Model Optimization and Tuning Phase Template

Date	19 July 2024
Team ID	SWTID1720012105
Project Title	WarLens: Transfer Learning for Event Classification in Conflict zones.
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	`# Train the model` `model.fit(` `  train_gen,` `  steps_per_epoch=len(train_paths) // batch_size,` `  validation_data=val_gen,` `  validation_steps=len(val_paths) // batch_size,` `  epochs=20` `)` `# Save the model` `model.save('war_lens_model_resnet50.h5')`

	`` # Create a new model on top model = Sequential([base_model, Flatten(), Dense(256, activation='relu'), Dense(len(label_to_index), activation='softmax') # Adjust the number of classes dynamically]) ```
Model 2	``` # Train the model history_new1 = model_new1.fit(train_gen, steps_per_epoch=len(train_paths) // batch_size, validation_data=val_gen, validation_steps=len(val_paths) // batch_size, epochs=20) # Save the model model_new1.save('war_lens_model_mobilenetv2.h5') ``` ``` # Create a new model on top model_new1 = Sequential([mobilenet_model, GlobalAveragePooling2D(), Dense(128, activation='relu'), Dropout(0.5), Dense(len(label_to_index), activation='softmax') # Adjust the number of classes dynamically]) # Compile the model ```

Final Model Selection Justification (2 Marks):

Final Model	Reasoning

Model 2	• **Model Efficiency:** • **ResNet50:** The ResNet50 model includes a Flatten layer, which results in a large number of parameters, potentially leading to overfitting and higher computational cost. • **MobileNetV2:** The MobileNetV2 model uses GlobalAveragePooling2D, which reduces the number of parameters and makes the model more efficient and less prone to overfitting. • **Regularization:** • **ResNet50:** No explicit regularization layer is added. • **MobileNetV2:** Includes a Dropout layer with a 50% drop rate, which helps prevent overfitting by randomly setting half of the units to zero during training. • **Model Complexity:** • **ResNet50:** The model is deeper and more complex, which can make it more challenging to train and tune properly. • **MobileNetV2:** The model is designed to be lightweight and efficient, making it easier to train and less likely to overfit, especially with limited data. • **Parameter Tuning:** • **ResNet50:** The dense layer with 256 units might not be optimal for your dataset, potentially leading to overfitting or underfitting. • **MobileNetV2:** The dense layer with 128 units, combined with dropout, strikes a balance between model complexity and generalization ability. **Epochs:** • Both models are trained for 20 epochs, which should be sufficient for convergence. However, MobileNetV2's efficient architecture might allow it to converge to a better minimum within the same number of epochs.