SIDDAGANGA INSTITUTE OF TECHNOLOGY, TUMAKURU-572103

(An Autonomous Institute under Visvesvaraya Technological University, Belagavi)

Project Report on

"Alzheimer's Disease Detection using Deep Learning"

submitted in partial fulfillment of the requirement for the award of the degree of

BACHELOR OF ENGINEERING

in

ELECTRONICS & COMMUNICATION ENGINEERING

Submitted by

Aravind V A	(1SI20EC011)
Harsha S N	(1SI20EC033)
J Vinay	(1SI20EC037)
Nikhil K G	(1SI20EC059)

under the guidance of

Dr. Shilpashree P S

Assistant Professor

Department of E&CE

SIT, Tumakuru-03

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

2023-24

SIDDAGANGA INSTITUTE OF TECHNOLOGY, TUMAKURU-572103

(An Autonomous Institute under Visvesvaraya Technological University, Belagavi)

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

CERTIFICATE

Certified that the project work entitled "ALZHEIMER'S DISEASE DETECTION USING DEEP LEARNING" is a bonafide work carried out by Aravind V A (1SI20EC011), Harsha S N (1SI20EC033), J Vinay (1SI20EC037), Nikhil K G (1SI20EC059) in partial fulfillment for the award of degree of Bachelor of Engineering in Electronics & Communication Engineering from Siddaganga Institute of Technology, an autonomous institute under Visvesvaraya Technological University, Belagavi during the academic year 2023-24. It is certified that all corrections/suggestions indicated for internal assessment have been incorporated in the report deposited in the department library. The Project report has been approved as it satisfies the academic requirements in respect of project work prescribed for the Bachelor of Engineering degree.

Dr. Shilpashree P S	Head of the Department	Principal
Assistant Professor	Dept. of E&CE	SIT, Tumakuru-03
Dept. of E&CE	SIT, Tumakuru-03	
SIT, Tumakuru-03		

External viva:

Name of Examiners	**Signature with date**
1.	
2.	

ACKNOWLEDGEMENT

We offer pranams at the lotus feet of **His** Holiness, **Dr. Sree Sree Sivakumara Swamigalu**, Founder President and **His** Holiness, **Sree Sree Siddalinga Swamigalu**, President, Sree Siddaganga Education Society, Sree Siddaganga Math for bestowing upon their blessings.

We deem it as a privilege to thank **Dr. M N Channabasappa**, Director, SIT, Tumakuru, **Dr. Shivakumaraiah**, CEO, SIT, Tumakuru, and **Dr. S V Dinesh**, Principal, SIT, Tumakuru for fostering an excellent academic environment in this institution, which made this endeavor fruitful.

We would like to express our sincere gratitude to **Dr. K V Suresh**, Professor and Head, Department of E&CE, SIT, Tumakuru for his encouragement and valuable suggestions.

We thank our guide **Dr. Shilpashree P S**, Assistant Professor, Department of Electronics & Communication Engineering, SIT, Tumakuru, **Dr. K V Suresh**, Professor and Head, Department of Electronics & Communication Engineering, SIT, Tumakuru, and **Mr. Dayanand Kumar**, Project Lead, Samsung Electro Mechanics Software India Ltd, World Trade Center, Bangalore for the valuable guidance, advice and encouragement.

Aravind V A	(1SI20EC011)
Harsha S N	(1SI20EC033)
J Vinay	(1SI20EC037)
Nikhil K G	(1SI20EC059)

Course Outcomes

After successful completion of major project, graduates will be able to

CO1: To identify a problem through literature survey and knowledge of contemporary engineering technology.

CO2: To consolidate the literature search to identify issues/gaps and formulate the engineering problem

CO3: To prepare project schedule for the identified design methodology and engage in budget analysis, and share responsibility for every member in the team

CO4: To provide sustainable engineering solution considering health, safety, legal, cultural issues and also demonstrate concern for environment

CO5: To identify and apply the mathematical concepts, science concepts, engineering and management concepts necessary to implement the identified engineering problem

CO6: To select the engineering tools/components required to implement the proposed solution for the identified engineering problem

CO7: To analyze, design, and implement optimal design solution, interpret results of experiments and draw valid conclusion

CO8: To demonstrate effective written communication through the project report, the one-page poster presentation, and preparation of the video about the project and the four page IEEE/Springer/ paper format of the work

CO9: To engage in effective oral communication through power point presentation and demonstration of the project work.

CO10:To demonstrate compliance to the prescribed standards/ safety norms and abide by the norms of professional ethics.

CO11: To perform in the team, contribute to the team and mentor/lead the team

PSOs:

PSO1:The ability to analyze and design systems in the areas related to microelectronics, communication, signal processing and embedded systems for solving real world problems **(Professional Skills)**.

PSO2: The ability to identify problems in the areas of communication and embedded systems and provide efficient solutions using modern tools/algorithms individually or working in a team **(Problem-Solving Skills)**.

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO-1												3		3
CO-2	3												3	
CO-3											3			3
CO-4						3	3							3
CO-5	3	3											3	
CO-6					3									3
CO-7			3	3									3	
CO-8										3				3
CO-9										3				3
CO-10								3						3
CO-11									3					3
Average	3	3	3	3	3	3	3	3	3	3	3	3	3	3

Attainment level: - 1: Slight (low) 2: Moderate (medium) 3: Substantial (high)

POs: PO1: Engineering Knowledge, PO2: Problem analysis, PO3: Design/Development of solutions, PO4: Conduct investigations of complex problems, PO5: Modern tool usage, PO6: Engineer and society, PO7: Environment and sustainability, PO8: Ethics, PO9: Individual and team work, PO10: Communication, PO11: Project management and finance, PO12: Lifelong learning

Abstract

Alzheimer's disease (AD) is a neurodegenerative disorder that is commonly seen in old age people. It involves the brain functions that control thought, memory and language. It can seriously affect a person's ability to carry out daily activities. The accurate diagnosis of AD plays an important role in treatment, especially at the early stages of the disease, because risk awareness allows the patients to undergo preventive measures even before the occurrence of irreversible brain damage. Detecting AD at an early stage can help in slowing down the progression of the disease by taking appropriate medication. Traditionally, AD diagnosis is done using cognitive testing, but these tests are ineffective at early stages of AD.

Advancements in technology has allowed for studying the human body and with the help of artificial intelligence the detection of certain diseases can be done. Machine learning (ML) and Deep learning (DL) models are used for the detection of diseases, the data collected from patients is used for training the models so that an efficient model is developed and used for detecting diseases.

This project uses Magnetic Resonance Imaging (MRI) of brain as dataset. ML and DL models are trained and tested against different datasets. The performance of the model against different datasets are compared, the evaluation metrics used for the comparison are accuracy, precision, F1-score and recall. Experiments have yielded promising results to improve the accuracy of AD detection which help many doctors in providing proper treatment for AD based on the severity of AD.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The advancements in artificial intelligence provide new ways for researchers to study and diagnose different neurological diseases relating human brain, making use of advanced technologies such as Magnetic Resonance Imaging, the various conditions of brain are studied. Datasets of brain MRI are used so that they can be trained and tested against ML and DL models which contribute to accurate analysis for the diagnosis of the disease. AD is a neurodegenerative disorder which destroys the brain cells. Alzheimer can not only be found with cognitive ability testing, but also through MRI, by looking at the ventricles of the brain and cortical atrophy one can tell the difference between healthy and Alzheimer's affected brain. Doctors find the patient with Alzheimer's have a brain that has enlarged ventricles (that lies in the center of the brain) as well as thinner cortical grey area of the brain [1]. Figure 1.1 shows the affects that are caused to brain when affected by AD, the ventricles enlarge, the central cortex and hippocampus regions of brain reduce in size and Figure 1.2 shows the two MRI scans of brain out of which one is of a healthy brain, and another is of brain with AD. AD predictions made using data

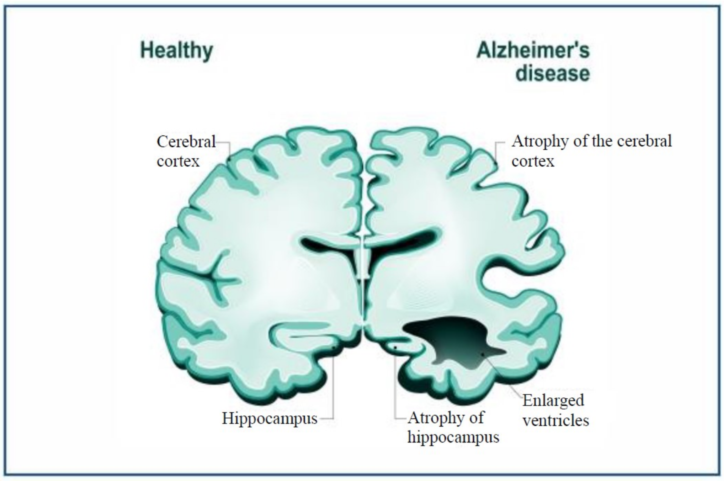


Figure 1.1: The comparison of a healthy brain (left part) and brain with Alzheimer's disease (right part) (Courtesy:Amini M et al [3])

from the Open Access Series of Imaging Studies (OASIS) [2], with performance evaluated by F1-score, Precision, Recall, and Accuracy. Clinicians can diagnose these illnesses using this classification. When these ML and DL algorithms are used for early diagnosis, the annual death rates from Alzheimer's disease can be significantly reduced.

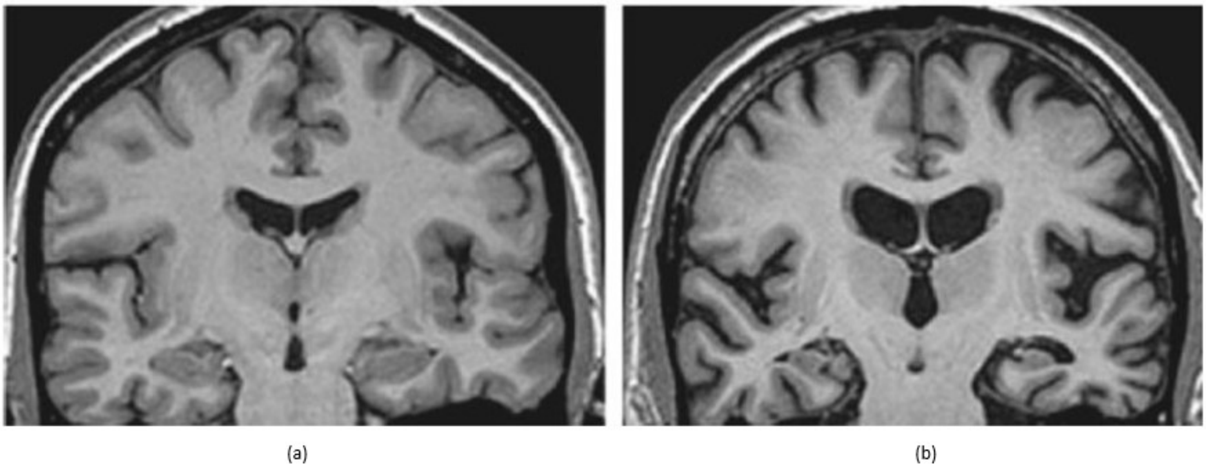


Figure 1.2: The MRI scan of a healthy brain (a) and brain with Alzheimer's disease (b) (Courtesy: Rao et al [5])

Early detection of AD is a tedious and costly process since we must collect a lot of data and use sophisticated tools for prediction and have an experienced doctor involved. Automated systems are more accurate than human assessment and can be used in medical decision support systems because they are not subject to human errors. Based on previous research on AD [3], researchers have applied images (MRI scans), biomarkers (chemicals, blood flow), and numerical data extracted from the MRI scans to study this Disease. As such, they were able to determine whether a person was demented or not. In addition to shortening diagnosis time, more human interaction will be reduced by automating Alzheimer's diagnosis. The severity of AD can be classified on the basis of degeneracy of brain, Figure 1.3 gives the different classes into which the AD severity is classified, they are given as:

Non-Demented: MRI indicates normal brain structure without significant atrophy or abnormalities. Clear and well-defined brain regions without evident signs of neurodegeneration.

Very Mild Demented: Subtle changes may be observed, such as mild atrophy in specific brain regions, particularly those associated with memory and cognitive function. Early signs of ventricular enlargement might be present.

Mild Demented: Increasing atrophy becomes more evident, especially in regions crucial for memory and cognition. Enlargement of ventricles and shrinkage of the hippocampus may be notable. These changes contribute to cognitive decline.

Moderate Demented: Increased atrophy and structural changes throughout the brain, affecting multiple regions. Significant enlargement of ventricles, reduced hippocampal volume, and widespread cortical atrophy. These changes correlate with severe cognitive impairment.

In addition, automation reduces overall costs and provides more accurate results. For example, we can predict whether a patient is demented by analyzing MRI scans and applying prediction techniques. If a person has early-stage Alzheimer's Disease, they are considered demented. By doing so, we can achieve better accuracy. Researchers across the world test different datasets and check the performances of different ML and DL models that are used. Using systems that classify the abnormal case from normal case using MRI scans will contribute immensely to the initial diagnosis of AD.

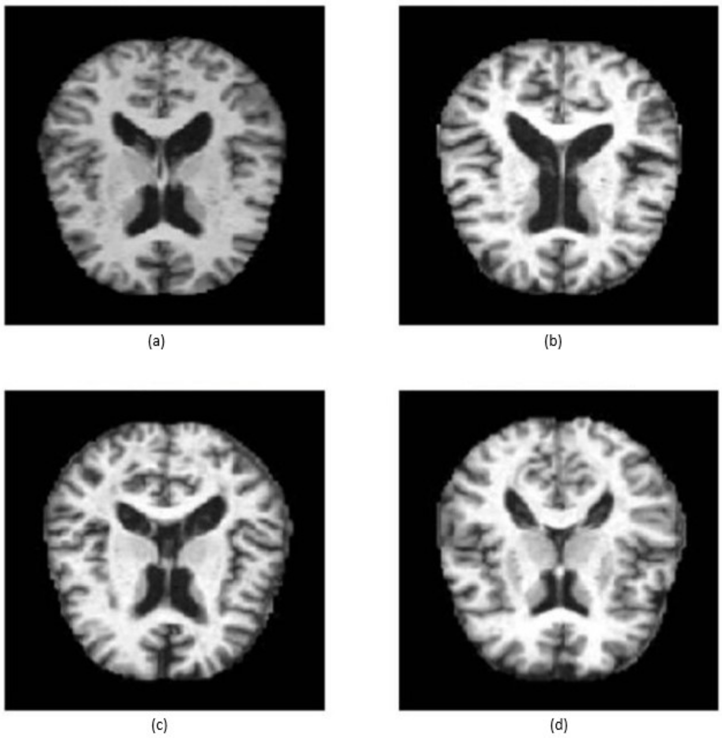


Figure 1.3: Severity of AD shown in different classes, (a) Mild demented, (b) Moderate demented, (c) Very mild demented, (d) Non demented (Courtesy: Rao et al [5])

1.1 Motivation

The motivation for taking up this project is the need for accurate AD detection methods, as the importance of the disease rises due to the aging population thus emphasizing the urgency for AD detection methods. The current methods for diagnosing AD are often inaccurate and time-consuming. This can lead to delays in treatment, which can have a significant impact on the quality of life for patients and their families. Machine learning and deep learning are two powerful tools that can be used to develop accurate and scalable detection methods for AD. These techniques can be used to analyze MRI scans of the brain to identify subtle changes that are associated with the disease. These methods can help to ensure that patients are diagnosed early and receive the appropriate treatment, which can help to slow the progression of the disease and improve quality of life.

1.2 Objectives of the project

The objectives of the project are:

• To build ML model using HOG and SVM for classify AD from MRI scans.

• To develop a DL algorithm for the classification of Alzheimer's disease.

• To compare the performance of the ML models and different variants of DL models against different datasets.

1.3 Organisation of the report

This report is divided into seven chapters. Chapter 1 contains the introduction, motivation and objectives of the project and organisation of the report. The literature survey of the project, which includes a brief description of different techniques available for AD detection is presented in chapter 2. Chapter 3 deals with the system overview that is planned for the course of this project. The software implementation and the tools used for the project is presented in chapter 4. Chapter 5 deals with the results obtained. The report concludes with the conclusion in chapter 6.

Chapter 2

Literature Survey

Early AD detection can be done using ML and DL models. Many scholars and researchers are working constantly on improving the efficiency in diagnosing AD with high accuracy. This has resulted in contribution in developing of many neural network architectures that can be useful in diagnosing AD. This chapter discusses about the disease and the different methods that are used and documented by various scholars and researchers around the world contributing to the diagnosis of AD.

2.1 Alzheimer's Disease diagnosis

Recent Alzheimer's [1] research has revealed more about the disease's causes and earlier detection. Scientists are moving beyond the amyloid beta [2] theory and identified numerous genes influencing risk. New blood tests hold promise for earlier diagnosis, even before symptoms appear. Advanced imaging techniques can pinpoint protein buildup in the brain, this can be seen in Figure 2.1, where different scans on brain were done to diagnose the AD by the help of proteins. These breakthroughs pave the way for earlier interventions, potentially including preventative measures and combination therapies. With continued progress, very early detection and personalized treatment for Alzheimer's could become a reality.

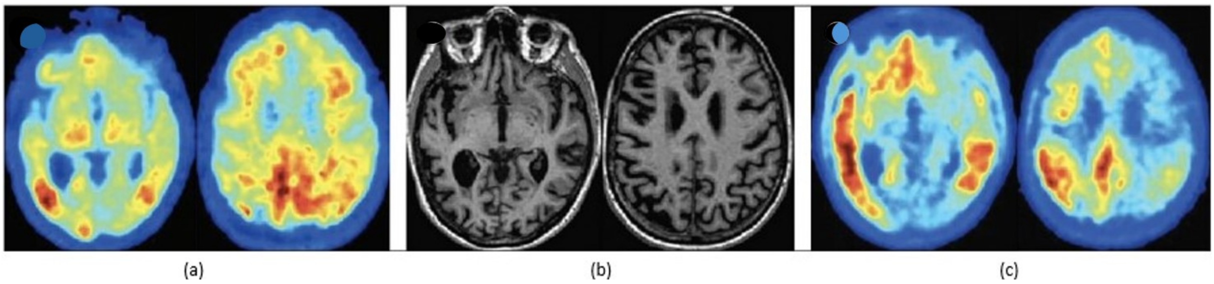


Figure 2.1: (a) Tomography scan of brain showing protein deposition (orange colour). (b) MRI images showing generalised cortical atrophy, left to right. (c) Tomography image using AV1451 tracer, showing deposition of tau proteins (orange colour). (Courtesy: Scheltens P et al [1])

2.2 AD diagnosis using ML

A machine learning study explored diagnosing AD severity with functional MRI (fMRI) scans. Brain scans from 675 patients are processed and trained models to categorize AD stages (low, very mild, mild and moderate) based on extracted features. Figure 2.2 represents the techniques used by authors to get different features from brain MRI. This approach showed good sensitivity (finding true AD cases) and precision (avoiding false positives) across severity levels [3].

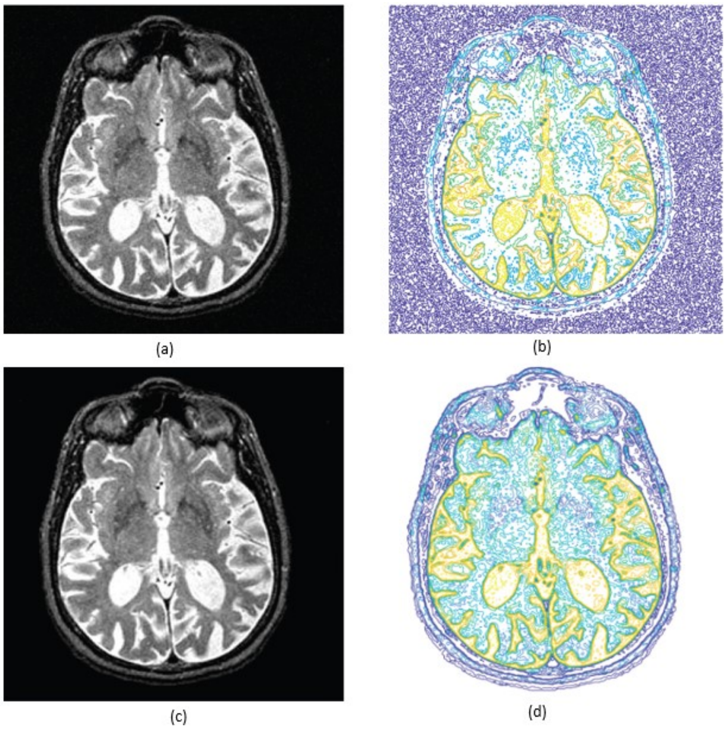


Figure 2.2: (a) is the Brain MRI. (b) is the Contour form of (a). (c) is the image obtained after noise reduction of (a). (d) is the contour of (c). (Courtesy: Amini M et al [3])

Various machine learning techniques, such as Decision Tree, Random Forest (RF), Support Vector Machine (SVM), Gradient Boosting, and Voting classifiers are used fro AD classification. Using data from the Open Access Series of Imaging Studies, the proposed classification scheme demonstrates promising results, achieving a high validation average accuracy of 83% on test data for AD. These machine learning algorithms provide clinicians with valuable tools for early diagnosis, contributing to the reduction of annual mortality rates associated with Alzheimer's disease. The study's outcomes surpass existing works, emphasizing the potential of machine learning in advancing early prediction and intervention strategies for AD [4]. ML algorithms are tested against different datasets,

Table 2.1: Accuracy of various ML models proposed by authors in their study using different datasets

Sl No.	Authors	Dataset used	Model used	Accuracy
1	[5]	BRFSS	SVM and MLP	0.9747 (Binary classification)
2	[7]	ADNI	SVM with image processing	0.9330 (Binary classification)
3	[8]	kaggle	SVM	0.9 (Binary classification)
4	[9]	kaggle	HOG with SVM	0.5 (Multiclass classification)
5	[10]	OASIS CSV dataset	Random Forest Classifier	0.9213 (Binary classification)
6	[11]	kaggle	SVM	0.74 (Binary classification)
7	[12]	OASIS	Logistic Regression	0.8433 (Binary classification)
8	[13]	ADNI	Generalised Linear Model	0.8824 (Multiclass classification)
9	[14]	OASIS	SVM	0.9677 (Binary classification)
10	[15]	ADNI	Deep Deterministic Policy Gradient (DDPG)(Reinforcement Learning Approach)	0.9834 (Binary classification)

Researchers are focusing on creating an automated algorithm using Machine Learning to accurately identify Alzheimer's in its early stages. White and grey matter are analyzed using 3D magnetic resonance imaging scans and machine learning algorithms to extract pertinent features, which are then used to predict and classify using Multi-Layer Perceptron (MLP) and SVM techniques [4]. Metrics including Precision, Recall, Accuracy, and F1-Score are used to assess the system's effectiveness; this technique shows promise for better Alzheimer's detection. The accuracy metrics of several machine learning models applied to diverse datasets by different authors are shown in Table 2.1.

Automatic Pipeline Methods and ML methods, incorporating biomarker methods, fusion, and registration for multimodality, have emerged for processing medical scans. These automated systems and machine learning techniques show success rates exceeding 95% in accurately identifying AD and its stages for binary class classification. Challenges remain in multi-class classification, particularly in distinguishing between AD and mild cognitive impairment (MCI), as well as sub-stages of MCI. The research underscores the importance of employing multi-modality approaches for effective validation in detecting AD and its stages [6].

2.3 AD diagnosis using DL

When compared to machine learning techniques, the Deep Learning model is more effective in classifying AD since it can process huge datasets faster and produce results sooner. Since risk awareness enables patients to take preventive action even before irreversible brain damage occurs, an accurate diagnosis of AD is crucial to patient therapy, particularly in the early stages of the disease. While computers have been employed in many recent research to diagnose AD, congenital observations remain the primary source of limitation for most machine detection methods [16]. Early diagnosis of AD is possible, however prediction is not possible because prediction is only useful prior to the disease's onset. DL is now a widely used method for AD early diagnosis.

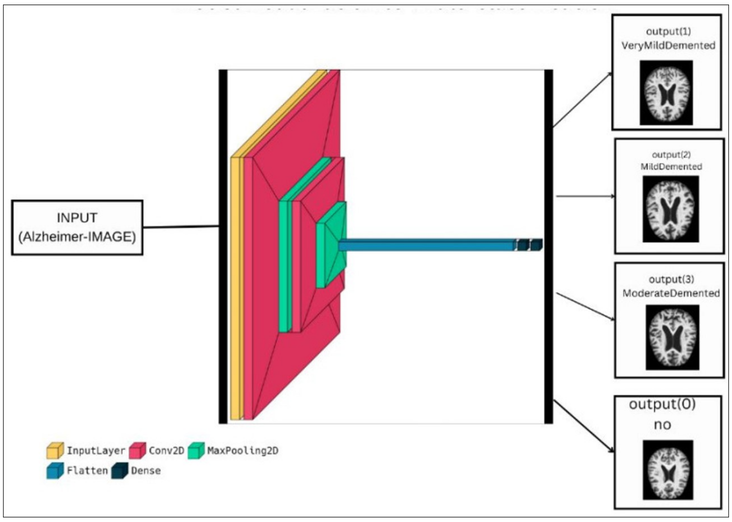


Figure 2.3: Deep learning model as proposed by El-Latif et al [17]. (Courtesy: El-Latif et al [17])

AD is a neurodegenerative disorder with cognitive impairment and abnormal brain protein buildup. A streamlined deep learning model (Figure 2.3) for accurate AD detection from magnetic resonance imaging. The model achieves high accuracy without the need for complex layers, combining feature extraction and classification in one stage. With only seven layers, the model is efficient and outperforms previous models on a Kaggle dataset, demonstrating its effectiveness for AD classification [17].

Use of 3D MRI brain images is done for enhancing the Alzheimer's disease diagnosis [18]. Figure 2.4 gives the representation of images in Alzheimer's Disease Neuroimaging Initiative (ADNI) dataset. The document presents the Histogram of Oriented Gradients from Three Orthogonal Planes (HOG-TOP), a novel feature descriptor. In order to reduce dimensionality because of high-dimensional features, probabilistic principal component analysis or PPCA, is used. The classification of Alzheimer's, MCI, and normal cases is done by binary classification using a random forest classifier. In all classifications, experimental results demonstrate the superior sensitivity of HOG-TOP. In tackling the difficulties associated with Alzheimer's detection, the study highlights the significance of texture-based features. Especially for large medical image datasets, future work will

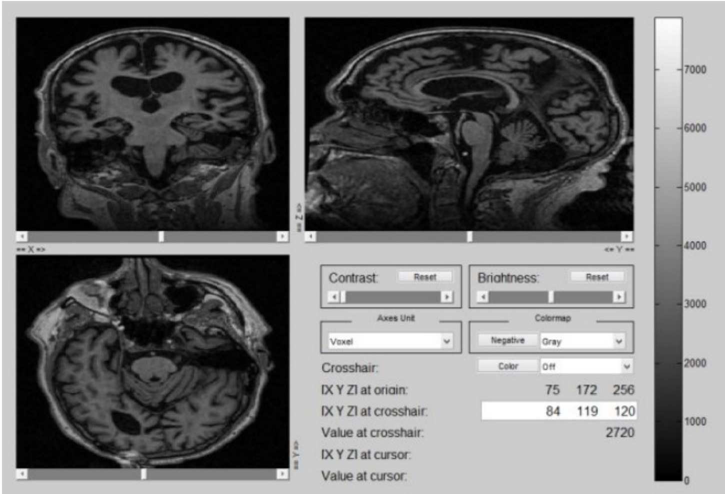


Figure 2.4: A Sample of Alzheimer's Brain MRI (ADNI database) from three orthogonal planes (Courtesy: El-Latif et al [18])

explore multimodal data, including 3D Tomography brain images. An end-to-end framework utilizing a deep learning approach, specifically convolutional neural networks, for the early detection and medical image classification of various AD stages [19]. Four stages of the AD spectrum are classified, and binary classifications are performed between each two-pair class. The method employs simple CNN architectures on 2D and 3D brain scans from the ADNI dataset, demonstrating suitable structures with promising accuracies of 93.61% and 95.17% for 2D and 3D multi-class AD stage classifications.

2.4 Summary of Literature survey

The literature survey is an essential component to establish a comprehensive understanding of the existing knowledge and methodologies related to the classification of Alzheimer's disease from MRI scans using both ML and DL approaches. Different ML models used by authors are tabulated in Table 2.1 which gives information about the model and dataset used and the performance of that model.

Furthermore, the study explores the application of DL algorithms for Alzheimer's disease classification. Convolutional Neural Networks have gained prominence due to their ability to automatically learn hierarchical representations from raw image data. Transfer learning, a technique where pre-trained models are fine-tuned on specific datasets, has also shown promise in enhancing the performance of DL models for AD classification.

The evaluation metrics employed in the literatures include Accuracy, Precision, Recall, and F1-score, providing a comprehensive understanding of model performance. Comparative analyses between ML and DL models reveal fine-tuned insights into the strengths and limitations ML and DL approach.

The models that are having complex layers and take more time to give results are discussed. This project aims to reduce the complexity and give results with more accuracy in less amount of time.

Chapter 3

System Overview

The early detection of AD is done by the use of brain MRI as input data. The ML model and DL model is tested for four class classification, where classification of the data is into the following four classes: Non Demented, Very Mild Demented, Mild demented, Moderate Demented.

3.1 Block Diagram description

The system overview is given in Figure 3.1, the data from dataset is the brain MRI with various cases of AD at different stages. The brain MRI is given as input to the ML and DL models that are built. The output of the models are the classification of test data. The performance of the models and its variants are compared to obtain the highest performing variant of the model for the given dataset.

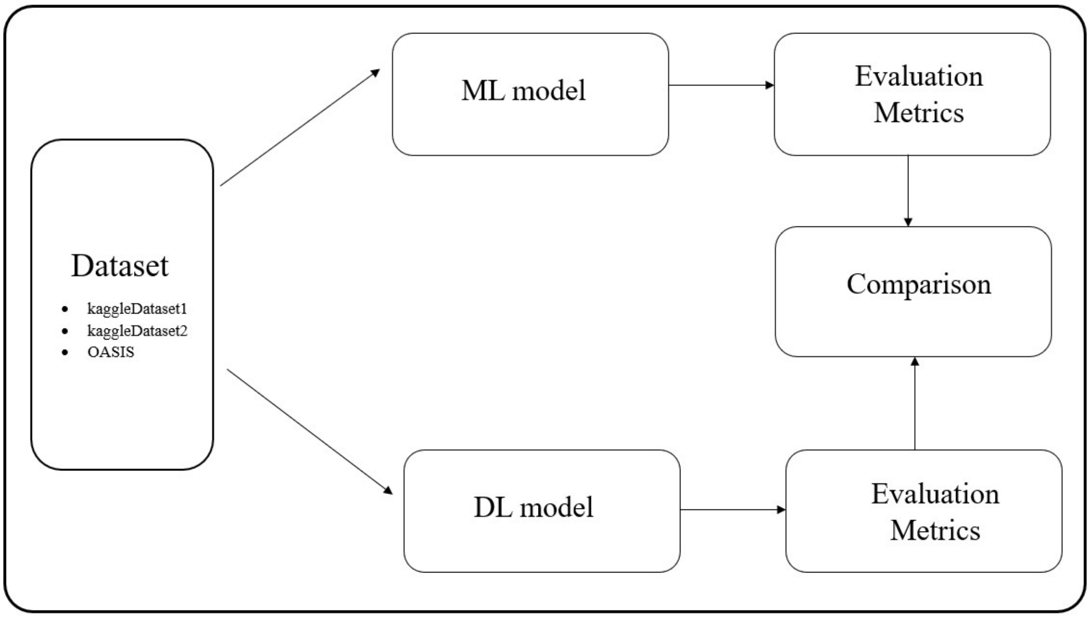


Figure 3.1: System Overview

- **Dataset**

The dataset consists of brain MRI which are in Joint Photographic Experts Group (JPEG) format. The datasets consists of brain MRI that are classified into four classes depending on the severity of the disease. Here dataset is processed so that it can be directly used for ML applications with minimal data preprocessing. Table 3.1 gives the number of classes in each dataset and the number of images that are present in each class. The kaggleDataset1 consists of images that have dimensions 176x208 pixels, the kaggleDataset2 consists of images with dimensions 128x128 pixels and the OASIS dataset consists of images with dimensions 496x248 pixels.

Table 3.1: Classification of datasets and the number of images in each class

Dataset	Class	Number of Images
kaggleDataset1	Non-Demented	2560
	Very mild Demented	1792
	Mild Demented	717
	Moderate Demented	52
kaggleDataset2	Non-Demented	3200
	Very mild Demented	2240
	Mild Demented	896
	Moderate Demented	64
OASIS dataset	Non-Demented	67222
	Very mild Demented	13725
	Mild Demented	5002
	Moderate Demented	488

- **Machine Learning**

Machine learning models use the kaggleDataset1 and kaggleDataset2 datasets for training and it is tested for performance evaluation. Two machine learning models are used, the first ML model uses HOG for feature selection and then uses SVM classifier for classification. The second model use K-Nearest Neighbor (KNN) and RF for classification. ML model implementation is discussed in chapter 5.

- **Deep Learning**

Deep learning model uses OASIS dataset for training and it is tested for performance evaluation. The DL model is trained and tested with two image classification models namely Densenet169 and MobilenetV2. Both the variants are tested against different learning rates. Adam and Adamax optimizers are used, the performance of model and its variants are trained and tested with the dataset. The DL model that is developed is discussed in chapter 5.

- **Evaluation metrics**

To compare the performance of individual models against same data set, evaluation metrics is used. The evaluation of performance of the model is done by comparing the Accuracy, Precision, Recall and F1-Score.

- **Accuracy** : The proportion of correctly classified instances out of the total instances. It provides an overall assessment of model correctness.
- **Precision** : It is the ratio of correctly predicted positive observations to the total predicted positives.
- **Recall** : It is the ratio of correctly predicted positive observations to the all observations in actual class.
- **F1-Score** : The harmonic mean of precision and recall. It provides a balance between precision and recall, especially when the classes are imbalanced.

The evaluation metrics can be understood by knowing the Confusion Matrix and its usage to calculate the accuracy, precision, F1-score and recall.

Confusion Matrix

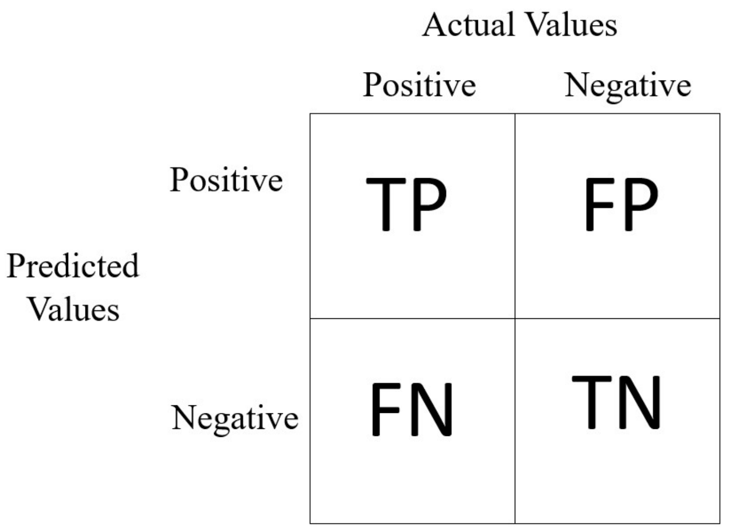


Figure 3.2: Confusion Matrix

Confusion matrix is a performance measurement for classification problem with two or more classes. Figure 3.2 shows the confusion matrix, it has 4 different combinations of predicted and actual values. They are,

- **True Positive** : Predicted value is positive and the actual value is true given by TP
- **True Negative** : Predicted value is negative and the actual value is true given by TN
- **False Positive** : Predicted value is positive and actual value is false given by FP
- **False Negative** : Predicted value is negative and actual value is false given by FN

Equations 3.1 to 3.4 use TP, TN, FP and FN to calculate the Accuracy, Precision, Recall and F1-Score.

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{3.1}$$

$$Precision = \frac{TP}{TP + FP} \tag{3.2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3.3}$$

$$F1 - Score = \frac{2 * Recall * Precision}{Recall + Precision} \tag{3.4}$$

Evaluation metrics in both ML and DL models serve as crucial benchmarks for assessing performance, guiding model selection, and fine-tuning. Metrics such as accuracy, precision, recall, and F1-score offer comprehensive insights into bias, generalization, and the ability to handle imbalanced datasets.

Chapter 4

Software Implementation

This chapter discusses about the ML and DL models that are used in this project. The models are tested by varying various parameters to get high accuracy, the results of each model and its variant that is tested against different datasets is given is chapter 5.

4.1 Machine Learning models

Machine learning models that are used in this project uses kaggleDataset1 and kaggleDataset2 for training and testing. Three models are tested against the datasets and the performance evaluation is done for each of the dataset, the model with high accuracy in classification of AD for the given datasets is known.

4.1.1 ML model using HOG and SVM

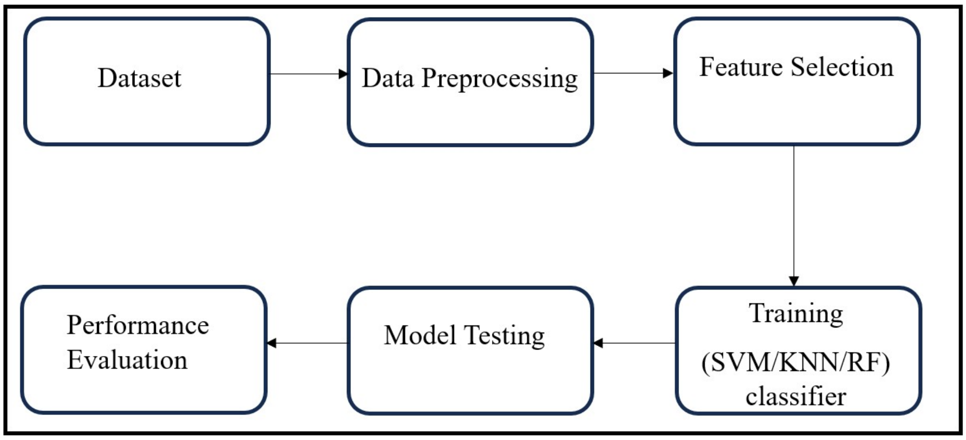
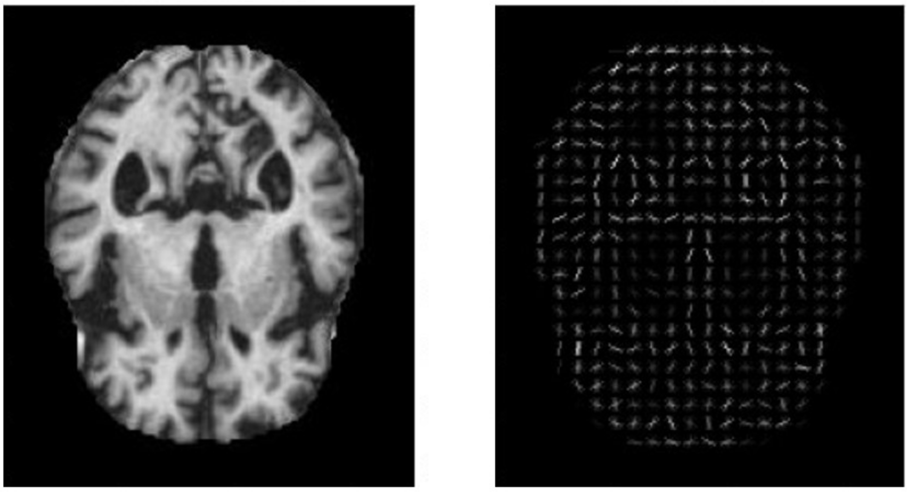


Figure 4.1: Flowchart for implementation of ML for classifying AD

This ML model is developed using HOG for feature selection and SVM for classification. Figure 4.1 gives the block diagram used for training and testing ML model. For preprocessing the images are converted from rgb to grayscale so that the number of computations are reduced, since the dataset is already preprocessed, hence no additional preprocessing is done. Feature selection is done using HOG and SVM is used as a classifier. The mathematical background of HOG and SVM are given below.

Histogram of Oriented Gradients

The Histogram of Oriented Gradients is an algorithm used for extracting features in an image. HOG computes pixel wise gradient and orientation, it stores these values in a vector and plots them on a histogram. It simplifies the representation of the image by retaining most relevant information. Figure 4.2 shows the output image after applying HOG to an image from dataset. Figure 4.2 (a) is the image from kaggleDataset1, Figure 4.2 (b) is the output after applying HOG to image (a).



(a) (b)

Figure 4.2: (a) Image from dataset, (b) Output image depicting the features of original image (a) that are selected using HOG.

To implement HOG, the gradient and orientation of each pixel is considered. For a pixel located at (x, y), the gradient of pixel along x axis is calculated by equation 4.1. Equation 4.2 is for calculating the gradient of pixel along y axis where $I_{(x,y)}$ is the intensity of the pixel. The magnitude of the gradient of a pixel is calculated using equation 4.3 using $G_{X(x,y)}$ and $G_{Y(x,y)}$. The orientation of gradient of the pixel ϕ is calculated by equation 4.4.

$$G_{X(x,y)} = I_{(x+1,y)} - I_{(x-1,y)} \tag{4.1}$$

$$G_{Y(x,y)} = I_{(x,y+1)} - I_{(x,y-1)} \tag{4.2}$$

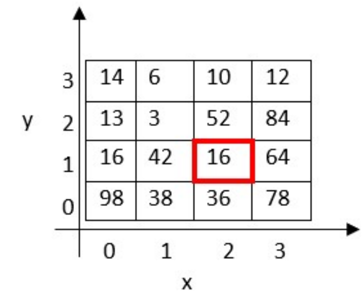
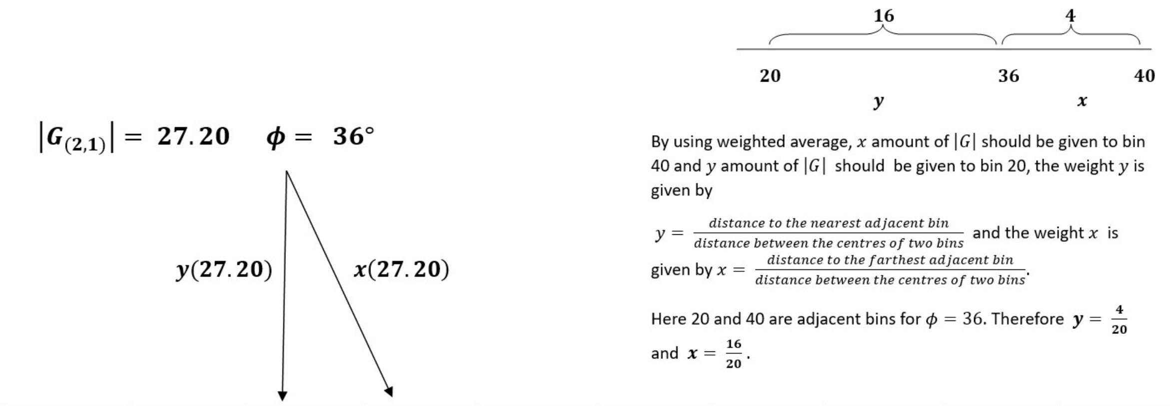


Figure 4.3: Pixel values of a H_i cell

$$| G_{(x,y)} | = \sqrt{G^2_{X(x,y)} + G^2_{Y(x,y)}} \tag{4.3}$$

$$\phi = \tan^{-1}(\frac{G_{Y(x,y)}}{G_{X(x,y)}}) \tag{4.4}$$



Magnitude		5.44	21.76						
Bin	0	20	40	60	80	100	120	140	160

Figure 4.4: Fitting of ϕ values

The values of G and ϕ are plotted on histogram and then normalised. To understand this process, consider a 64 x 64 image, divide the image into H cells of size 4 x 4. Let H_i be one of the cells with pixel values given as in Figure 4.3. For the pixel at position $(2, 1)$, equation 4.1 and 4.2 are used to calculate $G_{X(2,1)}$ and $G_{Y(2,1)}$ respectively. The value of magnitude of gradient obtained from substituting the values into equation 4.3 is 27.20 and the orientation obtained by substituting the values in equation 4.4 which is

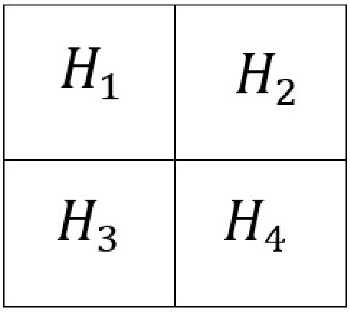


Figure 4.5: One N cell contains 4 H cells

equal to 36°. Orientation of any gradient lies between 0° to 180°, to hold the values of all the gradients at successive orientation levels, HOG prepares 9 bins which holds the values based on the orientation of pixels with each succesive bin having the difference of 20°. The fitting of the values in each bin is given as shown in Figure 4.4. First bin holds the sum of all the values of G that have orientation of 0°, this follows for rest of the bins. For each H_i a 1 x 9 matrix is formed which is also the feature vector of the cell. Let a_i be the HOG value of each bin then $v_i = (a_1, a_2, ..., a_9)$ is the feature vector of cell H_i For $\phi = 36°$ which does not fit in any bin is divided in such a way that the nearest bin to the value of ϕ gets most part of G and the adjacent bin receives the remaining part of G, Figure 4.4 shows the fitting of value of G. After calculation of HOG, for all H cells normalisation is done. For normalisation N cells of size 2 x 2 is taken where each N cell has 4 H cells as shown in Figure 4.5, the features i.e., HOG values of each H_i cell in one N_i cell are concatenated to get 1 x 36 matrix. The feature vector V of cell N_i is given by $V_i = (v_1, v_2, v_3, v_4)$ which is equal to $V_i = (a_1, a_2, a_3, ..., a_{34}, a_{35}, a_{36})$

$$k = \sqrt{\sum_{i=1}^{N=36} a_i^2} \tag{4.5}$$

A normalising factor k is taken which is given by equation 4.5. The values of V_i are divided by k to give the normalised value. Therefore the normalised vector is given by

$$V_{i_{normalised}} = (\frac{a_1}{k}, \frac{a_2}{k}, \frac{a_3}{k}, ..., \frac{a_{34}}{k}, \frac{a_{35}}{k}, \frac{a_{36}}{k})$$

One N cell holds 36 features. In a 64 x 64 image there are 64 N cells, hence the HOG feature of complete image is stored in 1 x 2304 matrix.

Support Vector Machine

The support vector machines are used for classification and regression problems in ML. For classification there are two types of SVM they are linear separable and non linear separable. In linear separable the classes can be divided by using a straight line called hyperplane and the dimensions in which they are placed depend on the number of classes, if the number of classes are considered as n then the number of hyperplanes used will be $n-1$ and the number of dimensions are equal to n. In non linear separable a hyperplane cannot be drawn hence the dimensionality is increased so that a hyperplane can be drawn to separate the classes, this is achieved by using different kernels. Two SVM kernels are used in this project, they are linear, polynomial (poly) and radial basis function (rbf) kernels.

Linear kernel: It is used for linearly separable classes, it is simple and computationally efficient hence it can be used for large datasets with linearly separable classes equation 4.6 is the representation of linear kernel where x and y are two feature vectors represented in row matrix and x^T is the transpose of row matrix x.

$$K(x, y) = x^T y \tag{4.6}$$

Polynomial kernel: It is used for classification of non linear separable classes. The kernel creates a higher dimensional feature space where the non linear classes can also be separated, the process of creating a higher dimensions is dependent on the degree of poly kernel. Equation 4.7 represents poly kernel where γ and r are hyperparameters where γ controls the scaling of dot product of x and y

$$K(x, y) = (\gamma x^T y + r)^d \tag{4.7}$$

rbf kernel: The rbf kernel calculates the similarity between data points using Gaussian function. It has the ability to handle complex non-linear separable dataset effectively. To achieve the optimal performance the γ value is tuned, equation 4.8 gives the mathematical representation of rbf kernel where $\|x - y\|^2$ is the euclidean distance between the data points x and y.

$$K(x, y) = \exp\left(-\gamma \|x - y\|^2\right) \tag{4.8}$$

The model is trained and tested against kaggleDataset1 and kaggleDataset2, the results of the tests conducted on the model and the performance evaluation are discussed in the chapter 5.

4.1.2 ML model using KNN and RF classifiers

Data Preprocessing

The data preprocessing is done to ensure the consistency of the machine learning model. Images are loaded and resized to 64x64 pixels. Feature extraction transforms the images into numerical data. Pixels are flattened into one-dimensional arrays. This allows machine learning models to analyze patterns from the data. The processed data is then ready for training and evaluating models to detect Alzheimer's Disease.

Data Augmentation

Data augmentation is a technique to create more training data from existing images. This is especially helpful for small datasets to prevent overfitting and improve model performance. It works by applying transformations to original images.
Three transformations are used in this code:

- Horizontal Flip: Creates a mirror image of the original.
- Random Rotation: Rotates the image by a small random angle.
- Random Zoom: Zooms in or out on the image by a small random amount.

These transformations create new images with slightly different appearances but maintain the original content. After applying these transformations, the images are resized to a standard size for the machine learning model.

KNN classifier

The KNN algorithm is a simple yet powerful tool in machine learning used for classification. KNN makes predictions based on the similarity of data points. Consider a two class dataset and let Figure 4.6 represent the dataset when plotted on a graph and let the blue circle be the new data point. For a new point, KNN checks the K closest known points (marked in red dashed circles in Figure 4.6) and assigns the new point to the class which have majority of its neighbors nearest to the neighbors. This makes it useful for various tasks like pattern recognition and doesn't require complex assumptions about the data.

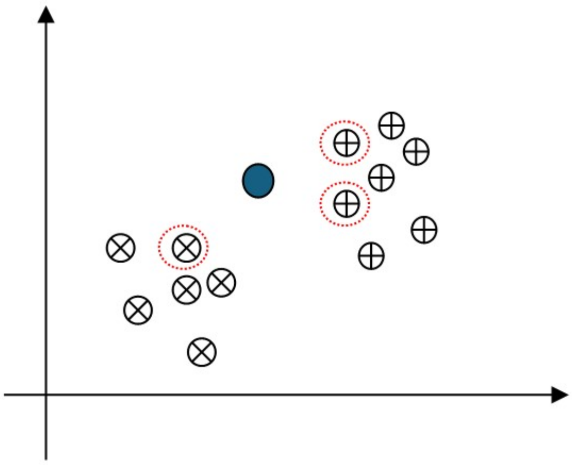


Figure 4.6: Representation of two class dataset, the circles with crosses belong to one class and circles with plus sign belong to another class and the blue circle is the new data point that is to be classified.

KNN algorithm uses the nearest points for classification, to determine the nearest point the euclidean measuring metric is used and it is given by equation 4.9 where D is the distance between the new data point and the nearest neighbor of a point in dataset, where x is the training dataset with m data points and each data point is represented by a n dimensional feature vector x_j and X is the new data point.

$$D = \sqrt{\sum_{i=1}^{n}(X_i - x_{j_i})^2} \tag{4.9}$$

The K-Nearest Neighbors algorithm works by first calculating the distances between a new data point and all the points in the training data. Then, it selects the K closest data points (neighbors) based on this distance. For classification tasks, the new point is assigned the most frequent class label among its neighbors. Imagine voting among your closest neighbors - who wins determines the class. In regression tasks, the KNN algorithm predicts a value by averaging the values of its neighbors, essentially letting them speak up to influence the prediction. KNN is easy to implement and understand. Unlike some models that require retraining, KNN constantly adapts to new information by incorporating all available data points for prediction. This makes it flexible and potentially improves results over time. Additionally, KNN keeps the configuration process simple. You only need to choose two key parameters: the number of neighbors (k) to consider and a distance metric to measure similarity between data points. This reduces the complexity of fine-tuning numerous parameters often encountered in other algorithms.

RF classifier

Random Forest is a machine learning algorithm that combines the strengths of multiple decision trees. Consider a group of people working on a problem, each person building a unique decision tree to solve a problem. Random Forest creates a collection of these trees (also called as "forest"), training each on a different random subset of data and features. This diversity prevents the trees from becoming too alike and reduces overfitting. When making predictions, the entire forest collaborates by classification or regression. This ensemble approach yields more robust and accurate results compared to a single decision tree, making Random Forest a popular choice for tackling complex data in classification and regression tasks.
Random Forest builds a "forest" of decision trees, each one unique. To prevent them from being too similar, randomness is injected in two ways:

- Random Feature Selection: Each tree considers only a random subset of features when making splits.
- Bootstrapping: Different training datasets are created by randomly sampling the original data (with replacement) for each tree.

4.2 Deep Learning model

In DL model different stages are represented using different layers. Figure 4.7 gives the flowchart for the implementation of DL. The concept of CNN is implemented to obtain a DL model. CNN's have multiple layers that process and extract features from data. CNN's have a Rectified Linear Unit (ReLU) layer to perform operations on elements. The output is a rectified feature map. Optimizers are used which act as the guiding force during training, suggesting how much and in which direction to adjust the model's

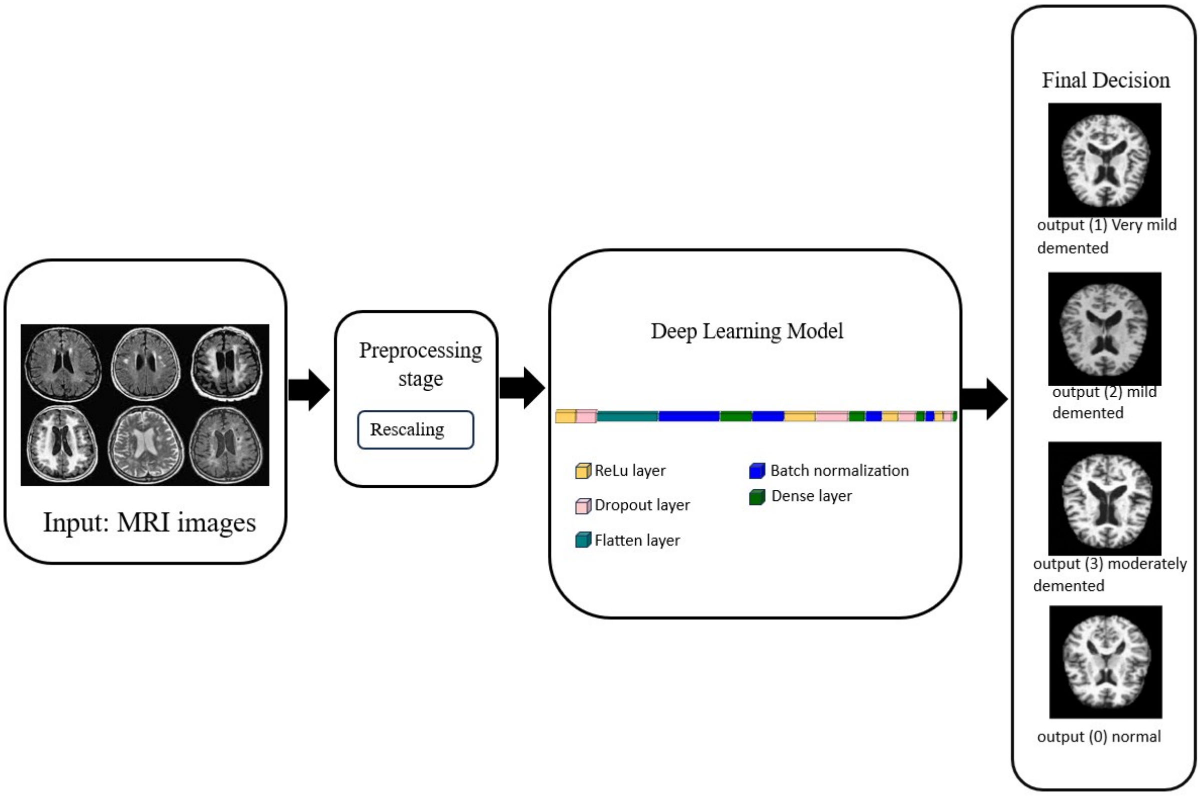


Figure 4.7: Flowchart of DL model built for classifying AD

internal parameters to steer it towards achieving the best possible performance on the task at hand. They play a vital role in the learning process of neural networks. Two convolutional neural network architectures are taken for image classification they are :

- DenseNet169
- MobileNetV2

These architectures are used to improve accuracy by utilizing fewer parameters. These architectures are pre-trained on large Datasets like ImageNet, this significantly improve performance and reduce training time.

4.2.1 DenseNet169 neural network

DenseNet169 is a convolutional neural network architecture known for its effectiveness in image classification tasks. It incorporates a concept called "dense connectivity," which significantly differs from traditional CNNs, some of the features of DenseNet169 are as follows.

Dense Connections for Enhanced Learning

In standard CNNs, information flows linearly from one layer to the next. DenseNets, however, connect each layer to all preceding layers in a feed-forward manner. This fosters stronger propagation of features throughout the network, allowing each layer to benefit from the knowledge learned by previous layers. It also promotes feature reuse, improving efficiency.

Overcoming the Vanishing Gradient Challenge

Training deep CNNs can be hindered by the vanishing gradient problem, where gradients weaken as they backpropagate through the network. Dense connections help alleviate this issue by providing more direct paths for gradients to flow, enabling the network to learn from deeper layers more effectively.

Efficiency with Fewer Parameters

Despite its dense connectivity, DenseNet169 achieves high accuracy with a surprisingly low number of parameters compared to other CNN architectures. This makes it a more efficient choice, particularly for tasks where training data might be limited.

Leveraging Pre-trained Knowledge

DenseNet169 models are frequently pre-trained on massive image datasets like ImageNet. This pre-training equips the network with a strong foundation for image recognition and classification tasks, even when applied to entirely new datasets. This pre-training can significantly reduce training time and improve performance on new problems.

4.2.2 MobileNetV2

MobileNetV2 is a convolutional neural network (CNN) architecture specifically designed for resource-constrained devices. Unlike traditional CNNs that can be bulky and high power consuming, MobileNetV2 prioritizes efficiency and compact size while maintaining good accuracy on image classification tasks. The features of the MobileNetV2 are as follows.

Streamlined Processing with Depthwise Separable Convolutions

Regular convolutions are a workhorse in image recognition, but they can be computationally expensive for mobile devices. MobileNetV2 tackles this by using depthwise separable convolutions. These break down the convolution process into two parts:

1. **Depthwise Convolution:** This extracts features from each input channel using a single filter, making it highly efficient.

2. **Pointwise Convolution:** This combines the extracted features using 1x1 filters, allowing for further processing without a significant computational burden.

Inverted Residual Blocks for Efficient Learning

MobileNetV2 incorporates inverted residual blocks, a clever twist on the residual block concept used in many CNNs. These blocks allow the network to learn from both the original input and modified versions within the block, improving accuracy. However, unlike traditional residual blocks, MobileNetV2 places the layer with the most filters in the middle. This allows for efficient processing with slimmer "bottleneck" layers at the beginning and end, reducing computational cost.

Linear Bottlenecks for Optimized Representation

Within these inverted residual blocks, MobileNetV2 utilizes linear bottlenecks. These are layers with a smaller number of filters compared to the input and output layers of the block. By strategically reducing complexity, MobileNetV2 maintains good representational power for the data while minimizing computations.
Similar to DenseNet169, MobileNetV2 also utilises the pre-trained knowledge for various tasks like object recognition and image classification, even when applied to entirely new datasets thus reduce training time and improve performance on new problems.

4.2.3 Layers breakdown in deep learning model

The deep learning model developed (represented in Figure 4.9) has a total of seventeen layers are used they are given as follows:

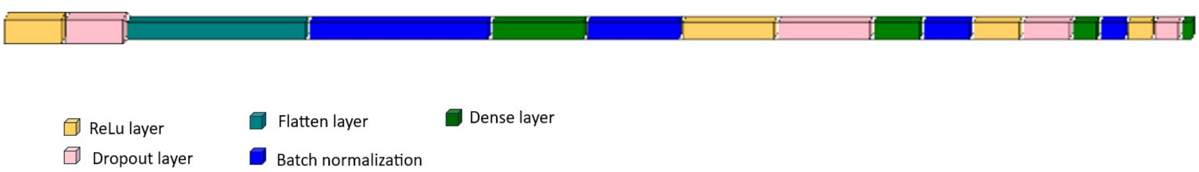


Figure 4.8: Layers of DL model architecture for classifying AD

1. **Base Model:** This includes the pre-trained model that is used for feature extraction, the models used are DenseNet169 and MobileNetV2, pre-trained weights on ImageNet dataset is used which helps to recognise basic image features.

2. **Rectified Linear Unit activation:** Rectified Linear Unit (ReLu) layer is used as an activation function in deep neural networks. The ReLU function operates on each element of its input independently. For any input value (x), it outputs as $f(x) = max(0, x)$ In simpler terms, the ReLu activation sets all negative input values to zero and leaves the positive values unchanged. This creates a threshold at zero, where anything below gets rectified to zero, while positive values pass through unaffected.

3. **Dropout:** This layer drops neurons during traing to prevent overfitting.

4. **Flatten:** The output of the convolutional layers are in 3D, to convert 3D data into 1D data this layer is used. 1D vector is useful in feeding into fully connected layers.

5. **Batch Normalization:** Adds Batch Normalization to improve training stability and speed.

6. **Dense layers:** Adds a fully connected layer with large number of neurons and he_uniform is used as initializer for weights.

7. The layers Batch Normalization, Activation and Dropout are repeated three times with decreasing numbers of neurons (2048, 1024, 512) these are dense layers that learn complex relationships between features.

8. **Dense:** Adds the final output layer with 4 neurons and the "softmax" activation for multi-class classification (corresponding to four different stages of Alzheimer's disease).

Softmax

The softmax function takes the raw outputs and transforms them into a probability distribution between 0 and 1 for each class. It ensures the sum of all these probabilities adds up to 1. Mathematically, it calculates the exponentials of the input values, divides them by the sum of all the exponentials, making the largest output value tend towards 1 (most probable class) and the others towards 0. After applying softmax, the output layer of the network produces a vector of probabilities. Each value represents the probability of the input data point belonging to a specific class.

Optimizers

Two optimizers at different learning rates are used to check the performance, a total of twelve variants of the model is obtained which have different optimizers and learning rate, the optimizers used are Adam and Adamax, the learning rates used are 0.01, 0.001, 0.0001. These learning rates and optimizers gives different results when the dataset is presented to the neural network.

4.3 Software tools used in the project

This section is about the software that will be used for conducting the project. The ML and DL model is written in python programming language. To run the machine learning models the google colab platform is used. Jupyter notebook is used to run the deep learning algorithm.

4.3.1 Python

Python is a general-purpose, adaptable programming language that has emerged as the clear front-runner for deep learning and machine learning. It has an appealing, legible syntax that is similar to plain English. When compared to denser languages like C++, this clarity makes the language easier to understand and write code with. This ease of use is crucial for prototyping and iterating through models, which are core activities in machine learning. Python's extensive ecosystem of libraries provides readily available tools for every step of the machine learning workflow. From data wrangling with Pandas to training deep learning models with TensorFlow or PyTorch, Python libraries do the heavy lifting, allowing you to focus on the algorithm and problem at hand. Active and supportive community of python programmers contributes to its constant evolution and vast resources. Compared to other languages, Python offers several advantages:
Faster development: Its conciseness allows rapid prototyping and experimentation.
Readability and maintainability: Code is easier to understand and modify, crucial for collaboration.
Large and diverse libraries: Libraries exist for virtually every data science and machine

learning task.

Active community: Abundant resources and support are readily available.

Python's dynamic nature can sometimes lead to performance constraints. However, its overall benefits far outweigh any drawbacks, making it the language of choice for aspiring and experienced machine learning practitioners alike.

4.3.2 Google Colaboratory

Colab provides a platform for writing and running code, it runs on Google's cloud servers, giving users access to powerful hardware resources for computationally intensive tasks like machine learning and data analysis. Collab is a free and open source product, however there is a hardware limit to the free version, by subscribing to Google Pro and Google Pro+ products, larger computations can be done easily. Colab notebooks can be easily shared with others, allowing for collaborative work on projects. The advantages of using Google Colab:

Accessibility: Colab runs on Google's powerful servers, giving access to GPUs and even TPUs (Tensor Processing Units) for lightning-fast processing. It can be accessed at any device with internet connectivity which has Google Chrome, Fire fox and other web browsers that are based on Fire fox.

Ease of use: Colab comes pre-loaded with popular libraries like TensorFlow, PyTorch, and scikit-learn. To implement different ML and DL algorithms, various packages and tools that are required for training and testing the models can be easily imported. Application Programming Interface (API) can also be utilized to run certain tools that are provided via API.

Collaboration made easy: Colab notebooks are shareable, allowing multiple users to work on the same model simultaneously.

4.3.3 Jupyter notebook

Jupyter notebook is a platform where anyone can code seamlessly with data explorations and rich visualizations, all in one interactive environment. It is a web-based playground for data scientists, educators, and anyone passionate about working on information. Jupyter notebooks are like versatile canvases, where user can code in various languages,run and do step-by-step analysis, and embed stunning visuals like charts and graphs. It is a open-source and free to use software that can be installed in any computer or laptop, the training and testing of ML and DL model can be done and by using appropriate amount of memory and Graphic Processing Unit the time taken for training can be reduced.

Chapter 5

Experimental Results

This chapter discusses about the results obtained after running the models. Models are tested against the kaggleDataset1 and kaggleDataset2. For testing 20% of the dataset is taken. Severe case of AD is represented by the class Moderate demented. The comprehensive overview of classifier and model performance by analyzing precision, recall, F1-score, and accuracy metrics across distinct categories is shown in Table 5.1. The DL model is trained and tested against OASIS dataset and the performance is tabulated in Table 5.2.

Table 5.1: Performance of the ML models that are tested against two datasets

Dataset	**Classifier**	**Kernel**	**Accuracy**	**Precision**	**Recall**	**F1-score**
kaggleDataset1	**SVM**	**linear**	0.940	0.962	0.957	0.96
		poly	0.98	0.99	0.98	0.985
		rbf	0.85	0.915	0.677	0.7325
	RF	-	0.8317	0.9049	0.6170	0.6751
	KNN	-	0.9013	0.9402	0.8581	0.8935
kaggleDataset2	**SVM**	**linear**	0.91	0.94	0.9325	0.9375
		poly	0.97	0.9827	0.9775	0.9825
		rbf	0.87	0.9225	0.7775	0.8325
	RF	-	0.8309	0.9016	0.6082	0.6595
	KNN	-	0.9054	0.9385	0.8568	0.8926

Table 5.1 compares SVM, RF and KNN classifiers which are tested against kaggleDataset1 and kaggleDataset2 dataset. From Table 5.1 it is seen that the highest performance is obtained for SVM classifier with poly kernel for both the datasets. Figure 5.1 gives the confusion matrix for SVM with poly kernel obtained after testing the model against kaggleDataset1, in this case the total number of images taken for testing is 1025 which is 20% of the total number of images in the dataset. In the Figure 5.1, for moderate demented a total of 16 images are taken for testing and all the 16 images are classified as moderate demented. Similarly for mild demented 159 images are tested out of which

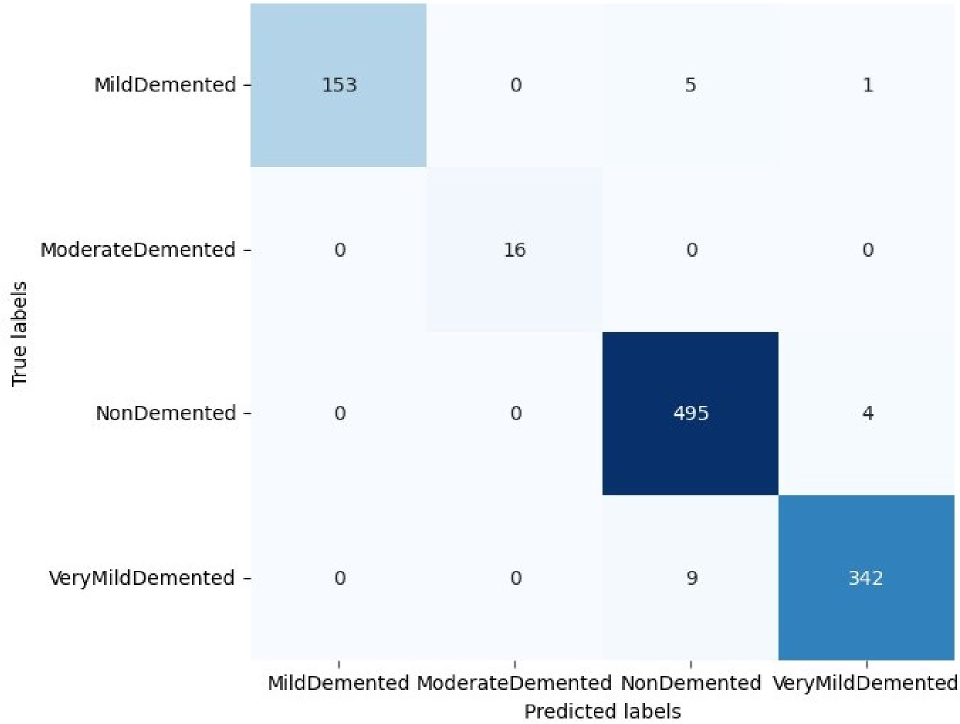


Figure 5.1: Confusion matrix of model tested using poly kernel of SVM classifier on kaggleDataset1

153 are classified as mild demented, 1 is misclassified as very mild demented and 5 are misclassified as non demented. Figure 5.2 gives the confusion matrix for SVM with poly kernel obtained after testing the model against kaggleDataset2, here the number of images used is 1280. Here 6 images of moderate demented class are tested against the model and all 6 classify into moderate demented class. For mild demented 201 images are considered for testing and out of 201, 193 are classified as mild demented, 5 are misclassified as very mild demented and 3 are misclassified as non demented. This model performs more efficiently than RF and KNN classifiers.

Figure 5.3 and Figure 5.4 gives the confusion matrix that has more efficient classification of Moderate demented class when KNN and RF classifiers are used against kaggleDataset1 and kaggleDataset2 respectively, for kaggleDataset1 when it is tested against KNN classifier, 25 images are taken from moderate demented class to test against the model and out of 25, 21 images classify as moderate demented class, 1 image is misclassified as mild demented class, 2 are misclassified as very mild demented class and 1 image is misclassified as non demented class hence it gives 84% accuracy while classifying for Moderate Demented class, for kaggleDataset2 the accuracy obtained by using KNN classifier is 75%,

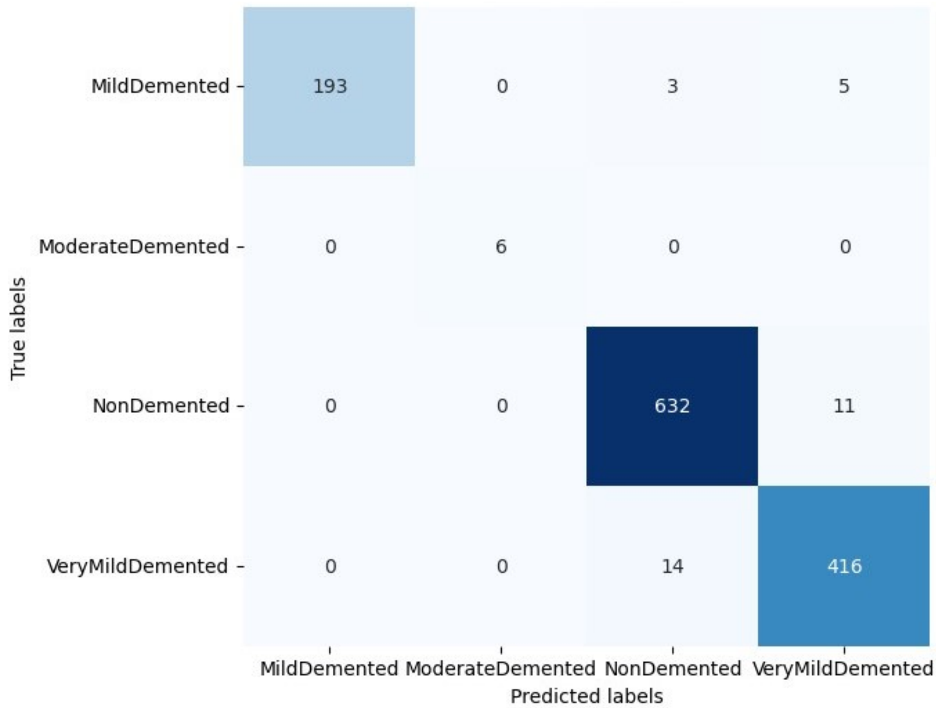


Figure 5.2: Confusion matrix of model tested using poly kernel of SVM classifier on kaggleDataset2

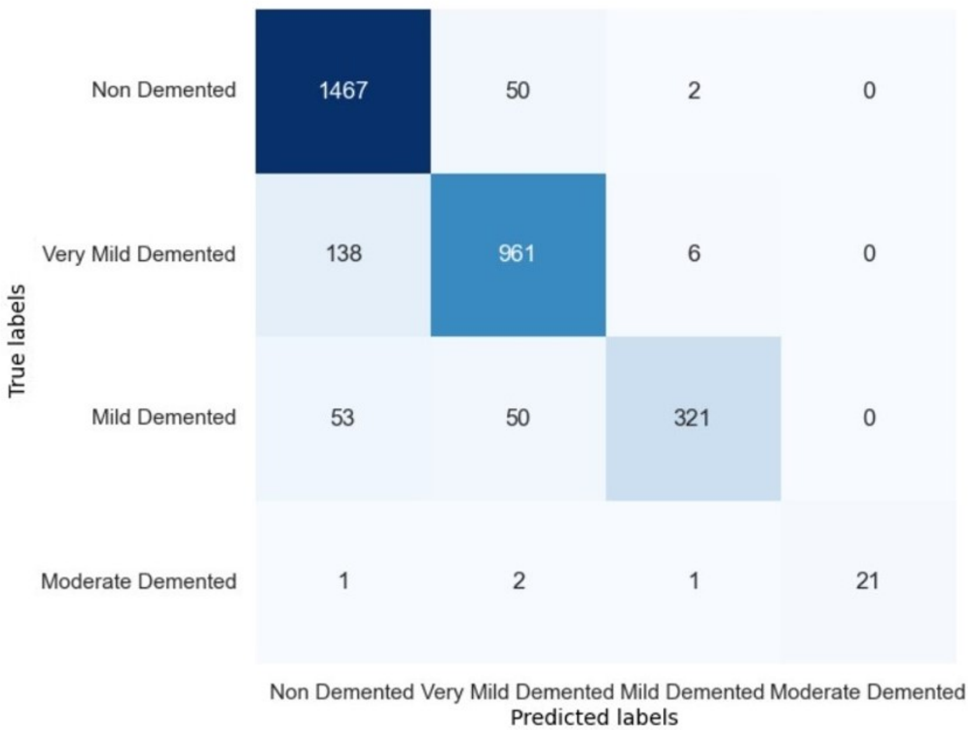


Figure 5.3: Confusion Matrix obtained for KNN classifier after testing the ML model on kaggleDataset1

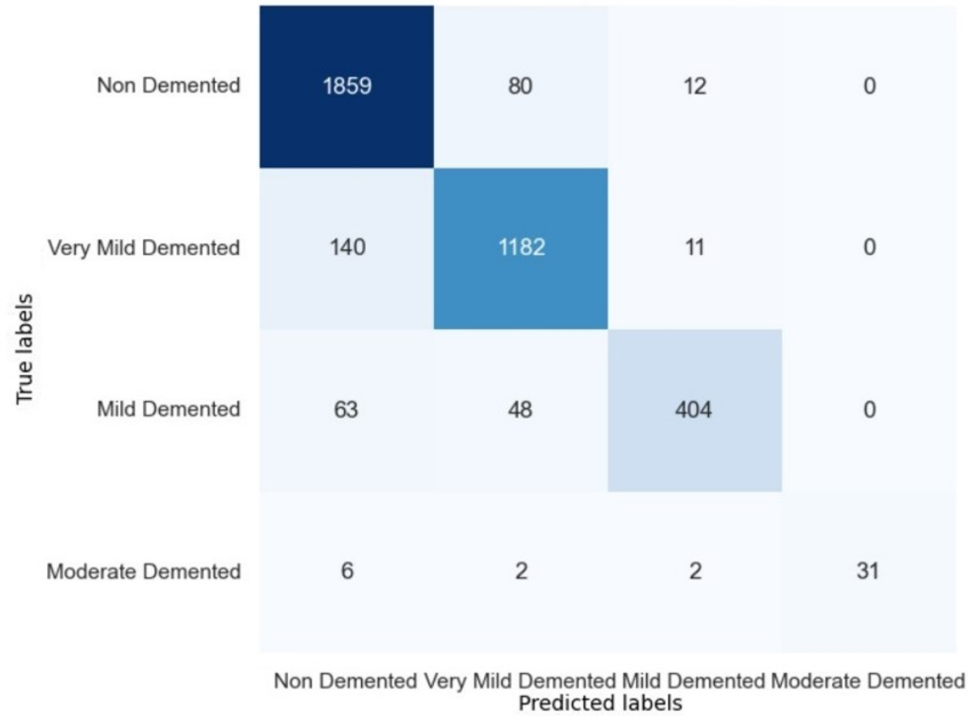


Figure 5.4: Confusion Matrix obtained for KNN classifier after testing the ML model on kaggleDataset2

from Figure 5.4, for moderate demented 41 images are taken for testing and out of which 31 are classified as moderate demented, 2 are misclassified as mild demented, 2 and 6 images are misclassified as very mild and non demented respectively. From Table 5.1 it can be seen that the poly kernel gives the accuracy of 98%, precision of 99%, recall of 98% and f1-score of 98.5% for kaggleDataset1. For kaggleDataset2 the accuracy obtained is 97%, the precision is 98.27%, recall is 97.75% and the f1-score is 98.25%. Poly kernel gives more accurate classification because the kernel is able to increase dimensionality according to the number of classes, hence there is clear distinction in the classes. The confusion matrix of the highest performing models against kaggleDataset1 and kaggleDataset2 are given in Figure 5.1 and Figure 5.2 respectively.

For DL model DenseNet169 and MobileNetV2 are used. By varying the learning rate and using two optimizers the performance of the model is evaluated, Table 5.2 gives the performances of each variant of the model. From Table 5.2 it is observed that Adam optimizer with learning rate 0.0001 using MobileNetV2 gives the highest accuracy among all the other variants with the accuracy of 95.28%, the variant with Adamax optimizer with learning rate 0.0001 using MobileNetV2 gives the least accuracy with accuracy of

Table 5.2: Accuracy of the different variants of DL model tested against OASIS dataset

Optimizers	**Adam**			**Adamax**		
Learning rate	0.01	0.001	0.0001	0.01	0.001	0.0001
Densenet169	0.9448	0.9428	0.9415	0.9329	0.9369	0.9063
MobileNetV2	0.9255	0.9267	0.9528	0.9300	0.9285	0.8984

Table 5.3: Accuracy of each class of the variants of DL model at different learning rate

		Adam			**Adamax**		
		0.01	**0.001**	**0.0001**	**0.01**	**0.001**	**0.0001**
DenseNet169	**Non Demented**	0.98	0.97	0.97	0.99	0.98	0.93
	Very Mild Demented	0.81	0.77	0.82	0.69	0.78	0.82
	Mild Demented	0.81	0.83	0.87	0.86	0.85	0.89
	Moderate Demented	0.96	0.86	0.97	0.96	0.95	0.86
MobileNetV2	**Non Demented**	0.97	0.98	0.98	0.97	0.97	0.96
	Very Mild Demented	0.77	0.71	0.86	0.77	0.77	0.69
	Mild Demented	0.77	0.81	0.85	0.83	0.82	0.72
	Moderate Demented	0.91	0.90	1.0	0.86	0.94	0.68

89.84% . Figure 5.5 gives the confusion matrix of the highest performing variant of the model. This variant classifies the severe case of dementia more accurately than any other variant with only 2 images classified as very mild demented among 488 images making the accuracy of the classification as 99.59%. For DenseNet169 the variant using Adam optimizer and learning rate of 0.0001 gives the highest accuracy for DenseNet169 architecture when tested against OASIS dataset. The confusion matrix of this variant is given in Figure 6.6, from the Figure 5.6 it is seen that for severe case the classification is not as good as the MobileNetV2 here out of 488 only 473 are classified as moderate demented, 9 are misclassified as mild demented, 4 are misclassified as very mild demented, and 2 are misclassified as non demented, that makes this variant 96.926% accurate in classifying the severe case test image when it is presented.

Table 5.3 gives the performance of each variant of the deep learning model for every class, the accuracy for each class is at different learning rates and two different optimizers is

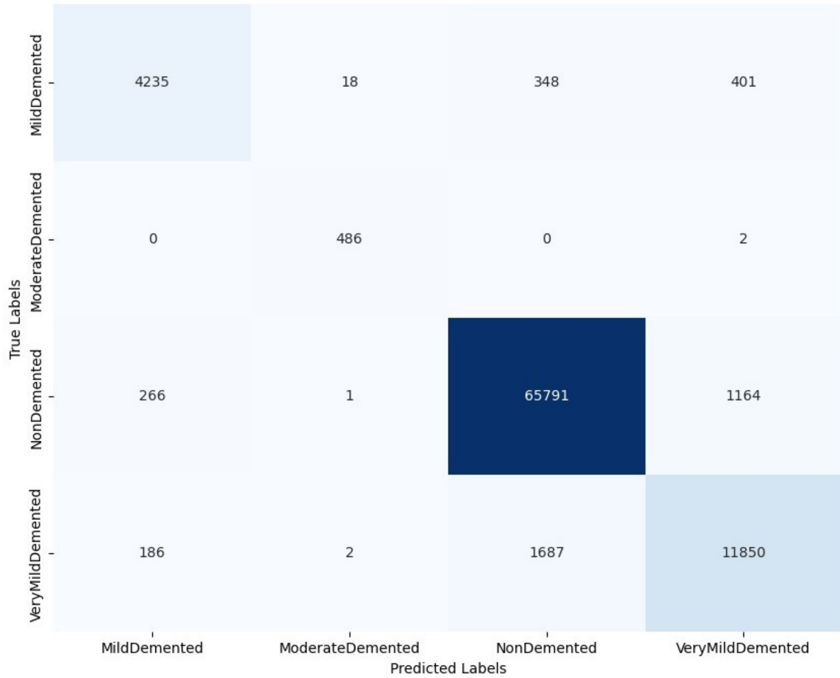


Figure 5.5: Confusion matrix of model tested using MobileNetV2 and Adam optimizer with learning rate 0.0001 on OASIS dataset

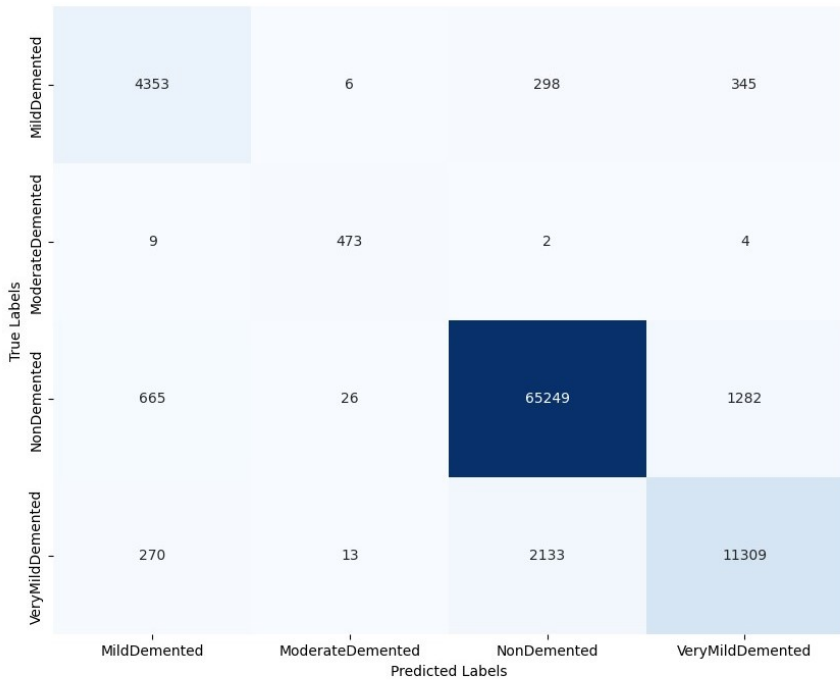


Figure 5.6: Confusion matrix of model tested using DenseNet169 and Adam optimizer with learning rate 0.0001 on OASIS dataset

tabulated. From the Table 5.3 it is observed that for severe case i.e., Moderate dementia, the DL model with MobileNetV2 and learning rate of 0.0001 using Adam optimizer gives the maximum accuracy of 95.28%, for DenseNet169, the same variant with learning rate of 0.01 is seen to give the maximum accuracy with accuracy of 94.48. Comparing this overall accuracy from Table 5.2 it is observed that even though the accuracy for DenseNet169 with learning rate 0.0001 using Adam optimizer has less accuracy with 94.15%, but it is good at classifying severe case of AD. Overall DL model with MobileNetV2 and learning rate of 0.0001 using Adam optimizer gives higher accuracy than any other variant used.

A web page for guessing the test image's class is made in order to enable the model to be used for prediction on subsequent test images. On the webpage's initial page, the user can upload an image and select a file from the local system where the website is hosted. A preview of the test image can be viewed by selecting the file and then clicking the upload button. After selecting Upload, the web application is shown in Figure 5.7. By selecting the necessary test file and pressing the "choose file" button once more, the user can alter the test image. The predict button is clicked if the user want to guess the test image's class after uploading. The web application reroutes to the results page when hitting the forecast button. The webpage displaying the projected class is shown in Figure 5.8. The anticipated class is indicated here in red. There is information on each class on this page as well. The upload another image button allows the user to reroute to the web application's home page and predict class for additional test photographs.

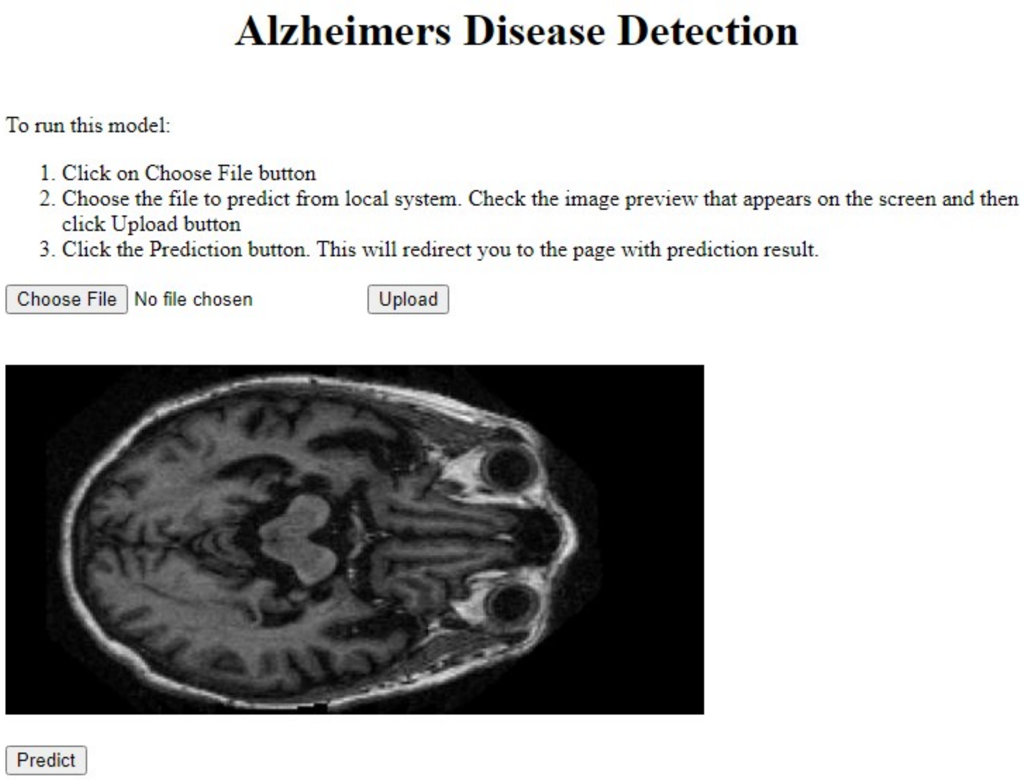
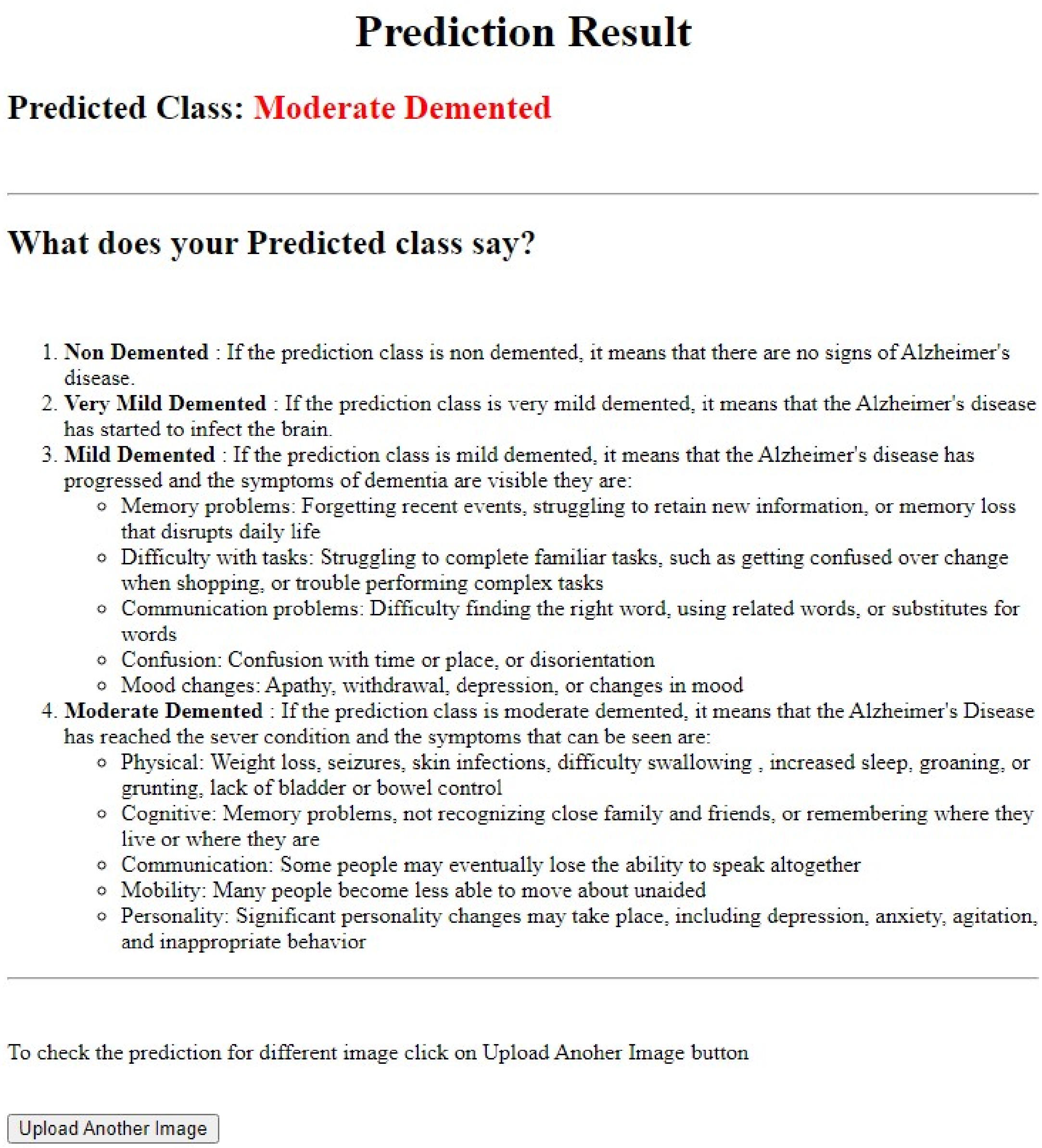


Figure 5.7: The first page of web page showing the preview of image uploaded.



Prediction Result

Predicted Class: Moderate Demented

What does your Predicted class say?

1. **Non Demented** : If the prediction class is non demented, it means that there are no signs of Alzheimer's disease.
2. **Very Mild Demented** : If the prediction class is very mild demented, it means that the Alzheimer's disease has started to infect the brain.
3. **Mild Demented** : If the prediction class is mild demented, it means that the Alzheimer's disease has progressed and the symptoms of dementia are visible they are:
   - Memory problems: Forgetting recent events, struggling to retain new information, or memory loss that disrupts daily life
   - Difficulty with tasks: Struggling to complete familiar tasks, such as getting confused over change when shopping, or trouble performing complex tasks
   - Communication problems: Difficulty finding the right word, using related words, or substitutes for words
   - Confusion: Confusion with time or place, or disorientation
   - Mood changes: Apathy, withdrawal, depression, or changes in mood
4. **Moderate Demented** : If the prediction class is moderate demented, it means that the Alzheimer's Disease has reached the sever condition and the symptoms that can be seen are:
   - Physical: Weight loss, seizures, skin infections, difficulty swallowing , increased sleep, groaning, or grunting, lack of bladder or bowel control
   - Cognitive: Memory problems, not recognizing close family and friends, or remembering where they live or where they are
   - Communication: Some people may eventually lose the ability to speak altogether
   - Mobility: Many people become less able to move about unaided
   - Personality: Significant personality changes may take place, including depression, anxiety, agitation, and inappropriate behavior

---

To check the prediction for different image click on Upload Anoher Image button

Upload Another Image

Figure 5.8: The second page of web page where the predicted class is displayed.

Chapter 6

Conclusion

AD early detection is a difficult task. This study considers the brain MRI dataset since it provides additional insights into the condition of the brain. Based on the severity of the AD, classification is carried out. Four distinct classes—Non Demented, Very Mild Demented, Mild Demented, and Moderate Demented represent the severity. Machine learning employs two models. In the first model, SVM serves as the classifier and HOG is utilized for feature selection. In the second model, KNN and RF classifiers are used in conjunction with data augmentation. When testing an ML model on the kaggleDataset1, the SVM classifier with a poly kernel yielded the highest accuracy of 98%. The DL model is evaluated using the OASIS dataset with DenseNet169 and MobileNetV2 image classifier architectures with Adam and Adamax optimizers. Using the Adam optimizer and a learning rate of 0.0001, the MobileNetV2 architecture achieved the highest accuracy of 95.28%.

6.1 Scope for future work

- A more efficient model can be developed that can work on more diversified dataset of brain MRI.
- DL model can be improved so that it can work with less number of data in dataset.
- Accuracy of DL model can be improved by tuning the model.

Bibliography

[1] Scheltens P, De Strooper B, Kivipelto M, Holstege H, Chételat G, Teunissen CE, Cummings J, van der Flier WM. "Alzheimer's disease". *The Lancet*, vol. 397, pp. 1577-1590, Apr 24 2021.

[2] Trejo-Lopez JA, Yachnis AT, Prokop S. "Neuropathology of Alzheimer's disease". *Neurotherapeutics*, vol. 19, pp. 173-185, Jan 1 2023.

[3] Amini M, Pedram M, Moradi A, Ouchani M. "Diagnosis of Alzheimer's disease severity with fMRI images using robust multitask feature extraction method and convolutional neural network (CNN)". *Computational and Mathematical Methods in Medicine*, vol. 14, pp. 71-86, Apr 27 2021.

[4] Kavitha C, Mani V, Srividhya SR, Khalaf OI, Tavera Romero, "Early-Stage Alzheimer's Disease Prediction Using Machine Learning Models." *Front. Public Health* , vol. 10, pp. 296-313, Mar 32022.

[5] Rao, Kongala Nageswara, et al. "Prediction and Classification of Alzheimer's Disease using Machine Learning Techniques in 3D MR Images." *2023 International Conference on Sustainable Computing and Smart Systems (ICSCSS). IEEE*, pp. 85-90, June 2023.

[6] Shukla, Amar, Rajeev Tiwari, and Shamik Tiwari. "Review on alzheimer disease detection methods: Automatic pipelines and machine learning techniques." *Sci*, vol 5, pp. 13-18, March 2023.

[7] Dara, Omer Asghar, et al. "Alzheimer's Disease Diagnosis Using Machine Learning: A Survey." *Applied Sciences*, vol. 13, pp. 8298-8311, July 2023.

[8] Shetty, Mangala, Manisha Bhat, and Manisha Devadiga. "Detection of Alzheimer's Disease Using Machine Learning." *2022 International Conference on Artificial Intelligence and Data Engineering (AIDE)*. IEEE, pp. 117-120, Dec 2022.

[9] Nakul Pranao D, "Alzheimer's Disease Prediction Using Machine Learning Methodologies." *2022 International Conference on Computer Communication and Informatics (ICCCI). IEEE*, pp. 1-6 , Jan. 2022.

[10] Amrutesh A, CG GB, Amruthamsh A, KP AR, Gowrishankar S. "Alzheimer's Disease Prediction using Machine Learning and Transfer Learning Models" *2022 6th International Conference on Computation System and Information Technology for Sustainable Solutions (CSITSS). IEEE*, pp. 1-6, Dec. 2022.

[11] Basha SM, Ashalatha G, Prasad SD, Al-Hameed MR, Zearah SA, Saikumar K. "A Study and Recognition of Alzheimer's Disease Using Machine Learning" *2023 IEEE 8th International Conference for Convergence in Technology (I2CT). IEEE*, pp. 1-6, Apr. 2023.

[12] Alroobaea R, Mechti S, Haoues M, Rubaiee S, Ahmed A, Andejany M, Bragazzi NL, Sharma DK, Kolla BP, Sengan S. "Alzheimer's Disease Early Detection Using Machine Learning Techniques" *Research Square*,pp. 1-16, Jun. 2021.

[13] Shahbaz M, Ali S, Guergachi A, Niazi A, Umer A. "Classification of Alzheimer's Disease using Machine Learning Techniques" *Data*, pp. 296-303, Jul. 2019.

[14] Battineni G, Hossain MA, Chintalapudi N, Traini E, Dhulipalla VR, Ramasamy M, Amenta F. "Improved Alzheimer's disease detection by MRI using multimodal machine learning algorithms." *Diagnostics* , vol. 11, pp. 2103-2125, Nov. 2021.

[15] Rajendran, P. Selvi. "Alzheimer's Disease Prediction with K-means Clustering and Reinforcement Learning Approach" *2023 International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE). IEEE*, pp. 1-6, April 2023.

[16] S. Al-Shoukry, T. H. Rassem and N. M. Makbol, "Alzheimer's Diseases Detection by Using Deep Learning Algorithms: A Mini-Review," *IEEE Access*, vol. 8, pp. 77131-77141, April 2020.

[17] El-Latif, Ahmed A. Abd, et al. "Accurate Detection of Alzheimer's Disease Using Lightweight Deep Learning Model on MRI Data." *Diagnostics* , vol 13, pp. 1216-1220, March 2023.

[18] Devvi Sarwinda, and Alhadi Bustamam. "3D-HOG Features–Based Classification using MRI Images to Early Diagnosis of Alzheimer's Disease." *IEEE ICIS,*, pp. 457-462, June 2018.

[19] Helaly, Hadeer A., Mahmoud Badawy, and Amira Y. Haikal. “Deep learning approach for early detection of Alzheimer’s disease.” *Cognitive computation* , vol. 14, pp.1711-1727, Nov. 2021.

[20] Al Shehri, Waleed. “Alzheimer’s disease diagnosis and classification using deep learning techniques.” *PeerJ Computer Science* , vol 8, pp. 1177-1181, Dec. 2022.

[21] Zhao, Zhen, et al. “Conventional machine learning and deep learning in Alzheimer’s disease diagnosis using neuroimaging: A review.” *Frontiers in Computational Neuroscience* , vol 17, Feb. 2023.

[22] Al Saeed, Duaa, and Samar Fouad Omar. “Brain MRI analysis for Alzheimer’s disease diagnosis using CNN-based feature extraction and machine learning.” *Sensors* , vol. 22, pp. 2911-2916, Apr. 2022.

[23] Zhiyuan Ding, Ling Wang, Yan Liu, Xiangzhu Zeng, and Zheng Wang. “Spatial Pyramid Based 3D Hog Feature Extraction for Alzheimer’s Disease Idendfication.” *IEEE International Conference on Signal Processing (ICSP)*, pp. 489-494, 2020.

Self Assessment of Project			
	PO PSO	Contribution from the project	Level
1	Engineering Knowledge: Knowledge of mathematics, engineering fundamentals engineering specialization to form of complex engineering problems	To train Deep learning model to detect severity of AD.	4
2	System Analysis: Identity, formulate, research literature, and analyse engineering problems to derive substantiate conclusions by first principles of mathematics, natural and engineering science	To design a model that can accurately identify and classify a brain MRI according to severity of AD.	4
3	Design/development of solutions: Design solutions of complex engineering problems, design system components or process that meet the specified process with appropriate consideration for the public health, safety and the cultural and environmental considerations.	To use ML and DL models with different classifiers for training and testing against different datasets.	4
4	Conduct investigations of complex problems: Use research based knowledge and research methods including design experiments, analysis and interpretation of data, and synthesis of information to provide valid conclusions.	The ML and DL models are tested against different datasets to find the variant of the model that gives high accuracy for the considered dataset.	4

5	Modern tool usage: Create, insert and apply appropriate techniques, resources and modern engineering and tools including prediction and modeling to complex engineering activities with an understanding of the limitations.	The project was carried out using Jupyter notebook. GPU is used to run the deep learning models so that the computation becomes faster.	4
6	The Engineer and Society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to professional engineering practice.	The project can help to find the severity of Alzheimer's disease in a person using the brain MRI.	4
7	Environment and Sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.	The DL model that is developed can be used for primary analysis of Alzheimer's disease and based on the severity that is predicted by the model, the user can consult a neurologist for further diagnosis.	4
8	Ethics: Apply ethical principles and commit to professional ethics and norms of the engineering practice.	This project report acknowledges the contributions of research scholars and presents findings on DrillBit performance.	4
9	Individual and Team Work: Function effectively as an individual and as a member or leader in diverse teams, and in multidisciplinary settings.	Equal and active participation is done among the team members.	3

10	Communication: communicate effectively on complex engineering activities with the engineering community and with the society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.	Effective documentation is done using Latex (Overleaf). Plagiarism check is done for the report.	4
11	Project Management and Finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.	Project was executed as per the planned timeline. The project is customizable for other speech disorders.	3
12	Life-long Learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in broadcast context of technological change.	Experienced problems while developing a code which could not be solved by the present knowledge, therefore to overcome this, many other platforms were explored to obtain solutions by different approaches.	3

13	PSO1: Apply the concepts of electronic circuits and systems to analyses and design systems related to Microelectronics, Communication, Signal processing and Embedded systems for solving real world problems	This project explores different spectrograms to analyze speech signals. Data augmentation will be used to enhance a dataset for training deep learning models aimed at feature extraction. The goal is to develop a model for use in medical and other speech-related applications.	4
14	PSO2: To identify problems in the area of communication and embedded systems and provide efficient solutions using modern tools/algorithms working in a team	Signal Processing Toolbox from Matlab was used to generate Time-Frequency representation of an audio file. Deep learning toolbox from Matlab is used to construct CNN architecture for developing speech recognition model.	4

Level	Grade
poor	1
average	2
good	3
vgood	4
excellent	5

DrillBit

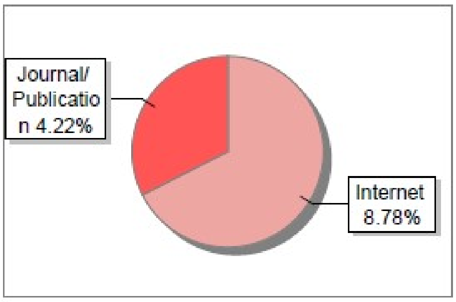
The Report is Generated by DrillBit Plagiarism Detection Software

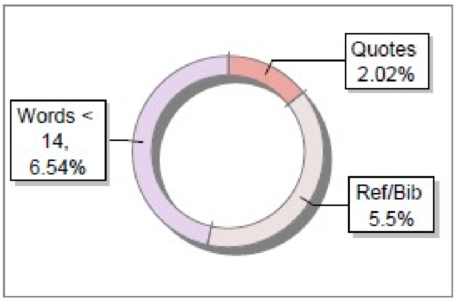
Submission Information

Author Name	aaa
Title	aaa
Paper/Submission ID	1712388
Submitted by	veena.karjigi@sit.ac.in
Submission Date	2024-04-29 10:26:14
Total Pages	52
Document type	Article

Result Information

Similarity **13 %**





Exclude Information

Quotes	Excluded
References/Bibliography	Excluded
Sources: Less than 14 Words %	Excluded
Excluded Source	0 %
Excluded Phrases	Not Excluded

Database Selection

Language	English
Student Papers	No
Journals & publishers	Yes
Internet or Web	Yes
Institution Repository	Yes

A Unique QR Code use to View/Download/Share Pdf File

DrillBit

DrillBit Similarity Report

13	34	B	A-Satisfactory (0-10%) B-Upgrade (11-40%) C-Poor (41-60%) D-Unacceptable (61-100%)
SIMILARITY %	MATCHED SOURCES	GRADE	

LOCATION	MATCHED DOMAIN	%	SOURCE TYPE
1	www.frontiersin.org	2	Internet Data
2	www.mdpi.com	1	Internet Data
3	umpir.ump.edu.my	1	Internet Data
4	www.sit.ac.in	1	Publication
5	www.mdpi.com	1	Internet Data
6	drttit.gvet.edu.in	1	Publication
7	www.frontiersin.org	1	Internet Data
8	www.mdpi.com	<1	Internet Data
9	www.mdpi.com	<1	Internet Data
10	www.mdpi.com	<1	Internet Data
11	www.dx.doi.org	<1	Publication
12	dspace.bracu.ac.bd	<1	Publication
13	www.mdpi.com	<1	Internet Data
14	www.mdpi.com	<1	Internet Data

15	www.ncbi.nlm.nih.gov	<1	Internet Data
16	www.hindawi.com	<1	Internet Data
17	link.springer.com	<1	Internet Data
18	ijrpr.com	<1	Publication
20	www.mdpi.com	<1	Internet Data
21	academic.kongu.edu	<1	Publication
22	information-science-engineering.newhorizoncollegeofengineering.in	<1	Publication
24	www.deskera.com	<1	Internet Data
25	www.dx.doi.org	<1	Publication
26	www.mdpi.com	<1	Internet Data
28	Restorative Qentistry Clinical Applications by Williams-1986	<1	Publication
29	arxiv.org	<1	Publication
30	recentscientific.com	<1	Publication
31	System for Recognition of Facial Expressions Using Machine Learning by Almeid-2020	<1	Publication
32	core.ac.uk	<1	Publication
33	ijrte.org	<1	Publication
34	www.edureka.co	<1	Internet Data
36	qdoc.tips	<1	Internet Data
37	www.mdpi.com	<1	Internet Data