iNeuron

Architecture Document

Alzheimer Disease Classifier

Revision Number: 1.0

Last Date of Revision: [27-02-2025]

Author(s): Jatindra Paul

Document Version Control

Date Issued	Version	Description	Author
[27-02-2025]	1.0	Initial Draft	Jatindra Paul

Contents

---

1. Abstract

Alzheimer's disease is a progressive neurodegenerative disorder that affects millions worldwide. Early detection using **deep learning** techniques can assist in timely interventions and better disease management. This project aims to build an **Alzheimer Disease Classifier** using a **VGG16-based** deep learning model. The system processes **MRI scan images** to classify whether a patient has Alzheimer's or not, leveraging pre-trained models and transfer learning to improve classification accuracy.

---

2. Introduction

2.1 Purpose

The purpose of this **Architecture Document** is to provide a **comprehensive overview** of the system design, data flow, and technology stack for the Alzheimer Disease Classifier. It outlines the **data processing pipeline, model architecture, inference mechanism, and deployment strategy** to ensure an efficient and scalable solution.

2.2 Scope

- **Application Type:** Medical Image Classification
- **Model Used:** VGG16-based CNN for **binary classification** (Alzheimer's Positive/Negative)
- **Data Source:** MRI scans of patients with and without Alzheimer's
- **Deployment:** Cloud-based inference system with a REST API
- **End Users:** Medical researchers, radiologists, and healthcare providers

2.3 Constraints

- The model should work with **limited labeled data** due to the scarcity of MRI scans.
- The system should comply with **HIPAA** or **GDPR** regulations for handling sensitive medical data.
- **Inference latency** should be minimal for real-time classification.

2.4 Risks

Risk	**Mitigation Strategy**
Model overfitting due to small dataset	Use data augmentation & dropout regularization
Poor generalization on new images	Implement cross-validation and fine-tune hyperparameters
Privacy concerns with patient data	Ensure encryption & secure cloud deployment

2.5 Out of Scope

- **Live clinical diagnosis:** The classifier is intended as an **assistive tool**, not a replacement for professional diagnosis.

- **Multi-class classification:** This version focuses only on **binary classification** (Alzheimer's vs. Non-Alzheimer's).

---

3. Technical Specifications

3.1 Dataset Description

- **Data Type:** MRI scans (DICOM, PNG, JPEG formats)
- **Classes:**
    - **Alzheimer's Positive** (patients diagnosed with AD)
    - **Alzheimer's Negative** (healthy individuals)
- **Dataset Size:** ~10,000 MRI images
- **Data Augmentation:** Rotation, flipping, brightness adjustment

3.2 Data Pre-processing

- **Image Resizing:** Convert images to **224x224 pixels** for compatibility with VGG16.
- **Normalization:** Scale pixel values between **0 and 1** for uniform input.
- **Augmentation:** Synthetic image transformations to increase dataset diversity.

3.3 Model Training

- **Base Model:** VGG16 (pre-trained on ImageNet).
- **Fine-tuning:** Freeze early layers, retrain last **fully connected layers**.
- **Loss Function:** Binary Cross-Entropy.
- **Optimizer:** Adam.
- **Metrics:** Accuracy, Precision, Recall, AUC-ROC.
- **Training Split:**
    - **80%** Training
    - **10%** Validation
    - **10%** Testing

3.4 Model Evaluation

- **Validation Metrics:** Confusion matrix, AUC-ROC curve, Precision-Recall curve.
- **Baseline Comparison:** Compare VGG16-based model with other CNN architectures.

3.5 Model Deployment

- **API Service:** Deploy model as a **Flask based web service**.

- **Containerization:** Use **Docker** for easy deployment.
- **Cloud Hosting:** AWS, GCP, or Azure for scalability.
- **Security:** Implement **HTTPS encryption** and **authentication mechanisms**.

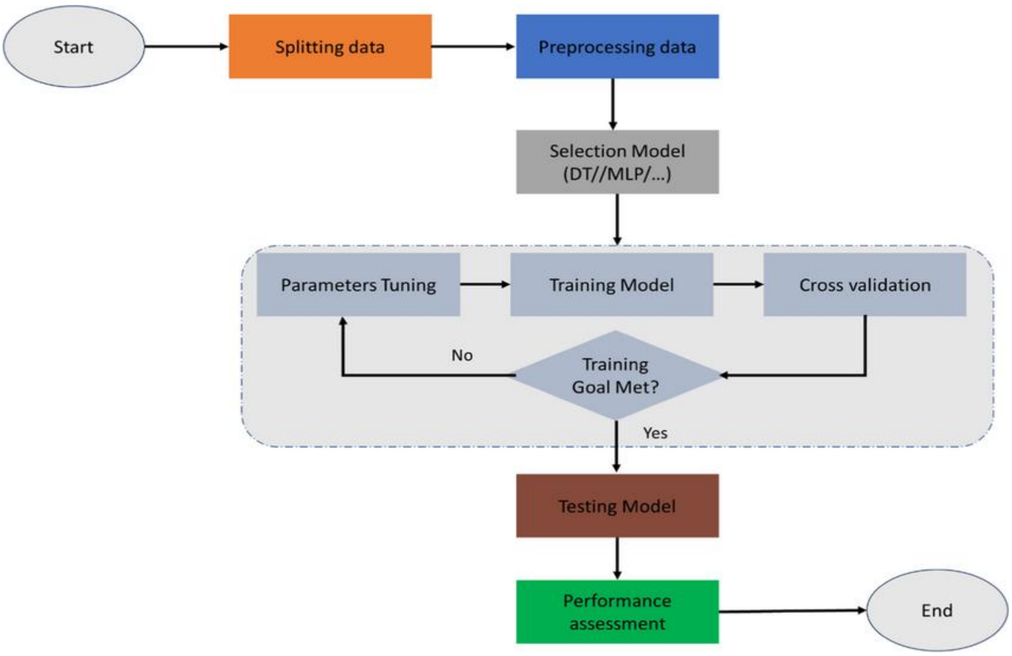
---

4. Technology Stack

Component	**Technology**
Model Development	TensorFlow
API Development	Flask
Database	AWS S3
Deployment	AWS
Version Control	GitHub

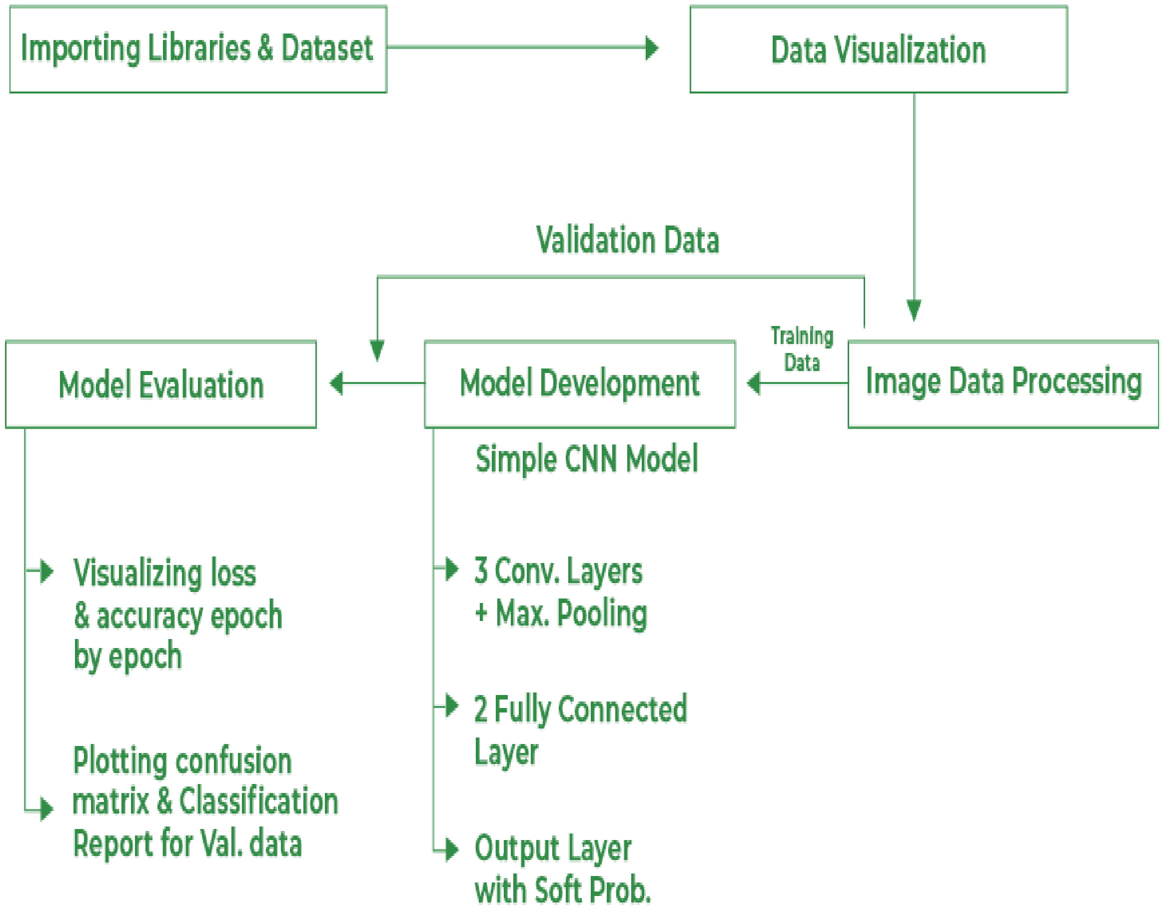
5. **Proposed Solution**

1. **Data Acquisition** → Collect and preprocess MRI scans.
2. **Model Training** → Train a **VGG16-based classifier**.
3. **Inference API** → Deploy a **Flask service** for real-time classification.
4. **Frontend/UI** → Provide an **interface for users to upload MRI images and view results**.



6. Model Training & Validation Workflow

1. **Data Preprocessing** → Normalize & Augment data.
2. **Model Training** → Fine-tune VGG16 with binary labels.
3. **Hyperparameter Tuning** → Optimize batch size, learning rate, dropout.
4. **Model Validation** → Test on unseen data & evaluate performance.



7. User I/O Workflow

1. **User Uploads an MRI Scan** → System preprocesses the image.
2. **Model Processes the Image** → Generates a classification (Alzheimer's / Non-Alzheimer's).
3. **User Views the Results** → Displays classification & confidence score.

8. Exceptional Scenarios

Step	Exception	Mitigation
Image upload	Unsupported file format	Validate file type before upload
Model inference	Model fails to load	Implement failover mechanism

Step	Exception	Mitigation
API response delay	High traffic	Scale using load balancing

9. Key Performance Indicators (KPIs)

- **Model Accuracy: 90%+** on test data.
- **Inference Time:** Predictions should be **< 500ms**.
- **Security Compliance:** Enforce **HIPAA/GDPR** standards.
- **Scalability:** Handle **1000+ requests per hour** without degradation.

Conclusion

This **Architecture Document** provides an in-depth view of the **Alzheimer Disease Classifier** system, covering **data processing, model training, deployment, and security considerations**. The system aims to provide **efficient, scalable, and secure** classification of MRI scans to assist in **early detection of Alzheimer's disease**.