Model Optimization and Tuning Phase Template

Date	15 March 2024
Team ID	team-739735
Project Title	Natural Disasters Intensity Analysis And Classification Using AI
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
CNN	• **Batch Size**: Batch Size: Defines the number of images processed in one training step. A balanced batch size helps achieve stable and efficient learning. **Example**: batch_size=32 • **Epochs**: Determines how many times the training dataset is iterated over. Sufficient epochs allow the model to learn patterns without overfitting. **Example**: epochs=25 • **Learning Rate**: Controls how quickly the model adapts to the problem. A well-chosen rate ensures convergence without skipping minima.. **Example**: learning_rate=0.0001 • **Optimizer**: Adam optimizer is used for its adaptive learning rate,

	which improves training efficiency and accuracy. **Example**: optimizer=Adam() • **Loss Function**: Measures how well the model's predictions match the true values. **Categorical cross-entropy** is widely used for multi-class classification. **Example**: loss='categorical_crossentropy' • model = Sequential() • model.add(Conv2D(32, (3, 3), activation='relu', input_shape=(128, 128, 3))) • model.add(MaxPooling2D((2, 2))) • model.add(Conv2D(64, (3, 3), activation='relu')) • model.add(MaxPooling2D((2, 2))) • model.add(Flatten()) • model.add(Dense(512, activation='relu')) • model.add(Dense(3, activation='softmax')) # 3 classes: mild, moderate, severe

Final Model Selection Justification (2 Marks):

Final Model	Reasoning
CNN	The CNN model effectively captures spatial patterns in satellite or aerial imagery related to natural disasters such as floods, wildfires, and earthquakes. It uses convolutional layers to extract meaningful features, followed by fully connected layers for classification. ReLU is applied

	in hidden layers and Softmax in the output layer for multi-class intensity prediction. Trained using categorical cross-entropy loss and the Adam optimizer, this architecture balances accuracy and computational efficiency, making it ideal for disaster intensity analysis and classification tasks..