loss: 0.0617 - accuracy: 0.9940 - val_loss: 2.0217 - val_accuracy: 0.4551
Epoch 15/30
42/42 [==============================] - 16s 380ms/step - loss: 0.0510 - accuracy: 0.9932 - val_loss: 1.9414 - val_accuracy: 0.4551
Epoch 16/30
42/42 [==============================] - 16s 376ms/step - loss: 0.0385 - accuracy: 0.9970 - val_loss: 1.5933 - val_accuracy: 0.5359
Epoch 17/30
42/42 [==============================] - 16s 374ms/step - loss: 0.0313 - accuracy: 0.9977 - val_loss: 1.5307 - val_accuracy: 0.5778
Epoch 18/30
42/42 [==============================] - 16s 377ms/step - loss: 0.0289 - accuracy: 0.9962 - val_loss: 1.5599 - val_accuracy: 0.5659
Epoch 19/30
42/42 [==============================] - 15s 369ms/step - loss: 0.0198 - accuracy: 0.9992 - val_loss: 1.4218 - val_accuracy: 0.5719
Epoch 20/30
42/42 [==============================] - 15s 365ms/step - loss: 0.0163 - accuracy: 1.0000 - val_loss: 1.4809 - val_accuracy: 0.5898
Epoch 21/30
42/42 [==============================] - 16s 380ms/step - loss: 0.0146 - accuracy: 1.0000 - val_loss: 1.4288 - val_accuracy: 0.6108
Epoch 22/30
42/42 [==============================] - 16s 382ms/step - loss: 0.0128 - accuracy: 0.9992 - val_loss: 1.4997 - val_accuracy: 0.5599
Epoch 23/30
42/42 [==============================] - 16s 373ms/step - loss: 0.0107 - accuracy: 1.0000 - val_loss: 1.4241 - val_accuracy: 0.6108
Epoch 24/30
42/42 [==============================] - 16s 370ms/step - loss: 0.0111 - accuracy: 1.0000 - val_loss: 1.4185 - val_accuracy: 0.6198
Epoch 25/30
42/42 [==============================] - 16s 374ms/step - loss: 0.0117 - accuracy: 0.9992 - val_loss: 1.4285 - val_accuracy: 0.6138
Epoch 26/30
42/42 [==============================] - 16s 374ms/step - loss: 0.0126 - accuracy: 0.9992 - val_loss: 1.3987 - val_accuracy: 0.6198
Epoch 27/30
42/42 [==============================] - 15s 363ms/step - loss: 0.0101 - accuracy: 1.0000 - val_loss: 1.4256 - val_accuracy: 0.6108
Epoch 28/30
42/42 [==============================] - 15s 368ms/step - loss: 0.0072 - accuracy: 1.0000 - val_loss: 1.4781 - val_accuracy: 0.6198
Epoch 29/30
42/42 [==============================] - 16s 373ms/step - loss: 0.0056 - accuracy: 1.0000 - val_loss: 1.4916 - val_accuracy: 0.5928
Epoch 30/30
42/42 [==============================] - 15s 365ms/step - loss: 0.0058 - accuracy: 1.0000 - val_loss: 1.4923 - val_accuracy: 0.6108
<keras.src.callbacks.History object at 0x000001AE3B2FA990>
```



In [4]:
```python
# Complex model
from sklearn.model_selection import KFold
from sklearn.metrics import confusion_matrix, ConfusionMatrixDisplay
import seaborn as sns
# Complex model
# K-Fold Cross-Validation
kfold = KFold(n_splits=5, shuffle=True, random_state=4)
accuracies = []
confusion_matrices = []

for train_index, val_index in kfold.split(birds):
    x_train_fold, x_val_fold = birds[train_index], birds[val_index]
    y_train_fold, y_val_fold = labels[train_index], labels[val_index]

    x_train_fold = x_train_fold.astype("float32") / 255
    x_val_fold = x_val_fold.astype("float32") / 255

    y_train_fold = keras.utils.to_categorical(y_train_fold, num_classes)
    y_val_fold = keras.utils.to_categorical(y_val_fold, num_classes)

    model = keras.Sequential([
        keras.Input(shape=input_shape),
        layers.Conv2D(32, kernel_size=(3, 3), activation="relu"),
        layers.BatchNormalization(),
        layers.MaxPooling2D(pool_size=(2, 2)),
        layers.Conv2D(64, kernel_size=(3, 3), activation="relu"),
        layers.BatchNormalization(),
        layers.MaxPooling2D(pool_size=(2, 2)),
        layers.Conv2D(128, kernel_size=(3, 3), activation="relu"),
        layers.BatchNormalization(),
        layers.MaxPooling2D(pool_size=(2, 2)),
        layers.Flatten(),
        layers.Dense(128, activation="relu"),
        layers.BatchNormalization(),
        layers.Dropout(0.5),
        layers.Dense(num_classes, activation="softmax"),
    ])

    model.compile(loss="categorical_crossentropy", optimizer="adam", metrics=["accuracy"])

    history = model.fit(x_train_fold, y_train_fold, batch_size=32, epochs=30, validation_data=(x_val_fold, y_val_fold), callbacks=[early_stopping])

    scores = model.evaluate(x_val_fold, y_val_fold, verbose=0)
    accuracies.append(scores[1])
    # Predict the labels for the validation set
    y_pred_fold = model.predict(x_val_fold)
    y_pred_fold = np.argmax(y_pred_fold, axis=1)
    y_val_fold = np.argmax(y_val_fold, axis=1)

    # Compute the confusion matrix
    cm = confusion_matrix(y_val_fold, y_pred_fold)
    confusion_matrices.append(cm)
    print(f"Fold accuracy: {scores[1]}")

print(f"Mean cross-validated accuracy: {np.mean(accuracies)}")

# Plot the training history of the last fold
plt.plot(history.history['accuracy'])
plt.plot(history.history['val_accuracy'])
plt.title('Model accuracy')
plt.ylabel('Accuracy')
plt.xlabel('Epoch')
plt.legend(['Train', 'Validation'], loc='lower right')
plt.show()


# Ensure labelencoder.classes_ contains the correct bird names
bird_names = [
    'Bent Beak Riffraff', 'Blue collared Zipper', 'Bombadil', 'Broad winged Jojo',
    'Canadian Cootamum', 'Carries Champagne Pipit', 'Darkwing Sparrow', 'Eastern Corn Skeet',
    'Green tipped Scarlet Pipit', 'Lesser Birchbeere', 'Orange Pine Plover', 'Ordinary Snape',
    'Pinkfinch', 'Purple Tooting Tout', 'Qax', 'Queenscoat', 'Rose crested Blue Pipit',
    'Scrawny Jay', 'Vermillion Trillian', 'Unknown'
]

# Assign bird names to labelencoder.classes_
labelencoder.classes_ = np.array(bird_names)

# Plot the confusion matrix of the last fold using seaborn
plt.figure(figsize=(10, 8))
sns.heatmap(confusion_matrices[-1], annot=True, fmt='d', cmap='Blues', xticklabels=labelencoder.classes_, yticklabels=labelencoder.classes_)
plt.title('Confusion Matrix')
plt.xlabel('Predicted Label')
plt.ylabel('True Label')
plt.show()
```

```
Epoch 1/30
53/53 [==============================] - 21s 385ms/step - loss: 3.1343 - accuracy: 0.2011 - val_loss: 3.8668 - val_accuracy: 0.1415
Epoch 2/30
53/53 [==============================] - 19s 368ms/step - loss: 2.2408 - accuracy: 0.3565 - val_loss: 4.2851 - val_accuracy: 0.1415
Epoch 3/30
53/53 [==============================] - 20s 374ms/step - loss: 1.7415 - accuracy: 0.4940 - val_loss: 4.1240 - val_accuracy: 0.1415
Epoch 4/30
53/53 [==============================] - 20s 373ms/step - loss: 1.4310 - accuracy: 0.5900 - val_loss: 3.7771 - val_accuracy: 0.1439
Epoch 5/30
53/53 [==============================] - 21s 390ms/step - loss: 1.4873 - accuracy: 0.5498 - val_loss: 3.2694 - val_accuracy: 0.1775
Epoch 6/30
53/53 [==============================] - 20s 387ms/step - loss: 1.1894 - accuracy: 0.6303 - val_loss: 3.3458 - val_accuracy: 0.1439
Epoch 7/30
53/53 [==============================] - 20s 382ms/step - loss: 0.9433 - accuracy: 0.7179 - val_loss: 3.1965 - val_accuracy: 0.1799
Epoch 8/30
53/53 [==============================] - 20s 383ms/step - loss: 0.6622 - accuracy: 0.8145 - val_loss: 3.0012 - val_accuracy: 0.1871
Epoch 9/30
53/53 [==============================] - 20s 383ms/step - loss: 0.5910 - accuracy: 0.8379 - val_loss: 2.6114 - val_accuracy: 0.2326
Epoch 10/30
53/53 [==============================] - 20s 384ms/step - loss: 0.4215 - accuracy: 0.8830 - val_loss: 2.2815 - val_accuracy: 0.2950
Epoch 11/30
53/53 [==============================] - 20s 386ms/step - loss: 0.3008 - accuracy: 0.9352 - val_loss: 2.0329 - val_accuracy: 0.3765
Epoch 12/30
53/53 [==============================] - 20s 374ms/step - loss: 0.3449 - accuracy: 0.9082 - val_loss: 1.5175 - val_accuracy: 0.5300
Epoch 13/30
53/53 [==============================] - 20s 373ms/step - loss: 0.2270 - accuracy: 0.9490 - val_loss: 1.4253 - val_accuracy: 0.5683
Epoch 14/30
53/53 [==============================] - 22s 423ms/step - loss: 0.2311 - accuracy: 0.9472 - val_loss: 1.7982 - val_accuracy: 0.4388
Epoch 15/30
53/53 [==============================] - 25s 470ms/step - loss: 0.1448 - accuracy: 0.9724 - val_loss: 1.5116 - val_accuracy: 0.5084
Epoch 16/30
53/53 [==============================] - 21s 403ms/step - loss: 0.1072 - accuracy: 0.9802 - val_loss: 1.6813 - val_accuracy: 0.5276
Epoch 17/30
53/53 [==============================] - 23s 434ms/step - loss: 0.1276 - accuracy: 0.9784 - val_loss: 1.4021 - val_accuracy: 0.5971
Epoch 18/30
53/53 [==============================] - 23s 429ms/step - loss: 0.0952 - accuracy: 0.9838 - val_loss: 1.3323 - val_accuracy: 0.6115
Epoch 19/30
53/53 [==============================] - 21s 402ms/step - loss: 0.0631 - accuracy: 0.9898 - val_loss: 1.2876 - val_accuracy: 0.6259
Epoch 20/30
53/53 [==============================] - 20s 385ms/step - loss: 0.0422 - accuracy: 0.9964 - val_loss: 1.2738 - val_accuracy: 0.6403
Epoch 21/30
53/53 [==============================] - 22s 408ms/step - loss: 0.0330 - accuracy: 0.9982 - val_loss: 1.4468 - val_accuracy: 0.6259
Epoch 22/30
53/53 [==============================] - 21s 399ms/step - loss: 0.0308 - accuracy: 0.9982 - val_loss: 1.4292 - val_accuracy: 0.6307
Epoch 23/30
53/53 [==============================] - 21s 391ms/step - loss: 0.0432 - accuracy: 0.9940 - val_loss: 1.5293 - val_accuracy: 0.5947
Epoch 24/30
53/53 [==============================] - 22s 408ms/step - loss: 0.3757 - accuracy: 0.8908 - val_loss: 3.4266 - val_accuracy: 0.2782
Epoch 25/30
53/53 [==============================] - 20s 384ms/step - loss: 0.2151 - accuracy: 0.9418 - val_loss: 1.7890 - val_accuracy: 0.4724
14/14 [==============================] - 1s 66ms/step
Fold accuracy: 0.6402877569198608
Epoch 1/30
53/53 [==============================] - 22s 403ms/step - loss: 3.0469 - accuracy: 0.2209 - val_loss: 4.8622 - val_accuracy: 0.0911
Epoch 2/30
53/53 [==============================] - 22s 408ms/step - loss: 2.0441 - accuracy: 0.4088 - val_loss: 7.6056 - val_accuracy: 0.0911
Epoch 3/30
53/53 [==============================] - 22s 423ms/step - loss: 1.6866 - accuracy: 0.5006 - val_loss: 11.0941 - val_accuracy: 0.0911
Epoch 4/30
53/53 [==============================] - 21s 404ms/step - loss: 1.2783 - accuracy: 0.6261 - val_loss: 11.4796 - val_accuracy: 0.0911
Epoch 5/30
53/53 [==============================] - 22s 416ms/step - loss: 0.9716 - accuracy: 0.7083 - val_loss: 13.0121 - val_accuracy: 0.0911
Epoch 6/30
53/53 [==============================] - 22s 417ms/step - loss: 1.0617 - accuracy: 0.6735 - val_loss: 11.4910 - val_accuracy: 0.0911
14/14 [==============================] - 1s 74ms/step
Fold accuracy: 0.09112709760665894
Epoch 1/30
53/53 [==============================] - 24s 438ms/step - loss: 3.0332 - accuracy: 0.2233 - val_loss: 3.6905 - val_accuracy: 0.1055
Epoch 2/30
53/53 [==============================] - 21s 387ms/step - loss: 2.0818 - accuracy: 0.4142 - val_loss: 3.5471 - val_accuracy: 0.1055
Epoch 3/30
53/53 [==============================] - 21s 392ms/step - loss: 1.7268 - accuracy: 0.4982 - val_loss: 3.8307 - val_accuracy: 0.1055
Epoch 4/30
53/53 [==============================] - 21s 393ms/step - loss: 1.3892 - accuracy: 0.5810 - val_loss: 3.5618 - val_accuracy: 0.1055
Epoch 5/30
53/53 [==============================] - 22s 419ms/step - loss: 1.0686 - accuracy: 0.6687 - val_loss: 3.2439 - val_accuracy: 0.1055
Epoch 6/30
53/53 [==============================] - 21s 392ms/step - loss: 0.9131 - accuracy: 0.7215 - val_loss: 2.8272 - val_accuracy: 0.2278
Epoch 7/30
53/53 [==============================] - 22s 421ms/step - loss: 1.2428 - accuracy: 0.6134 - val_loss: 2.8040 - val_accuracy: 0.1535
Epoch 8/30
53/53 [==============================] - 21s 391ms/step - loss: 1.1549 - accuracy: 0.6519 - val_loss: 2.6380 - val_accuracy: 0.1391
Epoch 9/30
53/53 [==============================] - 21s 397ms/step - loss: 0.9023 - accuracy: 0.7329 - val_loss: 2.1847 - val_accuracy: 0.3237
Epoch 10/30
53/53 [==============================] - 22s 410ms/step - loss: 0.6209 - accuracy: 0.8205 - val_loss: 1.9595 - val_accuracy: 0.3813
Epoch 11/30
53/53 [==============================] - 21s 397ms/step - loss: 0.4713 - accuracy: 0.8745 - val_loss: 1.8413 - val_accuracy: 0.4053
Epoch 12/30
53/53 [==============================] - 20s 383ms/step - loss: 0.3362 - accuracy: 0.9244 - val_loss: 1.7396 - val_accuracy: 0.4508
Epoch 13/30
53/53 [==============================] - 21s 390ms/step - loss: 0.2698 - accuracy: 0.9412 - val_loss: 1.5629 - val_accuracy: 0.5132
Epoch 14/30
53/53 [==============================] - 21s 391ms/step - loss: 0.2063 - accuracy: 0.9580 - val_loss: 1.5134 - val_accuracy: 0.5108
Epoch 15/30
53/53 [==============================] - 22s 417ms/step - loss: 0.1863 - accuracy: 0.9568 - val_loss: 1.4898 - val_accuracy: 0.5372
Epoch 16/30
53/53 [==============================] - 22s 411ms/step - loss: 0.1747 - accuracy: 0.9622 - val_loss: 1.4781 - val_accuracy: 0.5396
Epoch 17/30
53/53 [==============================] - 21s 405ms/step - loss: 0.1226 - accuracy: 0.9808 - val_loss: 1.4490 - val_accuracy: 0.5875
Epoch 18/30
53/53 [==============================] - 22s 424ms/step - loss: 0.0902 - accuracy: 0.9844 - val_loss: 1.4477 - val_accuracy: 0.5755
Epoch 19/30
53/53 [==============================] - 21s 397ms/step - loss: 0.0722 - accuracy: 0.9928 - val_loss: 1.4890 - val_accuracy: 0.5755
Epoch 20/30
53/53 [==============================] - 22s 409ms/step - loss: 0.5020 - accuracy: 0.8487 - val_loss: 4.5295 - val_accuracy: 0.2542
Epoch 21/30
53/53 [==============================] - 21s 402ms/step - loss: 0.2280 - accuracy: 0.9484 - val_loss: 3.4259 - val_accuracy: 0.3022
Epoch 22/30
53/53 [==============================] - 20s 371ms/step - loss: 0.1656 - accuracy: 0.9610 - val_loss: 1.8523 - val_accuracy: 0.5204
Epoch 23/30
53/53 [==============================] - 21s 394ms/step - loss: 0.1225 - accuracy: 0.9754 - val_loss: 1.7309 - val_accuracy: 0.5516
14/14 [==============================] - 1s 72ms/step
Fold accuracy: 0.5755395889282227
Epoch 1/30
53/53 [==============================] - 23s 406ms/step - loss: 3.0513 - accuracy: 0.2166 - val_loss: 3.9937 - val_accuracy: 0.0385
Epoch 2/30
53/53 [==============================] - 21s 399ms/step - loss: 2.1692 - accuracy: 0.3905 - val_loss: 3.8521 - val_accuracy: 0.1322
Epoch 3/30
53/53 [==============================] - 21s 397ms/step - loss: 1.7901 - accuracy: 0.4841 - val_loss: 3.4302 - val_accuracy: 0.1322
Epoch 4/30
53/53 [==============================] - 20s 382ms/step - loss: 1.6180 - accuracy: 0.5369 - val_loss: 3.2806 - val_accuracy: 0.1322
Epoch 5/30
53/53 [==============================] - 21s 404ms/step - loss: 1.2637 - accuracy: 0.6275 - val_loss: 3.0480 - val_accuracy: 0.1322
Epoch 6/30
53/53 [==============================] - 21s 395ms/step - loss: 1.1393 - accuracy: 0.6665 - val_loss: 2.9735 - val_accuracy: 0.1587
Epoch 7/30
53/53 [==============================] - 21s 393ms/step - loss: 1.2121 - accuracy: 0.6239 - val_loss: 2.4840 - val_accuracy: 0.2716
Epoch 8/30
53/53 [==============================] - 20s 379ms/step - loss: 0.8941 - accuracy: 0.7295 - val_loss: 2.4586 - val_accuracy: 0.2067
Epoch 9/30
53/53 [==============================] - 21s 390ms/step - loss: 0.7041 - accuracy: 0.8038 - val_loss: 2.2256 - val_accuracy: 0.3510
Epoch 10/30
53/53 [==============================] - 21s 397ms/step - loss: 0.5390 - accuracy: 0.8512 - val_loss: 2.1513 - val_accuracy: 0.3558
Epoch 11/30
53/53 [==============================] - 20s 383ms/step - loss: 0.9482 - accuracy: 0.7067 - val_loss: 2.4015 - val_accuracy: 0.2981
Epoch 12/30
53/53 [==============================] - 21s 392ms/step - loss: 0.6021 - accuracy: 0.8266 - val_loss: 2.1386 - val_accuracy: 0.3582
Epoch 13/30
53/53 [==============================] - 20s 379ms/step - loss: 0.4674 - accuracy: 0.8830 - val_loss: 1.7563 - val_accuracy: 0.4591
Epoch 14/30
53/53 [==============================] - 19s 362ms/step - loss: 0.3156 - accuracy: 0.9262 - val_loss: 1.5328 - val_accuracy: 0.5625
Epoch 15/30
53/53 [==============================] - 20s 379ms/step - loss: 0.2463 - accuracy: 0.9406 - val_loss: 1.6530 - val_accuracy: 0.5240
Epoch 16/30
53/53 [==============================] - 20s 369ms/step - loss: 0.1893 - accuracy: 0.9664 - val_loss: 1.5038 - val_accuracy: 0.5649
Epoch 17/30
53/53 [==============================] - 20s 383ms/step - loss: 0.2034 - accuracy: 0.9538 - val_loss: 1.7858 - val_accuracy: 0.5120
Epoch 18/30
53/53 [==============================] - 20s 378ms/step - loss: 0.2033 - accuracy: 0.9562 - val_loss: 1.8658 - val_accuracy: 0.4928
Epoch 19/30
53/53 [==============================] - 21s 395ms/step - loss: 0.1494 - accuracy: 0.9706 - val_loss: 1.7280 - val_accuracy: 0.5072
Epoch 20/30
53/53 [==============================] - 20s 385ms/step - loss: 0.1196 - accuracy: 0.9790 - val_loss: 1.6123 - val_accuracy: 0.5361
Epoch 21/30
53/53 [==============================] - 21s 393ms/step - loss: 0.0862 - accuracy: 0.9874 - val_loss: 1.7479 - val_accuracy: 0.5385
13/13 [==============================] - 1s 69ms/step
Fold accuracy: 0.5649038553237915
Epoch 1/30
53/53 [==============================] - 22s 390ms/step - loss: 3.2215 - accuracy: 0.1764 - val_loss: 3.7861 - val_accuracy: 0.0288
Epoch 2/30
53/53 [==============================] - 21s 391ms/step - loss: 2.3126 - accuracy: 0.3503 - val_loss: 4.0193 - val_accuracy: 0.1082
Epoch 3/30
53/53 [==============================] - 20s 372ms/step - loss: 1.8204 - accuracy: 0.4739 - val_loss: 4.0976 - val_accuracy: 0.1082
Epoch 4/30
53/53 [==============================] - 20s 380ms/step - loss: 1.4908 - accuracy: 0.5567 - val_loss: 3.7559 - val_accuracy: 0.1082
Epoch 5/30
53/53 [==============================] - 20s 373ms/step - loss: 1.3136 - accuracy: 0.5927 - val_loss: 3.1504 - val_accuracy: 0.1635
Epoch 6/30
53/53 [==============================] - 19s 358ms/step - loss: 1.0381 - accuracy: 0.6899 - val_loss: 3.3870 - val_accuracy: 0.0986
Epoch 7/30
53/53 [==============================] - 19s 360ms/step - loss: 0.8056 - accuracy: 0.7660 - val_loss: 2.9644 - val_accuracy: 0.1082
Epoch 8/30
53/53 [==============================] - 20s 370ms/step - loss: 1.2031 - accuracy: 0.6263 - val_loss: 2.9665 - val_accuracy: 0.2091
Epoch 9/30
53/53 [==============================] - 19s 364ms/step - loss: 0.7517 - accuracy: 0.7768 - val_loss: 3.1005 - val_accuracy: 0.1106
Epoch 10/30
53/53 [==============================] - 20s 379ms/step - loss: 0.5996 - accuracy: 0.8212 - val_loss: 2.5687 - val_accuracy: 0.2163
Epoch 11/30
53/53 [==============================] - 19s 367ms/step - loss: 0.3958 - accuracy: 0.8974 - val_loss: 2.1850 - val_accuracy: 0.4111
Epoch 12/30
53/53 [==============================] - 20s 372ms/step - loss: 0.3541 - accuracy: 0.9130 - val_loss: 1.8196 - val_accuracy: 0.4495
Epoch 13/30
53/53 [==============================] - 20s 381ms/step - loss: 0.3025 - accuracy: 0.9262 - val_loss: 1.6133 - val_accuracy: 0.5000
Epoch 14/30
53/53 [==============================] - 20s 382ms/step - loss: 0.2237 - accuracy: 0.9526 - val_loss: 1.5226 - val_accuracy: 0.5385
Epoch 15/30
53/53 [==============================] - 19s 367ms/step - loss: 0.2059 - accuracy: 0.9484 - val_loss: 1.3593 - val_accuracy: 0.5913
Epoch 16/30
53/53 [==============================] - 21s 389ms/step - loss: 0.1613 - accuracy: 0.9670 - val_loss: 1.6026 - val_accuracy: 0.5385
Epoch 17/30
53/53 [==============================] - 19s 361ms/step - loss: 0.0935 - accuracy: 0.9862 - val_loss: 1.4096 - val_accuracy: 0.5841
Epoch 18/30
53/53 [==============================] - 19s 361ms/step - loss: 0.2821 - accuracy: 0.9316 - val_loss: 2.3867 - val_accuracy: 0.4135
Epoch 19/30
53/53 [==============================] - 19s 360ms/step - loss: 0.1493 - accuracy: 0.9664 - val_loss: 1.8219 - val_accuracy: 0.5168
Epoch 20/30
53/53 [==============================] - 19s 366ms/step - loss: 0.1222 - accuracy: 0.9718 - val_loss: 1.5594 - val_accuracy: 0.5481
13/13 [==============================] - 1s 65ms/step
Fold accuracy: 0.5913461446762085
Mean cross-validated accuracy: 0.49264088869094846
```





In [ ]: