USN:1RVU22CSE153
NAME:SHREYAS R

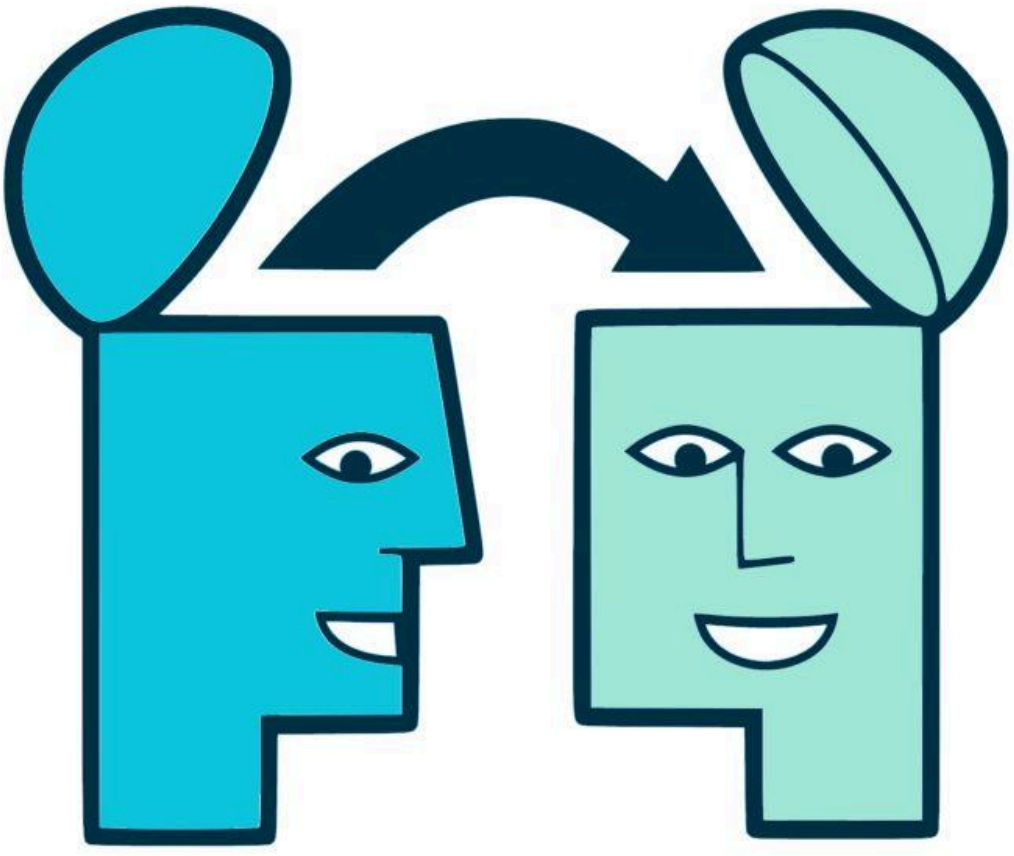
Title: Transfer Learning with CNN for Image Classification

Objective:

To implement transfer learning using a pre-trained Convolutional Neural Network (CNN) model to classify images into different categories.

Description:

In this experiment, we utilize a pre-trained Convolutional Neural Network (CNN) model for image classification through transfer learning. The model is fine-tuned on a specific dataset to classify images into predefined categories. The pre-trained model's weights serve as a starting point, and the network undergoes additional training to adapt to the new task



.

USN:1RVU22CSE153
NAME:SHREYAS R

Model

1. **Initialization**:
    - Load the pre-trained CNN model (e.g., VGG16, ResNet).
    - Initialize any additional layers or modifications.
2. **Data Preprocessing**:
    - Load and preprocess the dataset (e.g., resize images, normalize pixel values).
    - Split the data into training and testing sets.
3. **Model Fine-Tuning**:
    - Freeze the initial layers of the pre-trained model.
    - Train the model on the new dataset while updating the weights of the last few layers.
4. **Cost Function**:
    - Use an appropriate loss function (e.g., cross-entropy) to compute the error between predicted outputs and actual labels.
5. **Training**:
    - Train the model over several epochs, monitoring the training and validation accuracy.

Code Explanation:

1. **Model Setup:**
    - **Import Libraries:** Load TensorFlow/Keras modules.
    - **Load Pre-trained Model:** Use MobileNetV2 from TensorFlow Hub with frozen weights.
    - **Build Model:** Combine MobileNetV2 as a feature extractor with a final dense layer for flower classification.
2. **Data Preparation:**
    - **Image Shape:** Set images to 224x224 pixels.
    - **Download and Organize Dataset:** Obtain flower images and categorize them.
    - **Load, Resize, and Normalize Images:** Prepare the dataset by resizing and normalizing the pixel values.
3. **Training:**
    - **Split Data:** Divide the dataset into training and test sets.
    - **Compile and Train Model:** Use Adam optimizer with cross-entropy loss to train the model for 5 epochs.
4. **Evaluation:**
    - **Evaluate Model:** Test the model's accuracy and loss on the test dataset.
5. **Prediction:**
    - **Preprocess New Image:** Prepare a single image for prediction by resizing and normalizing.
    - **Predict Class:** Use the trained model to classify the image and map it to the corresponding flower name.

USN:1RVU22CSE153
NAME:SHREYAS R

Results:

- **Training Accuracy:**92.46%
- **Testing Accuracy:** 86.38%
- **Loss on Test Set:** 0.3853

Github Link: https://github.com/shreyasrajiv327/CS3232-DeepLearning/tree/main/Lab5