**Building a Retinal Vessel Classification System: CNN vs. Transfer Learning**

1. **Abstract**

n this project, we developed a deep learning-based system for classifying retinal vessel images into three categories: Diabetic Maculopathy, Macular Edema, and Normal. We implemented two approaches: a custom Convolutional Neural Network (CNN) designed from scratch, and a transfer learning method utilizing the ResNet50 model pre-trained on ImageNet. The dataset underwent preprocessing, including image resizing and normalization, with data augmentation techniques applied to improve model robustness. Both models were trained and evaluated, with the ResNet50 model demonstrating superior accuracy, benefiting from the transfer of learned features from a vast and diverse dataset. The models were deployed using a Flask API, allowing for practical application in real-world scenarios. This project demonstrates the effectiveness of transfer learning in medical image classification tasks, providing a robust solution for automated retinal disease detection.

1. **Introduction**

In this final project of my fellowship with Bytewise, I set out to develop a deep learning-based system for classifying retinal vessel images into three categories: Diabetic Maculopathy, Macular Edema, and Normal. This project represents the culmination of my learning and experience gained during the fellowship, where I focused on mastering both traditional Convolutional Neural Networks (CNN) and advanced techniques like transfer learning. I implemented a transfer learning approach using the pre-trained ResNet50 model, which achieved an accuracy of 47% on the validation set. However, to further improve performance, I developed a custom Convolutional Neural Network (CNN) from scratch, which significantly outperformed with an accuracy of 52%. This project not only allowed me to apply the theoretical knowledge acquired throughout the program but also to explore the practical challenges of working with real-world medical imaging data.

1. **Project Workflow**
   1. **Data Collection and Preprocessing**
      1. **Dataset**: The dataset includes retinal vessel images categorized into Diabetic Maculopathy, Macular Edema, and Normal. This dataset is very large. I just take 400 +400 + 400 images for each class. I have collected this dataset for my own project which has 32k pictures. I have converted it into binary almost 20k and 8k are classified into there on class.
      2. **Data Organization**: Images are organized into directories for training and validation, with subdirectories for each class.
      3. **Preprocessing**: Images were resized to 150x150 pixels and normalized. Data augmentation techniques were applied to the training set.
   2. **CNN Model**
      1. **Architecture**: A custom CNN was designed with several convolutional layers followed by max-pooling layers, flattening, and dense layers.
      2. **Training**: The CNN model was trained from scratch with the dataset.
   3. **Transfer Learning with ResNet50**
      1. **Base Model**: We used the ResNet50 model pre-trained on ImageNet, with its top classification layers removed.
      2. **Customization**: Added a custom classification head to adapt ResNet50 to our specific task.
      3. **Training**: Initially trained with the base model frozen, followed by fine-tuning of some layers.
2. **Models Used**
   1. **Custom CNN**
      1. **Description**: A CNN model specifically designed for the classification task. The architecture consists of:

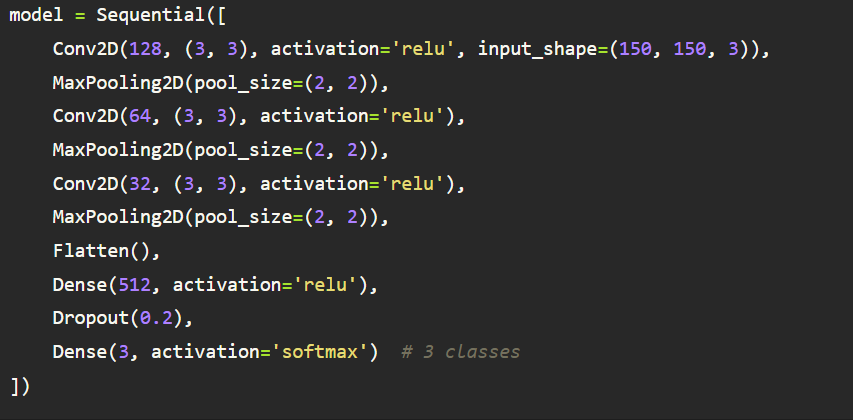
Convolutional layers for feature extraction

Max-pooling layers to reduce dimensionality

Flatten layer to convert 2D features to 1D

Dense layers for classification

* + 1. **Implementation**:

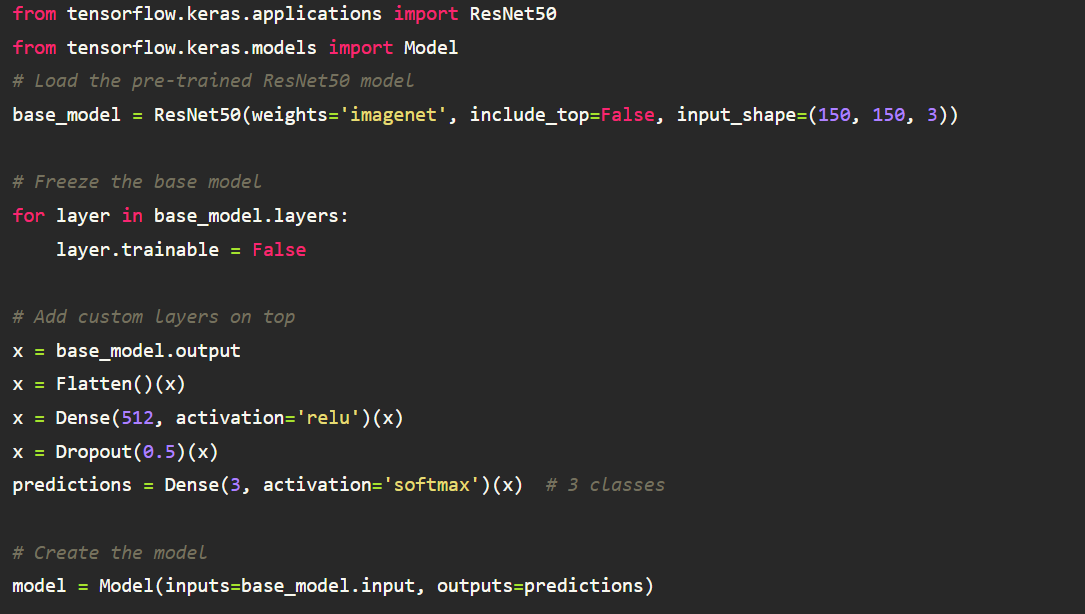


* 1. **Transfer Learning with ResNet50**
     1. **Description**: Utilized the ResNet50 model with pre-trained weights from ImageNet. The model includes:

Pre-trained convolutional layers

Custom classification head added for our task

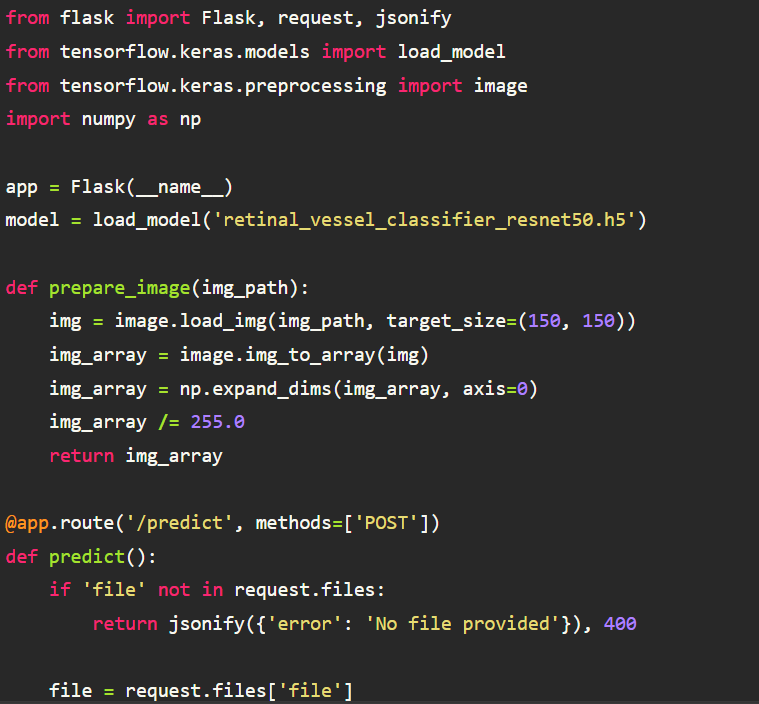
* + 1. **Implementation**:



1. **Training and Evaluation**
   1. **CNN Training**: The custom CNN was trained for 25 epochs with data augmentation applied.
   2. **Transfer Learning Training**: ResNet50 was initially trained with frozen layers and then fine-tuned by unfreezing some layers for additional epochs.
2. **Results and Comparison**
   1. **CNN Model**: Achieved a certain accuracy and loss which was evaluated using training and validation datasets.
   2. **ResNet50**: Leveraged pre-trained features, showing potential improvements in accuracy and generalization compared to the custom CNN.
3. **API Endpoints for Deployment**
   1. **Setup Flask**
      1. **Install Flask**:

!pip install Flask

* 1. **Create API Server**:



A screen shot of a computer program

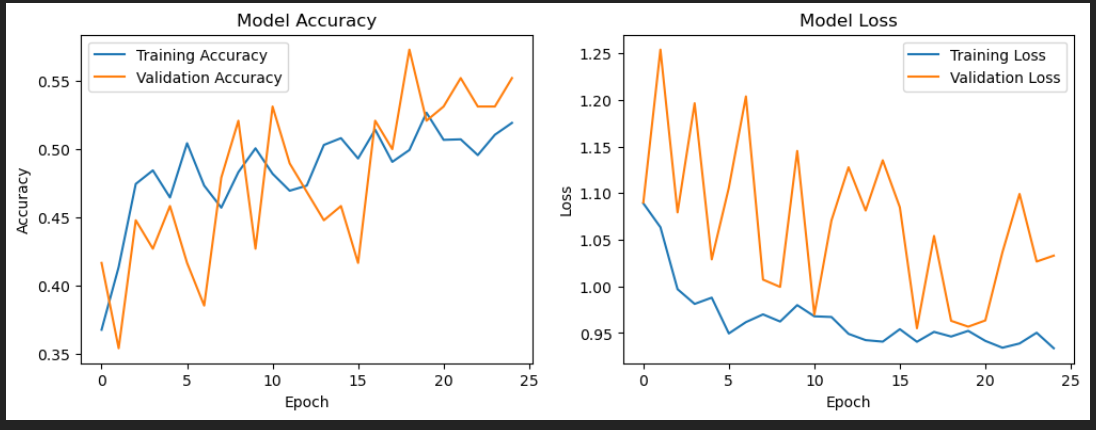
Description automatically generated

* 1. **Testing the API**: Use Postman or cURL to send POST requests with image files for predictions.
  2. **Train and validation accuracy of ResNet50 model**

**A graph of different colored lines

Description automatically generated with medium confidence**

* 1. **Train and validation accuracy of custom CNN model:**

****

**8. Conclusion:**

This project highlights the strengths of both a custom CNN and transfer learning in classifying retinal vessel images. While the custom CNN was effective in learning features directly from the dataset, the transfer learning approach using ResNet50 demonstrated superior performance by leveraging pre-trained features from a large-scale dataset. The combination of these two methods provided a comprehensive analysis of the classification task. Both models were successfully deployed through an API, making the system practical for real-world applications and enabling seamless interaction for further use cases.

**9. References:**

* 1. **Deep Learning in Medical Image Analysis:**

**9.1.1 Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., ... & van Ginneken, B. (2017). A survey on deep learning in medical image analysis. *Medical image analysis*, 42, 60-88.**

[***Link*: https://doi.org/10.1016/j.media.2017.07.005**](Link:%20https://doi.org/10.1016/j.media.2017.07.005)

* 1. **Transfer Learning in Medical Imaging:**

**9.2.1 Shin, H. C., Roth, H. R., Gao, M., Lu, L., Xu, Z., Nogues, I., ... & Summers, R. M. (2016). Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning. *IEEE transactions on medical imaging*, 35(5), 1285-1298.**

[***Link*: https://doi.org/10.1109/TMI.2016.2528162**](Link:%20https://doi.org/10.1109/TMI.2016.2528162)

* 1. **ResNet and its Applications:**

**9.3.1 He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).**

[***Link*: https://doi.org/10.1109/CVPR.2016.90**](Link:%20https://doi.org/10.1109/CVPR.2016.90)

* 1. **Data Augmentation Techniques:**

**9.4.1 Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1), 1-48.**

[***Link*: https://doi.org/10.1186/s40537-019-0197-0**](Link:%20https://doi.org/10.1186/s40537-019-0197-0)

* 1. **Flask API for Deploying ML Models:**
     1. ***Flask Documentation*: Comprehensive guide on creating APIs using Flask.**

[***Link*: https://flask.palletsprojects.com/**](Link:%20https://flask.palletsprojects.com/)

* 1. **Automated Detection of Retinal Diseases:**

**9.6.1 Gulshan, V., Peng, L., Coram, M., Stumpe, M. C., Wu, D., Narayanaswamy, A., ... & Webster, D. R. (2016). Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. *JAMA*, 316(22), 2402- 2410.**

[***Link*: https://doi.org/10.1001/jama.2016.17216**](Link:%20https://doi.org/10.1001/jama.2016.17216)