SMARTBRIDGE Let's Bridge the Gap a Veranda Enterprise

Smart Internz

Model Optimization and Tuning Phase Template

Date	4 June 2024
Team ID	SWTID1720109344
Project Title	Rice Type Classification using CNN
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
VGG16	**VGG16 Model Hyperparameters:** **learning_rate:** The learning rate for the Adam optimizer. Controls the step size at each iteration while moving toward a minimum of a loss function. Higher values can lead to faster convergence but may overshoot the minimum. **dropout_rate:** The dropout rate for the dense layer. Randomly sets input units to 0 with a frequency of dropout_rate at each step during training time, which helps prevent overfitting. **dense_units:** The number of units in the dense layer. Determines the dimensionality of the output space.

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

<pre><code># Defining the hyperparameters for VGG16
hyperparameters_vgg16 = {
    'learning_rate': [0.001, 0.01, 0.1],
    'dropout_rate': [0.1, 0.2, 0.3],
    'dense_units': [128, 256, 512]
}</code></pre>

</td></tr>
<tr><td>ResNet50</td><td>

ResNet50 Model Hyperparameters:

learning_rate: The learning rate for the Adam optimizer. Controls the step size at each iteration while moving toward a minimum of a loss function. Higher values can lead to faster convergence but may overshoot the minimum.

dropout_rate: The dropout rate for the dense layer. Randomly sets input units to 0 with a frequency of dropout_rate at each step during training time, which helps prevent overfitting.

dense_units: The number of units in the dense layer. Determines the dimensionality of the output space.

<pre><code># Defining the hyperparameters for ResNet50
hyperparameters_resnet50 = {
    'learning_rate': [0.001, 0.01, 0.1],
    'dropout_rate': [0.1, 0.2, 0.3],
    'dense_units': [128, 256, 512]
}</code></pre>

</td></tr>
<tr><td>Xception</td><td>Xception Model Hyperparameters:</td></tr>
</table>

	learning_rate: The learning rate for the Adam optimizer. Controls the step size at each iteration while moving toward a minimum of a loss function. Higher values can lead to faster convergence but may overshoot the minimum.
	
	dropout_rate: The dropout rate for the dense layer. Randomly sets input units to 0 with a frequency of dropout_rate at each step during training time, which helps prevent overfitting.
	
	dense_units: The number of units in the dense layer. Determines the dimensionality of the output space.
	
	`# Define the hyperparameters for Xception`
	`hyperparameters_xception = {`
	`    'learning_rate': [0.001, 0.01, 0.1],`
	`    'dropout_rate': [0.1, 0.2, 0.3],`
	`    'dense_units': [128, 256, 512]`
	`}`

Final Model Selection Justification (2 Marks):

Final Model	Reasoning
Xception	**Highest Validation Accuracy** The Xception model achieved the highest validation accuracy among the three models. After tuning the hyperparameters, the Xception model reached a validation accuracy of 92.5%, while VGG16 achieved 90.2%

	and ResNet50 achieved 91.8%. The higher validation accuracy of the Xception model indicates that it can better generalize to unseen data and make more accurate predictions. **Efficient Hyperparameter Tuning** The hyperparameter tuning process for the Xception model was able to find an optimal set of hyperparameters that significantly improved its performance. The tuned hyperparameters for Xception were: Learning rate: 0.01 Dropout rate: 0.2 Dense units: 256 This suggests that the Xception model architecture is well-suited for the rice classification task and the hyperparameter tuning process effectively optimized the model's performance.