Deep Learning to Predict Neurodegenerative Diseases

By Nikhita Mantravadi

What are Neurodegenerative Diseases?

Disorders characterized by the progressive loss of nerve cell function, often resulting in debilitating symptoms and decline in health.

- Characteristics:
  - Progressive and incurable
  - Affect movement, memory, and cognitive abilities
- Impact:
  - Millions affected worldwide
  - Significant burden on healthcare and families
- Common Examples:
  - Alzheimer's Disease: Impacts memory and cognition
  - Parkinson's Disease: Affects movement and coordination
- Research Importance:
  - No cures available, making research critical for future breakthroughs

About Alzheimer’s and Parkinson’s

- Alzheimer’s Disease:
  - Type: Most common form of dementia.
  - Symptoms: Memory loss, confusion, difficulty with language and problem-solving.
  - Prevalence: Affects about 6% of people aged 65 and older.
  - Cause: Plaques and tangles in the brain, though exact cause remains unclear.
- Parkinson’s Disease:
  - Type: Progressive neurological disorder.
  - Symptoms: Tremors, stiffness, slow movement, and impaired balance.
  - Prevalence: Affects about 1% of people aged 60 and older.
  - Cause: Loss of dopamine-producing brain cells; exact triggers unknown.
- Shared Challenges:
  - Both diseases progressively worsen and significantly impact daily living.
  - No cure currently available, but medications and therapies can manage symptoms.
- Research Focus:
  - Developing treatments to slow disease progression.
  - Improving diagnostic techniques to detect diseases earlier.

The steps I took

- Collecting the data.
- Converting and rotating the images in the Parkinson’s data
- Categorizing each class
- Building the Model
- Classification and checking the Accuracy

About the Datasets:

Alzheimer's Disease:

- Subdivided into Mild Demented, Moderate Demented, Very Mild Demented, Non Demented
- Consisted of .jpg images

Parkinson's Disease:

- Subdivided into PD-30, Prodormal-30 and Control-30.
- Consisted of .nii images

The 3 approaches

1. Classifying into Alzheimer's(Mild Demeneted, Very Mild Demented and Moderate Demented), Parkinson's(PD-30 and Prodromal-30)and None(Alzheimer's- Non Demented and Parkinson's-Control-30)
2. Classifying into Alzheimer's (Mild Demented and Moderate Demented), Parkinson's (PD-30, Prodromal-30 and Control-30) and None(Alzheimer's- Non_Demented and Parkinson's-Control-30)
3. Additionally- Classification of just the 4 classes in Alzheimer's

The Models I used:

1. Custom CNN Model
2. ResNet50 Model

1. Custom CNN Model

- Architecture:
  - Built from scratch using the Sequential API in TensorFlow.
  - Consists of 3 Convolutional layers followed by Max Pooling, a Flatten layer, a Dense layer with 512 units, and a Dropout layer to mitigate overfitting.
  - The final output layer has 3 neurons for classification with softmax activation.
- Training:
  - Used the Adam optimizer with a learning rate of 0.0001 for precise adjustments during training.
  - Trained for 20 epochs with both training and validation datasets for performance monitoring.

2. ResNet50 Based Model

- Architecture:
  - Utilizes the ResNet50 architecture pre-trained on ImageNet as the base.
  - The base model's weights are frozen to leverage learned features.
  - Custom top layers include Global Average Pooling, Dense, and Dropout layers before the final classification layer.
- Training:
  - Employed a low learning rate with the Adam optimizer for fine-tuning.
  - Went through 20 epochs, balancing the learning from pre-trained patterns and the new dataset.

About the Accuracy:

CLASSIFICATION	CNN	RESNET50
3 classes[Alzheimer's(Very_Mild_Demented, Mild_Demented, Moderate_Demented) , Parkinson's(PD-30, Prodromal-30), None(Non_Demented, Control-30)]	70.05%	62.31%
3 classes[Alzheimer's(Mild_Demented, Moderate_Demented), Parkinson's(PD-30, Prodromal-30), None(Non_Demented, Control-30)]	67.78%	62.80%
4 classes [Alzheimer's- Very_Mild_Demented, Moderate_Demented, Mild_Demented, Non_Demented]	55.88%	50.57%

Takeaways:

- Tailoring models to specific tasks can lead to improved performance over generic solutions.
- The model could have performed better with better data.

Future work: Model refinement and incorporation of better additional data, could help attain better accuracy scores.