Model Optimization and Tuning Phase Template

Date	20 March 2024
Team ID	SWTID1720165000
Project Title	CovidVision: Advanced COVID-19 Detection From Lung X-Rays With Deep Learning
Maximum Marks	10 Marks

Model Optimization and Tuning Phase

The Model Optimization and Tuning Phase involves refining neural network models for peak performance. It includes optimized model code, fine-tuning hyperparameters, comparing performance metrics, and justifying the final model selection for enhanced predictive accuracy and efficiency.

Hyperparameter Tuning Documentation (8 Marks):

Model	Tuned Hyperparameters
Model 1 EfficientNetB0	

```
# Image Data Generators
train_datagen = ImageDataGenerator(
    rescale=1./255,
    shear_range=0.2,
    zoom_range=0.2,
    horizontal_flip=True
)

test_datagen = ImageDataGenerator(rescale=1./255)

train_generator = train_datagen.flow_from_directory(
    train_dir,
    target_size=(224, 224),
    batch_size=32,
    class_mode='categorical'
)

validation_generator = test_datagen.flow_from_directory(
    test_dir,
    target_size=(224, 224),
    batch_size=32,
    class_mode='categorical'
)

# Load EfficientNetB0 model
base_model = EfficientNetB0(weights='imagenet', include_top=False, input_shape=(224, 224, 3))
```

<table>
<tr>
<td></td>
<td>

```
# Add custom layers on top of EfficientNetB0
x = base_model.output
x = GlobalAveragePooling2D()(x)
x = Dense(1024, activation='relu')(x)
predictions = Dense(4, activation='softmax')(x)

# Final model
model = Model(inputs=base_model.input, outputs=predictions)

# Compile the model
model.compile(optimizer=Adam(learning_rate=0.0001), loss='categorical_crossentropy', metrics=['accuracy'])

# Train the model
model.fit(
    train_generator,
    epochs=10,
    validation_data=validation_generator
)
```

- Epochs: The number of times the entire dataset is passed through the neural network during training.

- Rescale: A factor used to normalize the input data, usually scaling pixel values to a range between 0 and 1.

- Zoom: A data augmentation technique that randomly zooms into images during training to improve model robustness.

- Shear: A data augmentation technique that applies a shearing transformation to images, tilting them along one axis.

- Batch Size: The number of training samples processed before the model's internal parameters are updated.
</td>
</tr>
<tr>
<td>Model 2

xception</td>
<td>

```
Xception.fit(train,validation_data=test,epochs=15, batch_size = 16)
```

```
train_datagen = ImageDataGenerator(rescale=1./255,zoom_range=0.2,shear_range=0.2)
test_datagen = ImageDataGenerator(rescale=1./255)
```

Python

```
train = train_datagen.flow_from_directory(trainPath,target_size=(140,140),batch_size=16)
test = test_datagen.flow_from_directory(testPath,target_size=(140,140),batch_size=16)
```

Python

- Epochs: The number of times the entire dataset is passed through the neural network during training.
</td>
</tr>
</table>

	- Rescale: A factor used to normalize the input data, usually scaling pixel values to a range between 0 and 1. - Zoom: A data augmentation technique that randomly zooms into images during training to improve model robustness. - Shear: A data augmentation technique that applies a shearing transformation to images, tilting them along one axis. - Batch Size: The number of training samples processed before the model's internal parameters are updated.
Model 3 Inception	 - Epochs: The number of times the entire dataset is passed through the neural network during training.

```
# Image Data Generators (same as before)
train_datagen = ImageDataGenerator(
    rescale=1./255,
    shear_range=0.2,
    zoom_range=0.2,
    horizontal_flip=True
)

test_datagen = ImageDataGenerator(rescale=1./255)

train_generator = train_datagen.flow_from_directory(
    train_dir,
    target_size=(299, 299),  # InceptionV3 requires input size of 299x299
    batch_size=32,
    class_mode='categorical'
)

validation_generator = test_datagen.flow_from_directory(
    test_dir,
    target_size=(299, 299),
    batch_size=32,
    class_mode='categorical'
)

# Load InceptionV3 model
base_model = InceptionV3(weights='imagenet', include_top=False, input_shape=(299, 299, 3))

# Add custom layers on top of InceptionV3
x = base_model.output
x = GlobalAveragePooling2D()(x)
x = Dense(1024, activation='relu')(x)
predictions = Dense(4, activation='softmax')(x)  # 4 classes for your dataset

# Final model
model = Model(inputs=base_model.input, outputs=predictions)

# Compile the model
model.compile(optimizer=Adam(learning_rate=0.0001), loss='categorical_crossentropy', metrics=['accuracy'])
```

	- Rescale: A factor used to normalize the input data, usually scaling pixel values to a range between 0 and 1. - Zoom: A data augmentation technique that randomly zooms into images during training to improve model robustness. - Shear: A data augmentation technique that applies a shearing transformation to images, tilting them along one axis. - Batch Size: The number of training samples processed before the model's internal parameters are updated.
Model 4 VGG16	 - Epochs: The number of times the entire dataset is passed through the neural network during training.

```
# Image Data Generators (same as before)
train_datagen = ImageDataGenerator(
    rescale=1./255,
    shear_range=0.2,
    zoom_range=0.2,
    horizontal_flip=True
)

test_datagen = ImageDataGenerator(rescale=1./255)

train_generator = train_datagen.flow_from_directory(
    train_dir,
    target_size=(299, 299),  # InceptionV3 requires input size of 299x299
    batch_size=32,
    class_mode='categorical'
)

validation_generator = test_datagen.flow_from_directory(
    test_dir,
    target_size=(299, 299),
    batch_size=32,
    class_mode='categorical'
)

# Load InceptionV3 model
base_model = InceptionV3(weights='imagenet', include_top=False, input_shape=(299, 299, 3))

# Add custom layers on top of InceptionV3
x = base_model.output
x = GlobalAveragePooling2D()(x)
x = Dense(1024, activation='relu')(x)
predictions = Dense(4, activation='softmax')(x)  # 4 classes for your dataset

# Final model
model = Model(inputs=base_model.input, outputs=predictions)

# Compile the model
model.compile(optimizer=Adam(learning_rate=0.0001), loss='categorical_crossentropy', metrics=['accuracy'])
```

	- Rescale: A factor used to normalize the input data, usually scaling pixel values to a range between 0 and 1. - Zoom: A data augmentation technique that randomly zooms into images during training to improve model robustness. - Shear: A data augmentation technique that applies a shearing transformation to images, tilting them along one axis. - Batch Size: The number of training samples processed before the model's internal parameters are updated.

Final Model Selection Justification (2 Marks):

Final Model	Reasoning
Model1 (EfficientNetB0)	We finalized with EfficientNetB0 because it gave an accuracy of 87.5%