CHRIST (Deemed to be University)
Department of Computer Science
Master of Artificial Intelligence and Machine Learning

Course: MAI371 – Deep Learning
Exercise No: LAB Exercise – 6
Date: 01 – 04 – 2024
Duration: 2 Hrs

Question (10 Marks)

You are a data scientist working in a medical research institute specializing in dermatology. Skin cancer is a significant concern, and there's a pressing need to develop accurate and efficient methods for its detection. Deep learning models have shown promising results in automating this process, particularly convolutional neural networks (CNNs). Your task is to conduct a comparative study between two architectures: VGG16 or VGG19 and a custom CNN model specifically designed for skin cancer detection.You are provided with a dataset containing thousands of dermatoscopic images, each labeled as benign or malignant based on clinical diagnosis. Before diving into model development, you need to preprocess the data to ensure consistency and enhance its diversity. This includes standardizing image sizes, normalizing pixel values, and applying augmentation techniques like rotation, flipping, and scaling.

1. Data Preprocessing:
   - Standardize image sizes and normalize pixel values.
   - Apply augmentation techniques to increase data diversity.

2. Model Architectures:
   -VGG16 or VGG19
   - Custom CNN: Designed specifically for skin cancer detection, this model comprises:
   - Four convolutional layers with increasing filter sizes (32, 64, 128, 256) and ReLU activation.
   - Max pooling layers after each convolutional layer.
   - Two dense layers with 512 neurons each, followed by a dropout layer (dropout rate: 0.5).
   - Output layer with 1 neuron and sigmoid activation for binary classification.

3. Training:
   - Split the dataset into training, validation, and test sets.
   - For VGG models, employ transfer learning by initializing with pre-trained ImageNet weights and fine-tuning only the fully connected layers.
   - Train the custom CNN model from scratch.

- Use appropriate optimization algorithms (e.g., SGD, Adam) and monitor performance on the validation set for early stopping.

4. Evaluation:
  - Evaluate the trained models on the test set.
  - Calculate metrics including accuracy, precision, recall, and F1-score.
  - Visualize model predictions and analyze misclassifications.

4.Comparison:

  ● Visualize loss and accuracy curves during training for both architectures to analyze convergence patterns.
  ● Compare traditional metrics (accuracy, precision, recall, F1-score,ROC-AUC, specificity) between VGG and custom CNN models.
  ● Present ROC curves and confusion matrices to compare classification performance and error patterns.

Evaluation Rubrics:
- VGG Implementations **(3 Marks)**
- Custom CNN**(4 Marks)**
- Comparison**(3 Marks)**
Total:10 Marks
General Instruction:

1. Ensure that your code includes relevant comments to enhance readability and understanding. Subsequently, upload your code to GitHub for version control and collaborative access.
2. Include descriptive comments within the code, explaining its functionality and logic.
3. In the Google Classroom submission, include the GitHub URL where your code is hosted.
4. Attach a PDF document named "your_register_number_exercise_No.pdf" to the submission.The PDF document should include screenshots of the code and the output screen.
5. Upload the answer document&GitHub URL in Google Classroom on or before the deadline mentioned.Evaluation will not be considered for late submission