ALZHEIMER’S DISEASE DIAGNOSIS VIA INTUITIONISTIC FUZZY RANDOM VECTOR FUNCTIONAL LINK NETWORK

A PROJECT REPORT

Submitted by

KUMARA BALAJI S – 312319205076
MITHUN B - 312319205091

of

BACHELOR OF TECHNOLOGY
in

INFORMATION TECHNOLOGY

St. JOSEPH’S COLLEGE OF ENGINEERING

(An Autonomous Institution)

St. Joseph’s Group of Institutions

Jeppiaar Educational Trust,

OMR, Chennai-600119

ANNA UNIVERSITY: CHENNAI

March-2023

ANNA UNIVERSITY: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report **"ALZHIMER'S DISEASE**" is the bonafide work of **KUMARA BALAJI S (312319205076)** and **MITHUN B (312319205091)** who carried out the project under my supervision.

SIGNATURE	SIGNATURE
Project Guide,	**Head of the department,**
Ms.K. Deepa,M.E.	**Dr.V. Muthulakshmi, M.E.,Ph.D,**
Assistant Professor,	**Professor,**
Department of IT,	**Department of IT,**
St.Joseph's College of Engineering,	St.Joseph's College of Engineering,
OMR, Chennai- 600119.	OMR, Chennai- 600119.

CERTIFICATE OF EVALUATION

COLLEGE NAME : St. Joseph's College of Engineering, Chennai-600119.

BRANCH : **B.TECH., IT** (Information Technology)

SEMESTER : VIII

SL. NO	NAME OF THE STUDENT	TITLE OF THE PROJECT	NAME OF THE SUPERVISOR WITH DESIGNATION
1	KUMARA BALAJI S (3123192050576)	Alzheimer's Disease Diagnosis via Intuitionistic Fuzzy Random Vector Functional Link Network	Ms.K.DEEPA, M.E., ASSISTANT PROFESSOR
2	MITHUN B (312319205091)		

The report of the project work submitted by the above students in partial fulfillment for the award of Bachelor of Technology Degree in Information Technology of Anna University was confirmed to be report of the work done by the above students and then evaluated.

Submitted to Project and Viva Examination held on______________________.

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

At the outset we would like to express our sincere gratitude to the beloved **Chairman, Dr.Babu Manoharan, M.A.,M.B.A.,Ph.D.,** for his constant guidance and support.

We would like to express our heartfelt thanks to our respected **Managing Director Mrs. S. Jessie Priya, M.Com.,** for her kind encouragement and blessings.

We wish to express our sincere thanks to our **Executive Director Mr. B. Shashi Sekar, M.Sc.,** for providing ample facilities in the institution.

We express our deepest gratitude and thanks to our beloved **Principal Dr.VaddiSeshagiri Rao, M.E., M.B.A., Ph.D., F.I.E.,** for his inspirational ideas during the course of the project.

We wish to express our sincere thanks and gratitude to **Dr.V.Muthulakshmi, M.E.,Ph.D.,** Head of the Department , Department of Information Technology, St. Joseph's College of Engineering for her guidance and assistance in solving the various intricacies involved in the project.

It is with deep sense of gratitude that we acknowledge our indebtness to our supervisor **Ms.K.Deepa, M.E.,** for her expert guidance and connoisseur suggestion.

Finally we thank our department staff members who helped us in the successful completion of this project.

ABSTRACT

Alzheimer's disease is a progressive and debilitating neurological disorder that affects a significant number of people worldwide. Early diagnosis is crucial for effective management and treatment. The proposed model utilizes a computational model called the Intuitionistic Fuzzy Random Vector Functional Link Network (IFRVFLN) to diagnose Alzheimer's disease with high accuracy. The Kaggle dataset used in this project contains brain images(MRI) of patients with Alzheimer's disease and healthy individuals. The images are preprocessed to extract features, which are then used to train the IFRVFLN model. The model's performance is evaluated using several metrics, including sensitivity, specificity, and accuracy. The results show that the IFRVFLN model accurately diagnoses Alzheimer's disease with high sensitivity, specificity, and accuracy. The proposed approach provides a valuable tool for healthcare professionals in diagnosing Alzheimer's disease, improving patient care and management. The CNN used in this project employs multiple convolutional layers, followed by pooling layers and fully connected layers. The use of multiple layers allows the network to learn hierarchical representations of the input images, improving the accuracy of the model. Additionally, techniques such as dropout and batch normalization are utilized to prevent overfitting and improve the generalization performance of the CNN.

The utilization of the Kaggle dataset provides a readily available and accessible resource for future research in the field of Alzheimer's disease diagnosis. Overall, this study contributes to the growing body of literature on computational models for diagnosing Alzheimer's disease and demonstrates the potential of the IFRVFLN model in clinical settings.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1. ALZHEIMER’S DISEASE

The brain is considered one of the most crucial organs in our body. All the activities and responses that allow us to think and believe are controlled and facilitated by the brain. It also empowers our sentiments and recollections. Alzheimer's disease is brain dysfunction which is unrepairable and progressive in nature. Someone in the world is diagnosed with Alzheimer's disease every four seconds. It enhances at a languid pace and tears down the memory cells, thereby destroying an individual's thinking ability. It's a degenerative nerve disorder that leads to loss of function or even death of neurons. The average life expectancy after an Alzheimer's diagnosis is only about four to eight years. On an average, 1 out of 10 people over the age of 65is affected by this condition, but sometimes it can strike at a younger age and has been diagnosed in several people in their 20s. This disease is the primary cause of dementia in older people. Dementia causes a decline in cognitive skills that are used to perform daily activities, 60-80% of dementia cases are Alzheimer's.

1.1 MEDICAL IMAGING

One of the most common and preliminary Alzheimer's disease detection tests, along with psychological examinations, is the brain MRI scanning andanalysis. The medical professionals examine the MRI scans and assess possible factors that have the potential to reveal the presence of Alzheimer's disease, such as, brain matter degeneration, tumor,etc. Although manual examinations of MRI data prove to be effective in detecting the presence of Alzheimer's disease, this process tends to reducethe efficiency of expeditious arrival ofconclusions. This paper proposes an automatic detection of Alzheimer's disease.

1.2 ADVERSARIAL ATTACKS

This work is motivated to enhance the diagnostic capability of physicians and reduce the time required for accurate diagnosis. Image segmentation is performed on the input images. This enables easier analysis of the image thereby leading to better disease detection efficiency.

1.3 SYSTEM OVERVIEW

Segmentation of brain image is imperative in surgical planning and treatment planning in the field of medicine. In this work we have proposed a neural network approach to detect Alzheimer's disease using via convolutional neural network which comes under machine learning algorithm to attain high accuracy**.**

CHAPTER 2
LITERATURE SURVEY

[1] Hiroki Fuse; Kota Oishi; NorihideMaikusa; Tadanori Fukami, Detection of Alzheimer's Disease with Shape Analysis of MRI Images. Japanese Alzheimer's Disease Neuroimaging Initiative,2018 Joint 10th International Conference on Soft Computing and Intelligent Systems (SCIS) and 19th International Symposium on Advanced Intelligent Systems (ISIS)

In the current study, we tested the effectiveness of a method using brain shape information for classification of healthy subjects and Alzheimer's disease patients. A P-type Fourier descriptor was used as shape information, and the lateral ventricle excluding the septum lucidum was analyzed. Using a combination of several descriptors as features, we performed classification using a support vector machine. The results revealed classification accuracy of 87.5%, which was superior to the accuracy achieved using volume ratio to intracranial volume (81.5%), which is widely used for conventional evaluation of morphological changes. The current findings suggest that shape information may be more useful in diagnosis, compared with conventional volume ratio.

[2] Priyanka Thakare; V.R.Pawar, Alzheimer disease detection and tracking of Alzheimer patient. 2016 International Conference on Inventive Computation Technologies (ICICT)

Alzheimer disease is one of the forms of dementia. AD is tremendously increasing disease in the world. There are so many biomarkers detect the Alzheimer disease. From that Electroencephalograph signal is give correct result and performance. In Alzheimer disease, death of brain cells are occurs so there is many causes happened such as memory loss, poor in calculation and recent event happened etc. Early detection of Alzheimer disease is very useful for him and his family. Early detection of Alzheimer patients is very useful for him and his family. In detection, firstly EEG database is filter then noise and artifacts is removed from EEG database using independent component analysis. By wavelet transform four features are extracted and classification is done by support vector machine. In monitoring system, Alzheimer patient is track by using GPS and GSM. With the help of this monitoring system Alzheimer patient is travel anywhere without caregiver.

[3] Aarti Sharma; J. K. Rai; R. P. Tewari, Relative Measures to Characterize EEG Signals for Early Detection of Alzheimer. 2018 5th International Conference on Signal Processing and Integrated Networks (SPIN)

Number of Alzheimer's patients are continuously increasing worldwide. Early detection of Alzheimer will improve patients life and world health care cost. A lot of modalities are there with the help of which Alzheimer can be detected at an early stage. But choosing a potential feature is a challenging task. In the present study we investigate frequency relative energy, frequency relative power and relative entropy from the dataset of Mild Cognitive Impairment (MCI), Normal and Dementia subjects. Bhattacharya distance is used to rank the aforementioned features that can classify between MCI, Control and Dementia subjects. Relative Entropy is identified as the best feature for classification. All the findings are statistically validated using Kruskal-Walistest. The selection of the relevant feature will be beneficial for early Alzheimer detection and may increase the quality of life of the patients suffering from the disease.

[4] A. N. N. P Gunawardena; R. N Rajapakse; N. D Kodikara; I. U. K. Mudalige, Moving from detection to pre-detection of Alzheimer's Disease from MRI data. 2016 Sixteenth International Conference on Advances in ICT for Emerging Regions (ICTer)

Alzheimer's Disease (AD) is the most common form of dementia, affecting approximately 10% of individuals under 65 years of age, with the prevalence doubling every 5 years up to age 80, above which the prevalence exceeds 40%. Currently diagnosis of AD is largely based on the examination of clinical history and tests such as

MMSE (Mini-mental state examination) and PAL (Paired Associates Learning). However many present studies have highlighted the inaccuracies and limitations of such tests. Thus medical officers are now moving to the more accurate neuroimaging data (Magnetic Resonance Imaging- MRI) based diagnosis for these types of diseases where brain atrophy transpires. However it is a considerable challenge to analyse large numbers of images manually to get the most accurate diagnosis at present.

[5] H. M. Tarek Ullah; ZishanAhmedOnik; Riashat Islam., Alzheimer's Disease and Dementia Detection from 3D Brain MRI Data Using Deep Convolutional Neural Networks. Dip Nandi, 2018 3rd International Conference for Convergence in Technology (I2CT)

As reported by the the Alzheimer's Association, there are more than 5 million Americans living with Alzheimer's today, with an anticipated 16 million by 2050. The neurodegenerative disease is currently the 6th leading source of death in the US. In 2017 this disease would cost the nation $1.1 trillion. 1 in 3 seniors die in Alzheimer's disease or another dementia. It kills more than breast cancer and prostate cancer combined. As of the this papers writing, detecting Alzheimer's is a difficult and time consuming task, but requires brain imaging report and human expertise. Needless to say, this conventional approach to detect Alzheimer's is costly and often error prone. In this paper an alternative approach has been discussed, that is fast, costs less and more reliable. Deep Learning represents the true bleeding edge of Machine Intelligence. Convolutional Neural Networks are biologically inspired Multilayer perceptron specially capable of image processing. In this paper we present a state of the art Deep Convolutional Neural Network to detect Alzheimer's Disease and Dementia from 3D MRI image.

[6]M. K. Hasan et al., "Lightweight Encryption Technique to Enhance Medical Image Security on Internet of Medical Things Applications," in IEEE Access, vol. 9, pp. 47731-47742, 2021.

M. K. Hasan et al presented a lightweight encryption algorithm to develop a secure image encryption technique for the healthcare industry. The lightweight encryption technique employed two permutation techniques to secure medical images which was analyzed, evaluated, and then compared to conventionally encrypted ones in security and execution time. Also, the study suggested that the existing proposed techniques still face application-specific security concerns.

[7]Quist-Aphetsi Kester, Laurent Nana, Anca Christine Pascu, Sophie Gire, Jojo M. Eghan, NiiNarkuQuaynor, A Cryptographic Technique for Security of Medical Images in Health Information Systems, ProcediaComputer Science,2015.

The original plain image data in the cryptographic and watermarking methods should be fully recoverable due to the sensitivity of data conveyed in medical images. In satisfying the recoverability and the authentication process, a fully recoverable encrypted and watermarked image technique for the security of medical images in health information systems is suggested by Quist-Aphetsi Kester et al., which was used to authenticate and secure the medical images in health information systems that showed to be very effectiveand reliable for fully recoverable images.

[8]Xiaohui Wang, Zheng Zhu, Fan Wang, Renjie Ni, Jun Wang, and Yuhen Hu, "Medical image encryption based on biometric keys and lower–upper decomposition with partial pivoting," Appl. Opt. 60, 24-32 (2021).

An asymmetric medical image encryption scheme was proposed by Xiaohui Wang et al,. where the medical image was encrypted by two spiral phase masks (SPM) and the lower–upper decomposition with partial pivoting, where the SPM was generated from the iris, chaotic random phase mask, and amplitude truncated spiral phase transformation. It was revealed that the iris used for medical image encryption, improved the security of the encryption scheme and the combination of asymmetric optical encryption and three dimensional Lorenz chaos improves the key space and solved the linear problem based on double- random phase encoding.

[9]Selvi, C.T., Amudha, J. & Sudhakar, R. Medical image encryption and compression by adaptive sigma filterizedsynorr certificateless signcryptive Levenshtein entropy-coding-based deep neural learning. Multimedia Systems, 2021.

An adaptive sigma filterizedsynorr certificateless signcryptiveLevenshtein entropy coding-based deep neural learning (ASFSCSLEC-DNL) technique was presented to develop the image encryption and compression by Selvi, C.T et al.,. The adaptive sigma filter was employed to denoise the medical image. The medical image encryption and signature generation were done with synorr certificateless signcryption. Finally, Levenshtein entropy encoding was applied to compress images which uncovered certain edge cases.

[10]Singh, Gurinder & Deep, Gaurav, MRI Medical Image and Steganography, Research gate , 2016.

The working and functions of Magnetic Resonance was suggested by Singh et al.,. The medical records contain extremely sensitive information about the patient which should not be compromised where image steganography was used to hide the sensitive patient information into the carrier medical image to ensure confidentiality. Using steganography, the overhead of paperwork was reduced to record the patient information because this technique embedded the information in image. Various steganography techniques was reviewed which were previously used in medical field.

[11]M. Jain, R. C. Choudhary and A. Kumar, "Secure medical image steganography with RSA cryptography using decision tree," 2016 2nd International Conference on Contemporary Computing and Informatics (IC3I), 2016.

A conventional methodology about secure medical information transmission of patient inside medical cover image was presented by M. Jain et al, by concealing data using decision tree concept. Decision tree showed a robust mechanism by providing decisions for secret information concealing location in medical carrier image using secret information mapping concept and RSA encryption algorithm was used for patient's unique information enciphering. Receiver got to receive the hidden secret medical information of patient using RSA decryption, so only authorized recipient can recognize the text.

CHAPTER 3

SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

Existing made use of a 5 stage ML pipeline process for the detectionin which each stage had a sub-stage. Multiple classifiers wereapplied to this pipeline. He concluded that the random forestClassifier had better performance metrics.made use of the Random-Forest classifier to compare the performance in imputation and non-imputationmethods. They observed that the imputation method gives 87%accuracy, and the non-imputation method gives 83% accuracy.It further classified the subjects as demented or non-demented,respectively.

3.1.1 Disadvantages of Existing System

Traditional CAD systems necessitate training before pre-taught algorithms may be used. Most importantly, in the recent times, there has been a need to share the medical data online to doctors or medical practitioners to enable remote consultation. In such a scenario, medical images (reports, x-rays, scans) that are shared online are not secure and prone to perturbations that may cause critical issues such as false or misinterpretation of reports which will even lead to falsified treatment and even death of a patient. Studies reveal that machine learning and AI are highly unstable in medical image reconstruction and may lead to false positives and false negatives. Thus, there exists a serious need to securely transmit any type of medical information especially medical images over internet to ensure better care and patient treatment.

3.2 PROPOSED SYSTEM

The proposed model intends to explicitly fabricate adversarial attacks on medical images, reveal the obfuscated images and thereby suggest efficient encryption techniques and employ image hiding in plain sight using deep steganography, thus helps to achieve secure transfer of medical images and accurate proficient analysis of images for better interpretation. The brain MRI images considered here reveal the brain tumor MRI that included two major classes – with brain tumor(yes) which has 150 images and without brain tumor(no) which has 100 images. Since the data is minimal, GAN based image augmentation is done to scale the MRI dataset.

The cnn model has shown decent accuracy in classification of images. The model has displayed certain promising graphs.It used a batch size of 64. The model has runned for 10 epochs. The model provided an accuracy of about 95% in the train data and about 80% in the test data. The proposed model can help doctors diagnose Alzheimer's Disease more effectively and can be modified

A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other. The pre-processing required in a ConvNet is much lower as compared to other classification algorithms. A ConvNet is able to successfully capture the Spatial and Temporal dependencies in an image through the application of relevant filters. The architecture performs a better fitting to the image dataset due to the reduction in the number of parameters involved and reusability of weights.The objective of the

Convolution Operation is to extract the high-level features such as edges, from the input image. ConvNets need not be limited to only one Convolutional Layer. Conventionally, the first ConvLayer is responsible for capturing the Low-Level features such as edges, color, gradient orientation, etc.

3.2.1 Advantages of Proposed System

- High accuracy
- Classification methods done well

3.3 REQUIREMENT SPECIFICATION

3.3.1 Hardware Requirements

- Hard Disk: 40GB and above
- RAM: 512MB and above
- Processor: Pentium i7 and above

3.3.2 Software Requirements

- Windows operating system XP and above
- Google Colaboratory
- Python 3.6 and above
- Visual Studio Code/Jupyter notebook

3.4 LANGUAGE/LIBRARY SPECIFICATION

Python is a high-level, interpreted, interactive and objectoriented scripting language. Python is designed to be highly readable. When programming, computer vision representations of vectors and matrices and operations on them is highly required. This is handled by Python's NumPy module where both vectors and matrices are represented by the array type. This is also the representation for images. For visualizing results Matplotlib module and for more advanced mathematics is being used.

OpenCV is a cross-platform library using which developers can develop real-time computer vision applications. It mainly focuses on image processing, video capture and analysis including features like face detection and object detection. Computer Vision can be defined as a discipline that explains how to reconstruct, interrupt, and understand a 3D scene from its 2D images, in terms of the properties of the structure present in the scene. It deals with modeling and replicating human vision using computer software.

TensorFlow Serving is a flexible, high-performance serving system for machine learning models, designed for production environments. TensorFlow Serving makes it easy to deploy new algorithms and experiments, while keeping the same server architecture and APIs. TensorFlow Serving provides out-of-the-box integration with TensorFlow models, but can easily extended to serve other types of models and data. TensorFlow is an open source software library for high performance numerical computation. Its flexible architecture allows easy deployment of computation across a variety of platforms (CPUs, GPUs, TPUs), and from desktops to clusters of servers to mobile and edge devices.

CHAPTER 4

SYSTEM IMPLEMENTATION

4.1 SYSTEM ARCHITECTURE

The proposed model intends to explicitly fabricate adversarial attacks on medical images, reveal the obfuscated images and thereby suggest efficient encryption techniques and employ image hiding in plain sight using deep steganography, thus helps to achieve secure transfer of medical images and accurate proficient analysis of images for better interpretation.

The brain MRI image dataset considered here reveal the brain tumor MRI that included two major classes – with brain tumor(yes) which has 150 images and without brain tumor(no) which has 100 images. Since the data is minimal, GAN based image augmentation is done to scale the MRI dataset and a set of pre- processing for images is carried out. The adversarial attacks are explicitly carried out in the fabricated dataset. Initially, a series of geometrical attacks such as scaling, rotation, 2D convolution, gaussian blur are been carried out using a single python script wherein the outputs are saved in the appropriate folders. Followed by which, a Projected Gradient Descent (PGD) attack is established. PGD is a white-box attack as the attacker has access to the model gradients. This threat model gives the attacker far more power than black box attacks since they can tailor their attack to mislead the model instead of relying on transfer attacks, which frequently cause human-visible perturbations.

The final implementation of attack is the DAG (Dense Adversarial Generation) attack and where DAG is a semantic segmentation DNN adversarial attack using the base U-NET model. While deceiving a stateof-the-art segmentation DNN, and the attacks generates an adversarial image against a target that closely matches and the genuine image. The difference in adversarial images is revealed and the perturbations are shown evidently thus taking forward to implement secure cryptographic techniques to ensure secure transfer of images. This involves implementing image encryption using various chaos maps and comparing their merits based on key sensitivity, adjacent pixel autocorrelation and intensity histograms.

The chaos maps implemented were - Arnold cat maps, Henon maps and Logistic chaos maps. Arnold's Cat Maps are chaotic twodimensional maps that can be used to shift the position of a pixel in an image without deleting any data. The Henon map is a discrete time dynamic system introduced by michelhenon and the map depends on twoparameters, a and b, which for the classical Henon map have values of a = 1.4 and b = 0.3. These classical values make the map chaotic. The logistic map is a two-dimensional polynomial mapping that is frequently referenced as a classic example of how complex, chaotic behaviour can emerge from extremely simple non-linear dynamical equations.

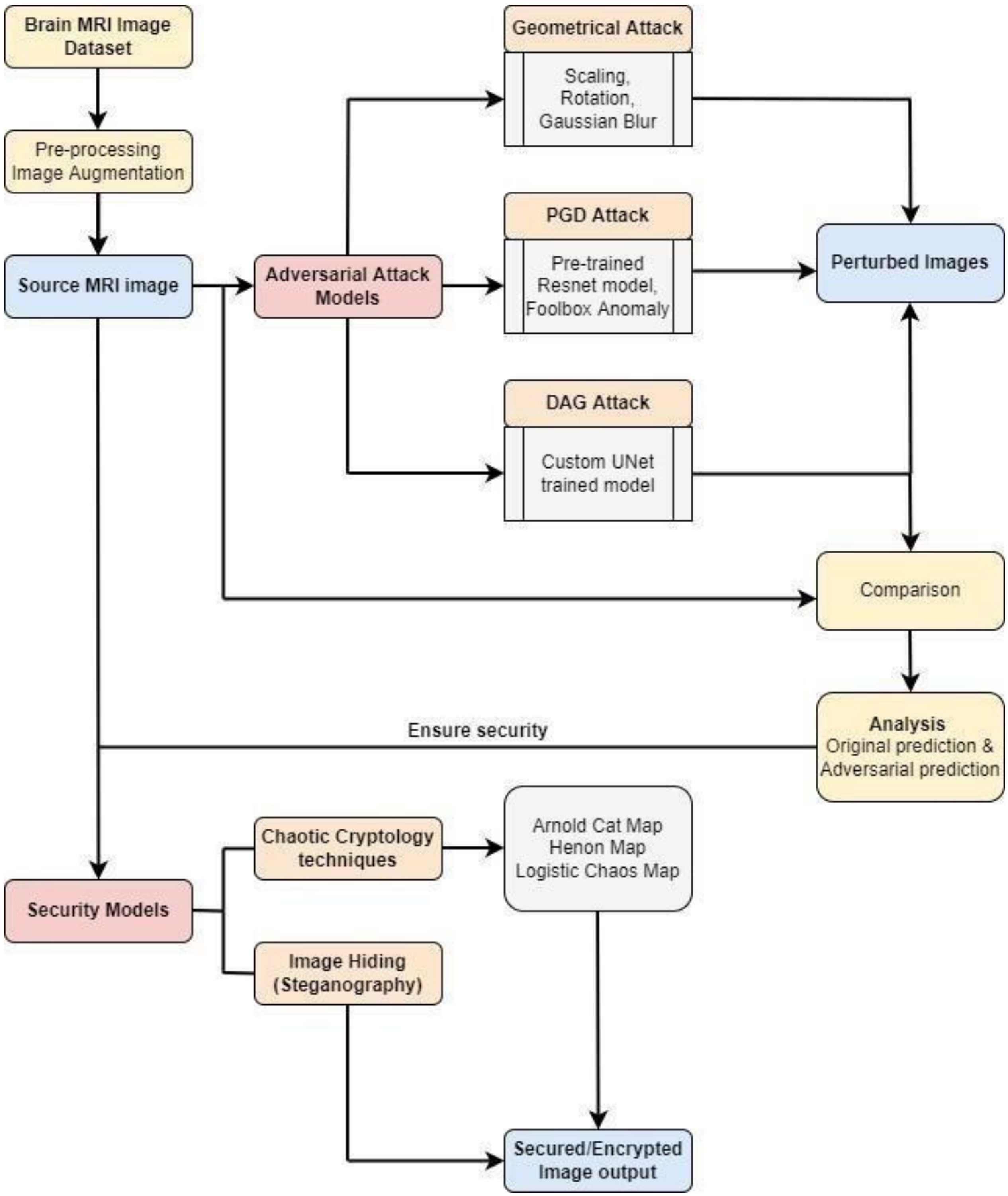


Figure 4.1 Architecture of the proposed system

Image hiding in plain sight (Deep steganography) is established using a set of source images (from Brain MRI dataset) and cover images (oxford flower dataset). The system is trained on images drawn from the oxford flower database, and works well on medical images from the MRI dataset. Beyond demonstrating the successful application of deep learning to hiding images, we carefully examine how the result is achieved and explore extensions. A hidden image and reveal image is produced as the output of the model trained that helps to securely transfer images or any type of data. A simple web application is developed where the encryption and decryption of images is implemented using a simple mathematical logic. It requires two things, image data and key and when XOR operation is applied on both the operands i.e data and key, the data gets encrypted but when the same process is done again with same key value, data gets decrypted.

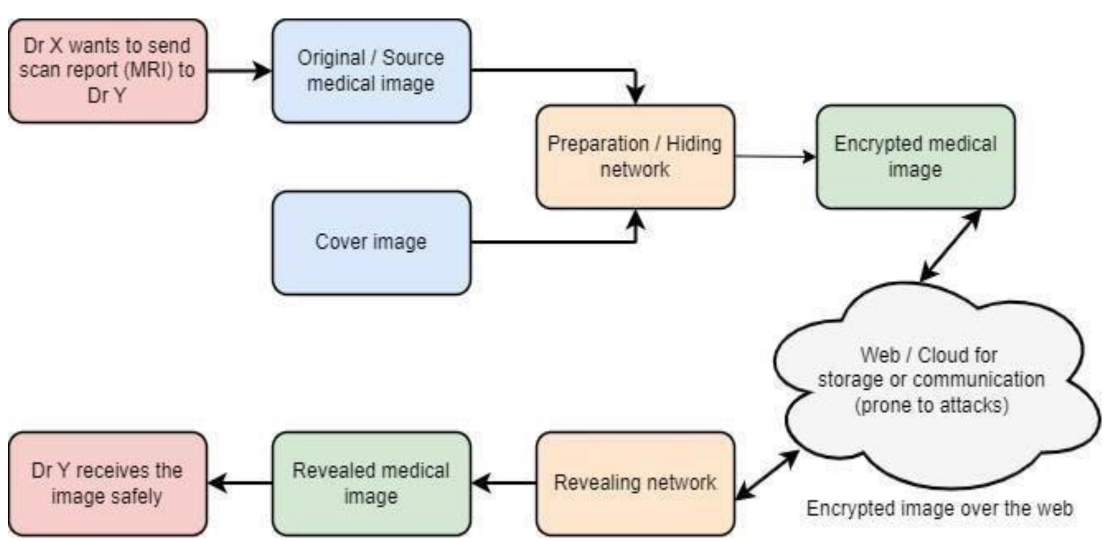


Figure 4.1.1 Use-case scenario diagram

Any two medical professionals or a doctor and a patient who wishes to exchange medical images safely might use a secure network to encrypt and decrypt their image data. Dr. X who wishes to safely send medical images to Dr. Y. Initially Dr X has the original medical image to be shared. It goes to the Preparation or Hiding Network where the source image and a cover image is consumed and an encrypted medical Image is outputted. The image is shared via cloud or World Wide Web where illegal or perturbation networks can disturb sensitive data. Thus, the encrypted image is sent over the web and the revealing network is defined to revealed the true medical image. Thus, Dr. Y who is ensured to receive the medical images reports safely.

4.2 USECASE DIAGRAM

A use case diagram at its simplest is a representation of a user's interaction with the system and depicting the specifications of a use case . A use case diagram can portray the different types of users of a system and the various ways that they interact with the system. The purposes of use case diagrams can be as defined as : Used to gather requirements of a system, Used to get an outside view of a system, Identify external and internal factors influencing the system, Show the interacting among the requirements are actors. UML use case diagrams are ideal for representing the goals of systemuser interactions, defining and organizing functional requirements in a system, specifying the context and requirements of a system, modeling the basic flow of events in a use case. The primary actors involved in the system are sender and receiver.

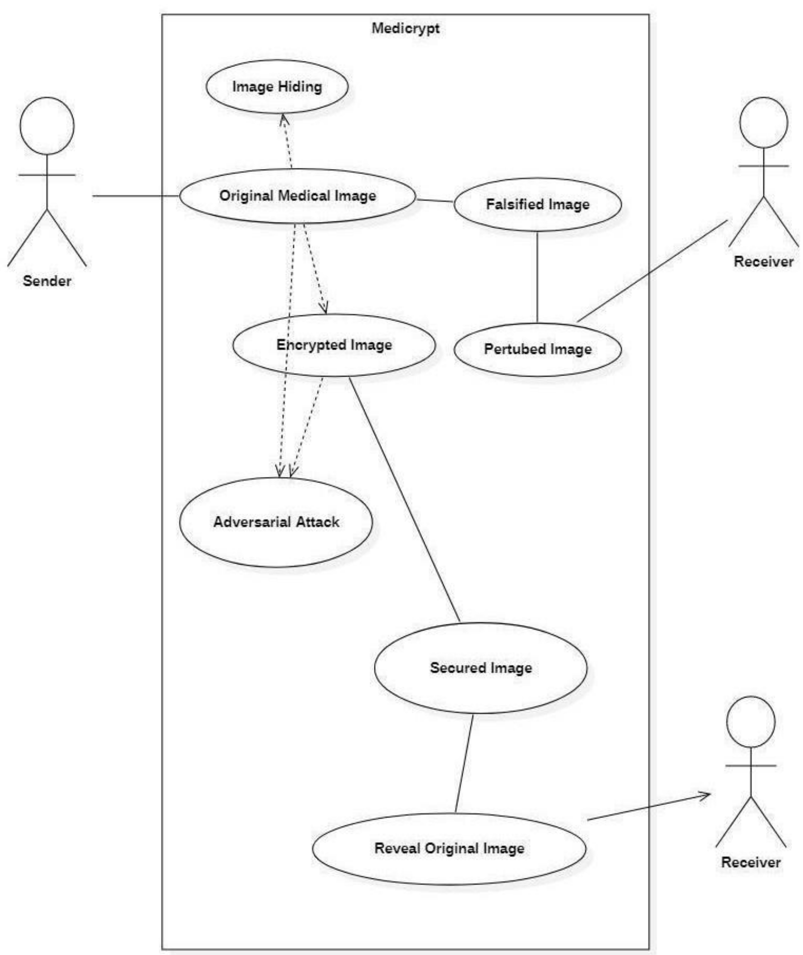


Figure 4.2 Usecase diagram

4.3 COMPONENT DIAGRAM

Component diagram is a special kind of diagram in UML. The purpose is also different from all other diagrams. It does not describe the functionality of the system but it describes the components used to make those functionalities. Purpose of the component diagram can be summarized as to visualize the components of a system, construct executables by using forward and reverse engineering, describe the organization and relationships of the components. The purpose of a component diagram is to show the relationship between different components in a system. For the purpose of UML 2.0, the term "component" refers to a module of classes that represent independent systems or subsystems with the ability to interface with the rest of the system. component diagrams show the structure of the software system, which describes the software components, their interfaces, and their dependencies.

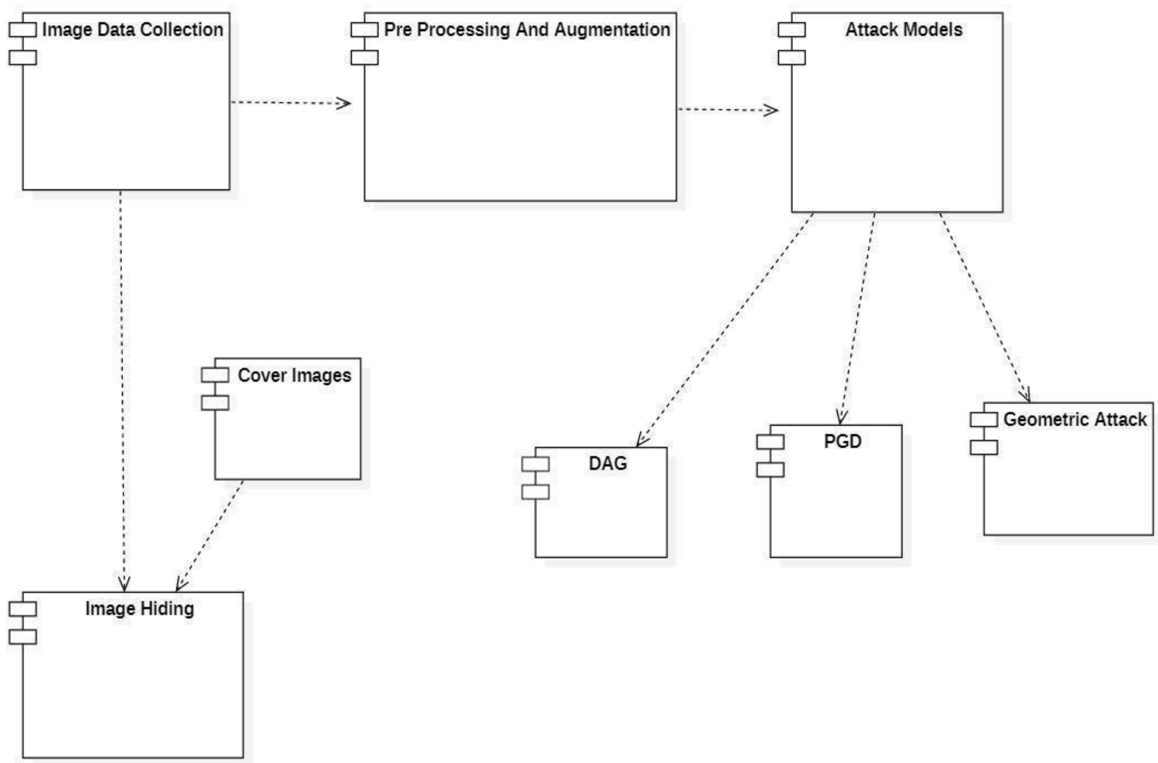


Figure 4.3 Component Diagram

4.4 DEPLOYMENT DIAGRAM

A deployment diagram in the Unified Modeling Language models the physical deployment of artifacts on nodes. To describe a web site, for example, a deployment diagram would show what hardware components exist what software components run on each node and how the different pieces are connected. The nodes appear as boxes, and the artifact appear as rectangles within the boxes. Nodes may have sub-nodes, which appear as nested boxes. A single node in a deployment diagram may conceptually represent multiple physical nodes, such as a cluster of database servers. Deployment diagrams are used to visualize the topology of the physical components of a system where the software components are deployed. So deployment diagrams are used to describe the static deployment view of a system. Deployment diagrams consist of nodes and their relationships. The purpose of deployment diagrams can be described as: visualize hardware topology of a system, describe the hardware components used to deploy software components, describe runtime processing nodes.

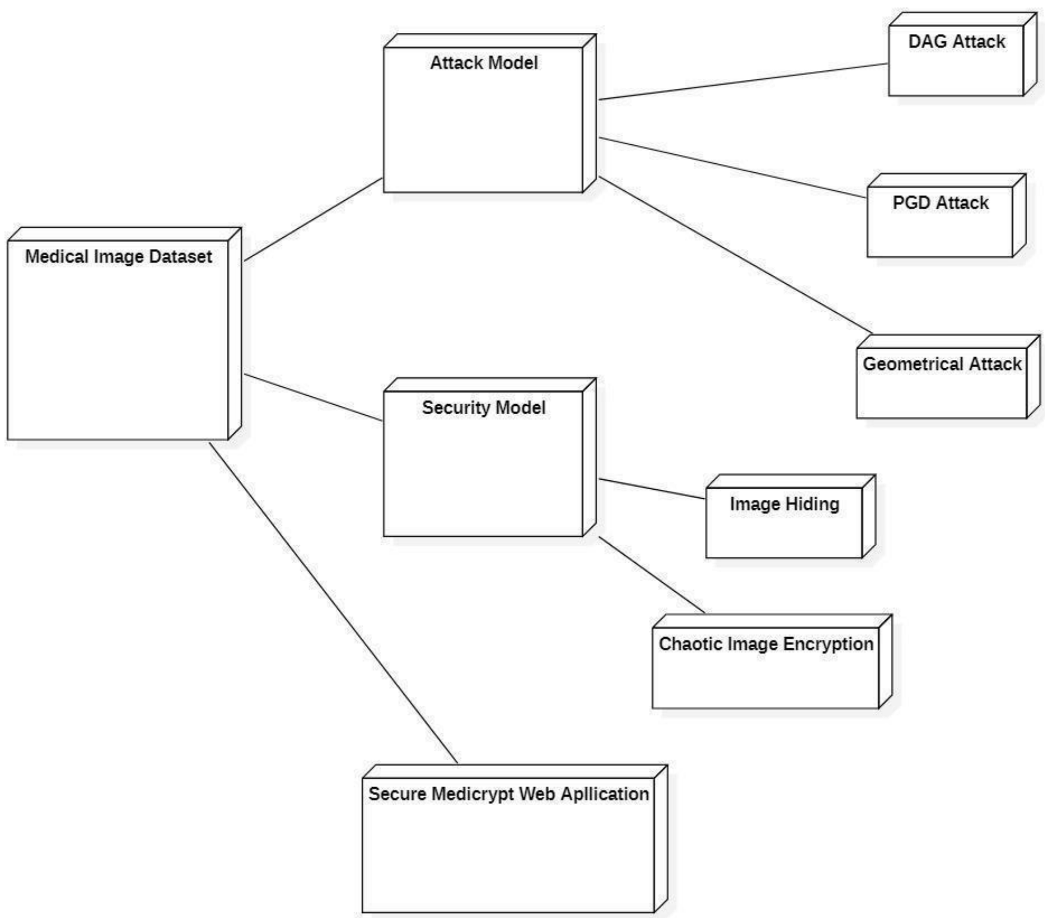


Figure 4.4 Deployment diagram

4.5 SEQUENCE DIAGRAM

A sequence diagram is an interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are used to represent message flow from one object to another object. Sequence diagrams are easierto maintain. Sequence diagrams are easier to generate. Sequence diagrams can be easily updated according to the changes within a system.

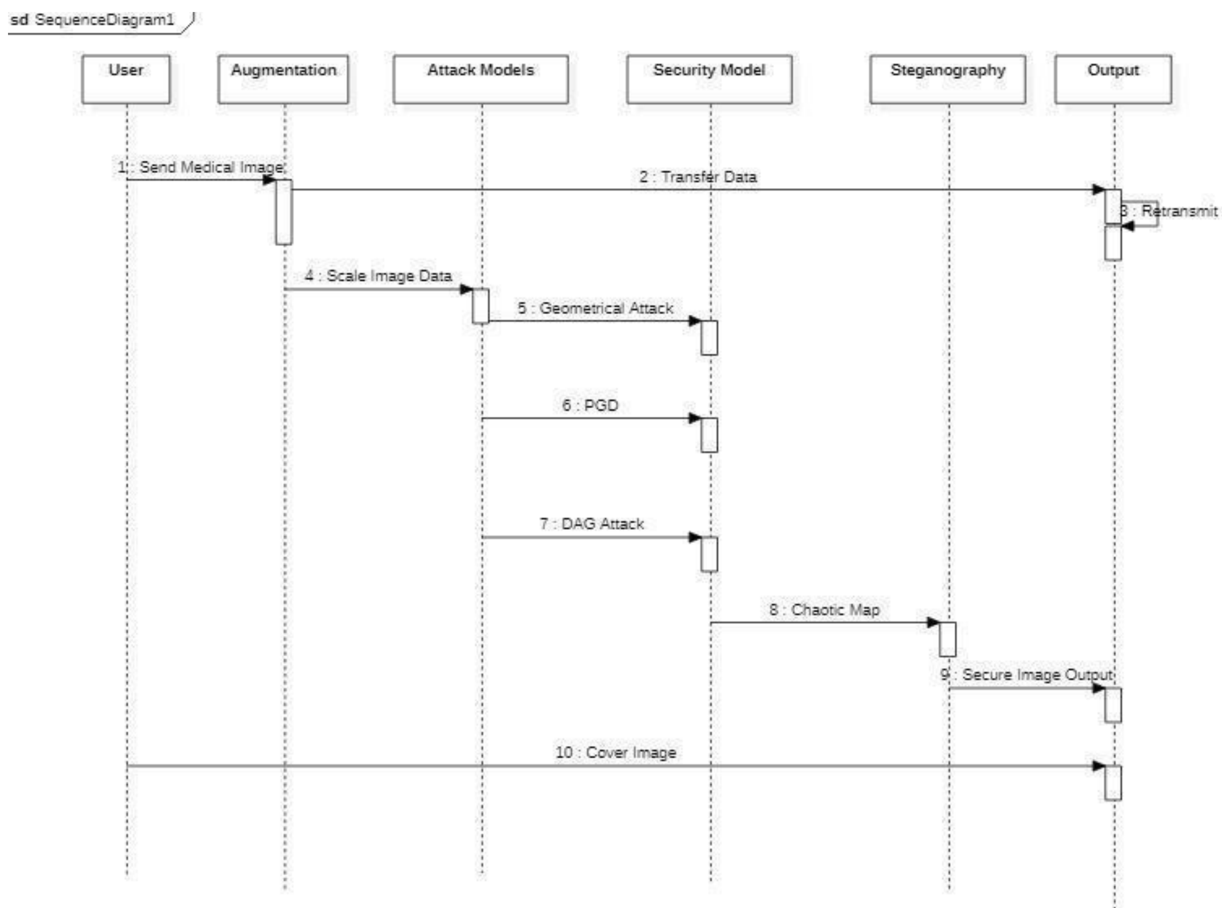


Figure 4.5 Sequence diagram

4.6 STATE DIAGRAM

A state diagram is a type of diagram used in computer science and related fields to describe the behavior of systems. State diagrams require that the system described is composed of a finite number of states. State diagrams are used to give an abstract description of the behavior of a system. This behavior is analyzed and represented by a series of events that can occur in one or more possible states. A state diagram is used to represent the condition of the system or part of the system at finite instances of time. It's a behavioral diagram and it represents the behavior using finite state transitions. State diagrams are also referred to as State machines and State-chart Diagrams.

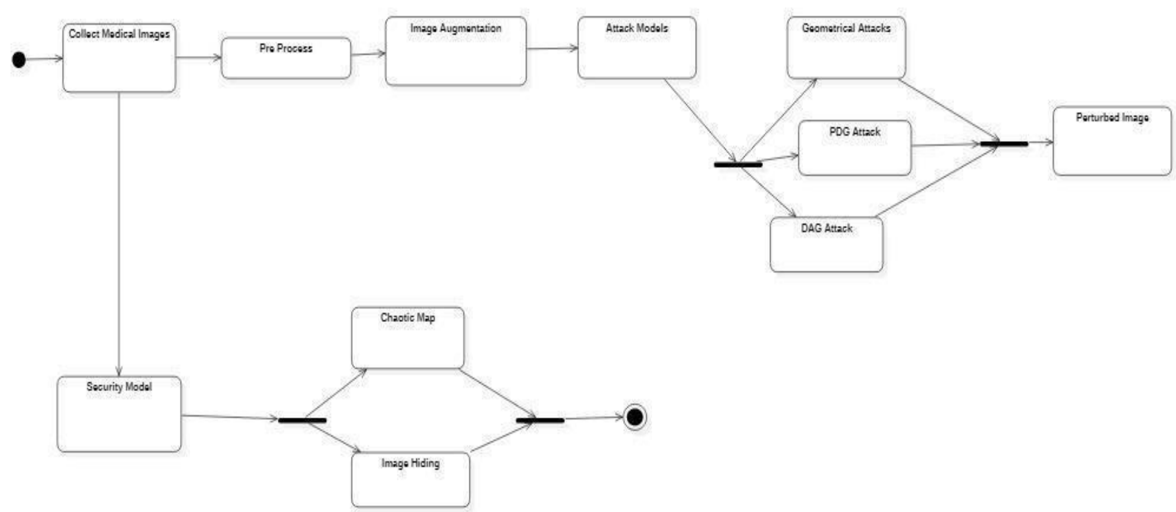


Figure 4.6 State diagram

4.7 CLASS DIAGRAM

Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application. Class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The class diagrams are widely used in the modeling of object oriented systems because they are the only UML diagrams, which can be mapped directly with object-oriented languages. Class diagram shows a collection of classes, interfaces, associations, collaborations, and constraints. It is also known as a structural diagram. The purpose of class diagram is to model the static view of an application. Class diagrams are the only diagrams which can be directly mapped with object-oriented languages and thus widely used at the time of construction.

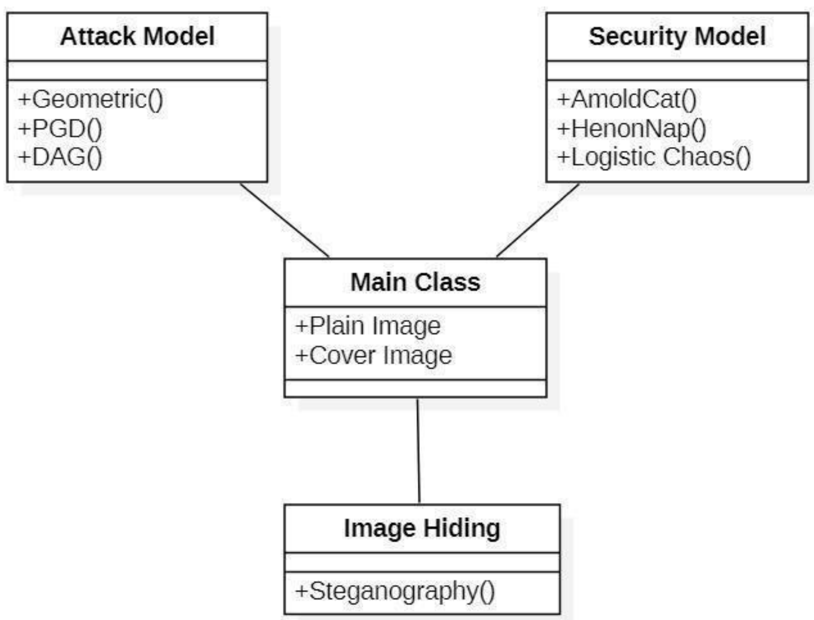


Figure 4.7 Class diagram

4.8 ACTIVITY DIAGRAM

An activity diagram visually presents a series of actions or flow of control in a system similar to a flowchart or a data flow diagram. Activity diagrams are often used in business process modeling. They can also describe the steps in a use case diagram. Actions have incoming and outgoing activity edges that specify the flow of control and data to and from other activity nodes. The actions in an activity start when all of the input conditions are met. One can add input pins and output pins to specify values that are passed to and from the action when it starts.

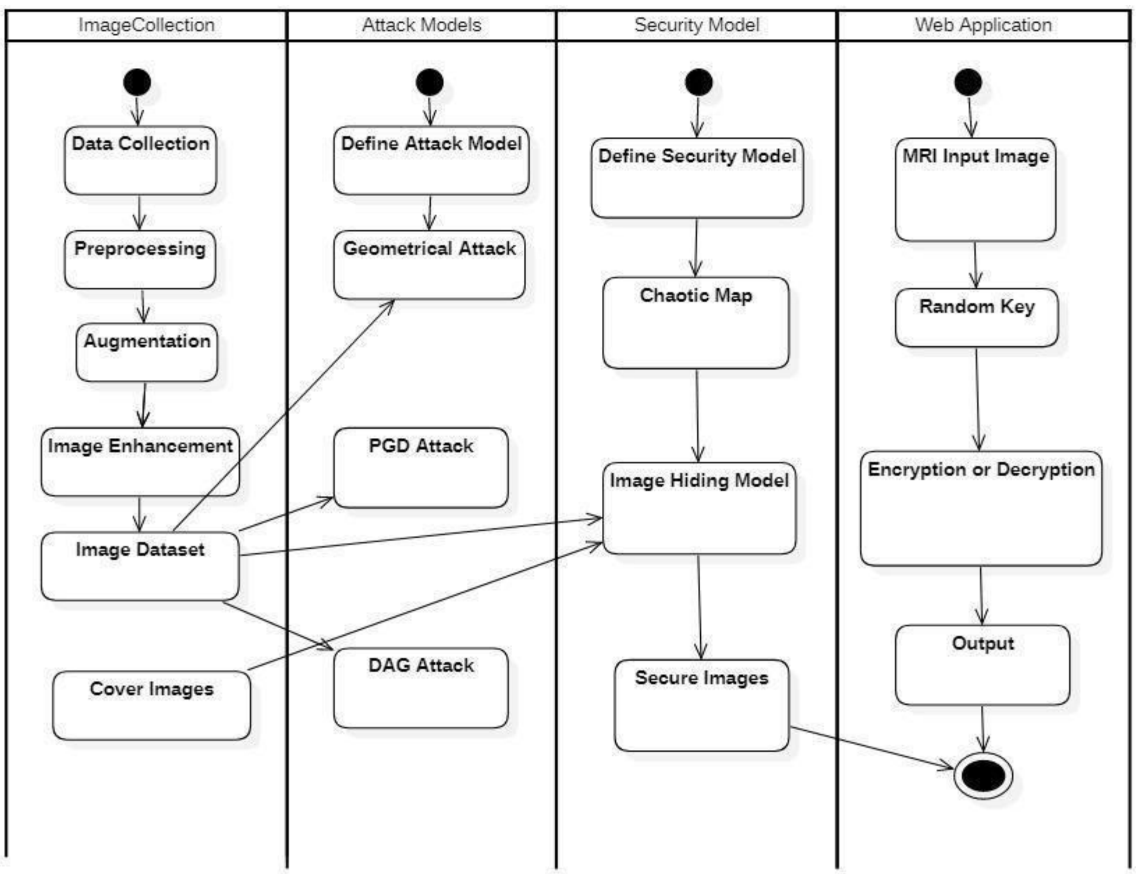


Figure 4.8 Activity diagram

CHAPTER 5

MODULE DESCRIPTION

5.1 MODULES

- Data collection
- Data preprocessing
- feature extaction
- Model creation
- Prediction

5.1.1 DATASET COLLECTION AND PRE-PROCESSING

The data is taken from an open online dataset library known as Kaggle, and the dataset hasn't been used by various other research projects and studies yet. It is an open-source dataset. This dataset contains almost 6,000 images distributed over four classes labelled Mildly, Moderately, Very Mildly and Non-Demented.

Pre-Processing

Preprocessing is a technique that is used to convert the raw data into a clean data set. In other words, whenever the data is gathered from different sources it is collected in raw format which is not feasible for the analysis.

- Image enhancement
- Input image
- Resize
- RGB to grey scale conversion

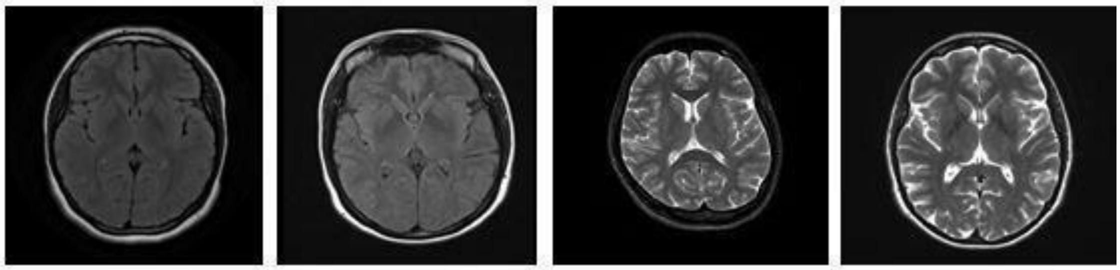


Figure 5.1 Sample images from dataset

5.1.2 MODEL CREATION:

This dataset contains almost 6,000 images distributed over four classes labelled Mildly, Moderately, Very Mildly and Non-Demented. The features are then distributed into 80% train dataset and 20% test dataset. 80% of the data is used in training means that each deep learning model has two phases that are training and testing, where it predicts the data that is provided to it. Both the models use the same dataset separated from the original Kaggle dataset and are divided in 8:2 ratio, which has 80% training and 20% validation dataset. The datasets have to have the same kind of distribution so that there is no such kind of discrepancy in the comparison of the prediction that both the models had a different type of input

5.1.3 PREDICTION:

Python predict() function enables us to predict the labels of the data values on the basis of the trained model.It returns the labels of the data passed as argument based upon the learned or trained data obtained from the model.

predict() : given a trained model, predict the label of a new set of data. This method accepts one argument, the new data X_new (e.g. model. predict(X_new)), and returns the learned label for each object in the array.

5.1.4 CONVOLUTION LAYERS

There are three types of layers that make up the CNN which are the convolutional layers, pooling layers, and fully-connected (FC) layers. When these layers are stacked, a CNN architecture will be formed. In addition to these three layers, thereare two more important parameters which are the dropout layer and the activation function which are defined below.

5.1.4.1 CONVOLUTION LAYER

This layer is the first layer that is used to extract the various features from the input images. In this layer, the mathematical operation of convolution is performed between the input image and a filter of a particular size MxM. By sliding the filter over the input image, the dot product is taken between the filter and the parts of the input image with respect to the size of the filter (MxM).

The output is termed as the Feature map which gives us information about the image such as the corners and edges. Later, this feature map is fed to other layers to learn several other features of the input image.

5.1.4.2 POOLING LAYER

In most cases, a Convolutional Layer is followed by a Pooling Layer. The primary aim of this layer is to decrease the size of the convolved feature map to reduce the computational costs. This is performed by decreasing the connections between layers and independently operates on each feature map. Depending upon method used, there are several types of Pooling operations.

In Max Pooling, the largest element is taken from feature map. Average Pooling calculates the average of the elements in a predefined sized Image section. The total sum of the elements in the predefined section is computed in Sum Pooling. The Pooling Layer usually serves as a bridge between the Convolutional and FC layer.

5.1.4.3 FULLY CONNECTED LAYER

The Fully Connected (FC) layer consists of the weights and biases along with the neurons and is used to connect the neurons between two different layers. These layers are usually placed before the output layer and form the last few layers of a CNN Architecture.

In this, the input image from the previous layers are flattened and fed to the FC layer. The flattened vector then undergoes few more FC layers where the mathematical functions operations usually take place. In this stage, the classification process begins to take place.

5.1.4.4 DROPOUT

Usually, when all the features are connected to the FC layer, it can cause overfitting in the training dataset. Overfitting occurs when a particular model works so well on the training data causing a negative impact in the model's performance when used on a new data.

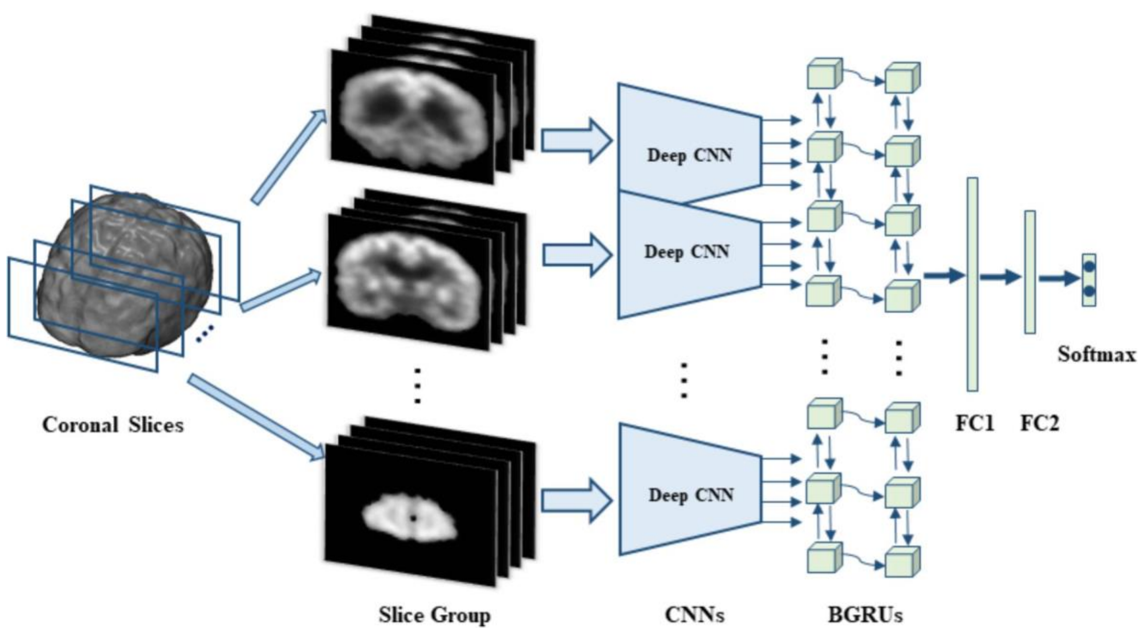
To overcome this problem, a dropout layer is utilised wherein a few neurons are dropped from the neural network during training process resulting in reduced size of the model. On passing a dropout of 0.3, 30% of the nodes are dropped out randomly from the neural network.

5.1.4.5 ACTIVATION FUNCTIONS

Finally, one of the most important parameters of the CNN model is the activation function. They are used to learn and approximate any kind of continuous and complex relationship between variables of the network. In simple words, it decides which information of the model should fire in the forward direction and which ones should not at the end of the network. It adds non-linearity to the network. There are several commonly used activation functions such as the ReLU, Softmax, tanH and the Sigmoid functions. Each of these functions have a specific usage. For a binary classification CNN model, sigmoid and softmax functions are preferred an for a multi-class classification, generally softmax used.

5.1.4.6 DENSE LAYER.

The Dense layers are the ones that are mostly used for the output layers. The activation used is the 'Softmax' which gives a probability for each class and they sum up totally to the model will make it's prediction based on the class with highest probability



CHAPTER 6

CONCLUSION

6.1 CONCLUSION

Medical imaging is developing rapidly due to developments in image processing techniques including image recognition, analysis, and enhancement. Thereby, the system intentionally fabricates adversarial attacks on medical images, reveal the obfuscated images and thereby impose multiple efficient encryption techniques and also employ image hiding in plain sight using deep steganography, which helps to achieve secure transfer of medical images and accurate proficient analysis of images for better interpretation. More focus is made on the enhancement process get clear image using various enhancement to remove the noise. These techniques enhance the lower & higher contrast area of an image in both spatial & frequency domain.

6.1.1 FUTURE ENHANCEMENT

As a part of future work, the proposed model can be enhanced further by increasing the size of image dataset to provide much more stable medical image outputs that are sensitive to contrast, sensitivity and intensity. Further enhancements such as employing attack models using with better hyper parameters, improved image pre-processing techniques will be implemented to cater better secure image models. Real time deployment model betweens sender and receiver can be established and deployed in medical scenarios.

APPENDIX 1

CODING

App.py:

```
from flask import Flask,url_for,render_template,redirect,request#
import sqlite3 as SQL
app = Flask(___name__)
# importtensorflow as tf
from tensorflow import keras
import cv2
# import matplotlib.pyplot as plt
import numpy as np
# import matplotlib.image as mpimg
import os
# from keras.applications.vgg16 import preprocess_input,VGG16
global graph
UPLOAD_FOLDER = 'static/uploader/'
app.config['UPLOAD_FOLDER'] = UPLOAD_FOLDER
SIZE = 120
@app.route('/')
def home():
    return render_template("index.html")
@app.route('/upload',methods =['POST','GET'] )def
```

```
Upload():
    if request.method == 'POST':
        file = request.files['image']
        print(file)
        file.save(os.path.join(app.config['UPLOAD_FOLDER'],'1.png'))
            model = keras.models.load_model(r'D:\project ml\alzheimer-stage-
classifier-master\model\model2.h5')
         categories = ["NonDemented", "MildDemented", "ModerateDemented",
"VeryMildDemented"]
              nimage = cv2.imread(r"D:\project ml\alzheimer-stage-classifier-
master\static\uploader\1.png", cv2.IMREAD_GRAYSCALE)
        image = cv2.resize(nimage,(SIZE,SIZE))
        image = image/255.0
        prediction = model.predict(np.array(image).reshape(-1,SIZE,SIZE,1))
        pclass = np.argmax(prediction)
        pValue = "Predict: {0}".format(categories[int(pclass)])print(pValue)
        realvalue = "Real Value 1"
        print('success')
          img = "/uploader/1.png"
        return render_template('result.html',value=pValue)return

    render_template('index.html')

if___name___== "___main__":
    app.run(debug=True)
```

Model creation

```
from tensorflow import keras


from tensorflow.keras import layers
import os

def createModel(train_data=None):
    if os.path.exists('./model/model.h5') and train_data is None:try:
            print(__name__)
            model = keras.models.load_model('./model/model.h5')
            print("returned")
            return model
        except Exception as e:
            print("error")

       eliftrain_data is not None:
        model = keras.Sequential([
            keras.Input(shape=train_data.shape[1:]), layers.Conv2D(64,
            kernel_size=(3, 3), activation="relu"),
            layers.MaxPooling2D(pool_size=(2, 2)), layers.Conv2D(64,
            kernel_size=(3, 3), activation="relu"),
            layers.MaxPooling2D(pool_size=(2,  2)),
            layers.Flatten(),
            layers.Dense(128, activation="relu"),

            layers.Dropout(0.5),
            layers.Dense(4, activation="softmax")

        ])
        return model
```

Training

```
import numpy as np
from tensorflow import keras

import matplotlib.pyplot as plt
import os
import cv2
import random
import sklearn.model_selection as model_selection
import datetime
from model import createModel
from contextlib import redirect_stdout
categories = ["NonDemented", "MildDemented", "ModerateDemented", "VeryMildDemented"]
SIZE = 120
def getData():
    rawdata = []
    data = []
    dir = "./data/"
    for category in categories:
        path = os.path.join(dir, category)
        class_num = categories.index(category)for
        img in os.listdir(path):
            try:
                        rawdata = cv2.imread(os.path.join(path, img),
cv2.IMREAD_GRAYSCALE)
```

```
            new_data = cv2.resize(rawdata, (SIZE, SIZE))

            data.append([new_data, class_num])
        except Exception as e:
            pass
    random.shuffle(data)

    img_data = []
    img_labels = []
    for features, label in data:
        img_data.append(features)
        img_labels.append(label)
    img_data = np.array(img_data).reshape(-1, SIZE, SIZE, 1)
    img_data = img_data / 255.0
    img_labels = np.array(img_labels)

    return img_data, img_labels


data, labels = getData()
train_data, test_data, train_labels, test_labels = model_selection.train_test_split(data, labels, test_size=0.20)

train_data, val_data, train_labels, val_labels = model_selection.train_test_split(train_data, train_labels,test_size=0.10)
print(len(train_data), " ", len(train_labels), len(test_data), " ", len(test_labels))

model = createModel(train_data)
```

```
checkpoint = keras.callbacks.ModelCheckpoint(filepath='./model/model1.h5',
save_best_only=True, monitor='val_loss', mode='min')

opt = keras.optimizers.Adam(learning_rate=0.001)
model.compile(optimizer=opt,              loss="sparse_categorical_crossentropy",
metrics=["accuracy"], )

history        =        model.fit(train_data,        train_labels,        epochs=3,
validation_data=(val_data, val_labels)
                  )
model.save('./model/model1.h5')
test_loss, test_acc = model.evaluate(test_data, test_labels)
print("Model Accuracy: ", test_acc, "Model Loss: ", test_loss)
plt.plot(history.history['acc'])
plt.plot(history.history['val_acc'])
plt.title('Model acc')
plt.ylabel('acc') plt.xlabel('epoch')
plt.legend(['train', 'test'], loc='upper left')
plt.show()
# summarize history for loss
plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])
plt.title('Model loss')
plt.ylabel('loss')
plt.xlabel('epoch')
plt.legend(['train', 'test'], loc='upper left')
plt.show()
```

Prediction

```
import tensorflow as tf
from tensorflow import keras
import cv2
import matplotlib.pyplot as plt
import  numpy  as np
SIZE = 120
model                                                                          =
keras.models.load_model(r'C:\Users\RE_cs_3\Desktop\TECHNICAL-
SOLUTIONS-FOR-VISUALLY-IMPAIRED-master\alzheimer-stage-
classifier-master\model\model.h5')
categories   =   ["NonDemented",   "MildDemented",   "ModerateDemented",
"VeryMildDemented"]

nimage         =          cv2.imread(r"C:\Users\RE_cs_3\Desktop\TECHNICAL-
SOLUTIONS-FOR-VISUALLY-IMPAIRED-master\alzheimer-stage- classifier-
master\test\1.jpg", cv2.IMREAD_GRAYSCALE)
image = cv2.resize(nimage,(SIZE,SIZE))
image = image/255.0
prediction = model.predict(np.array(image).reshape(-1,SIZE,SIZE,1))
pclass = np.argmax(prediction)
plt.imshow(image,cmap="gray")
pValue = "Prediction: {0}".format(categories[int(pclass)])
plt.title(pValue)
realvalue = "Real Value 1"
plt.figtext(0,0,realvalue)
plt.show()
```

Webpages

Index

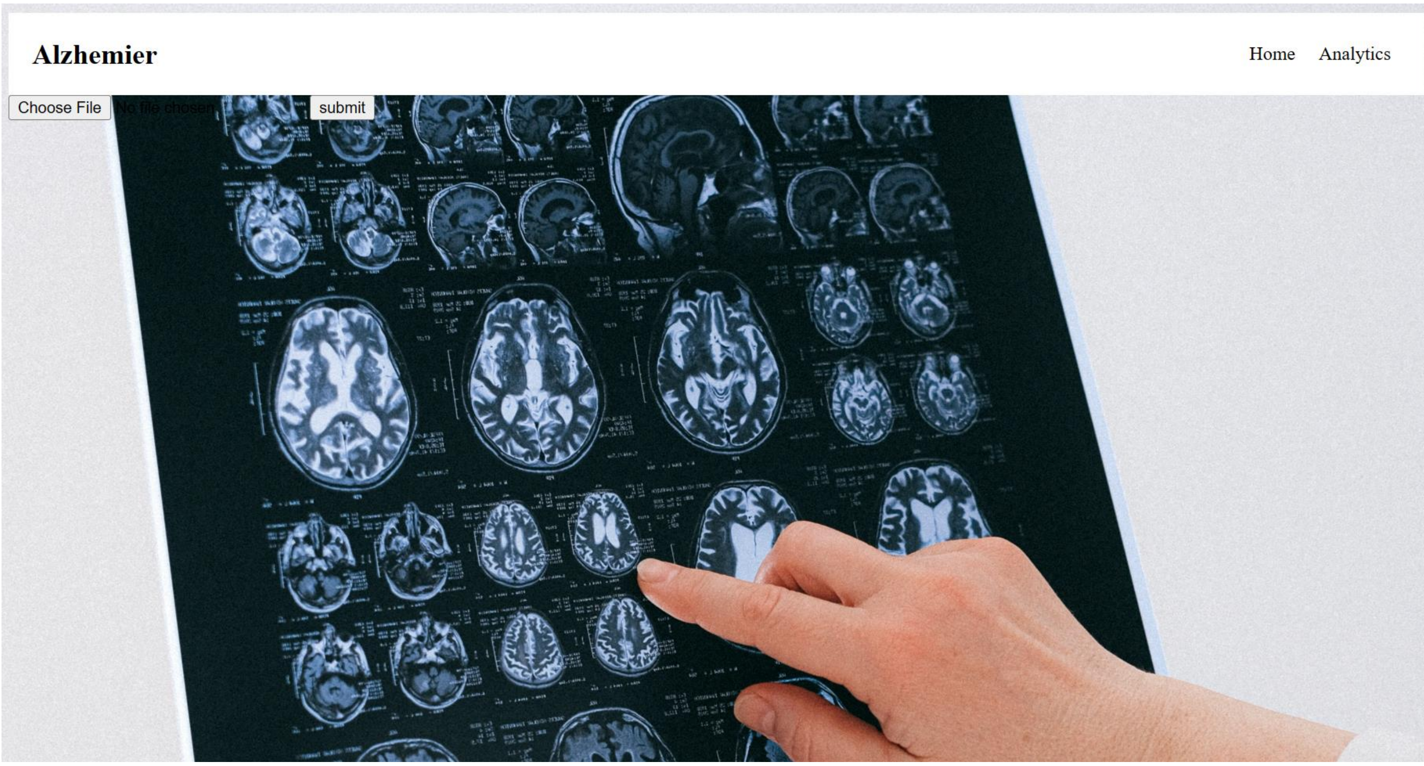
```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta http-equiv="X-UA-Compatible" content="IE=edge">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Predection</title>
   <style>
       body{
           background-image:url('/static/1.jpg');
           background-repeat:no-repeat;
           /* width:100%; */
           background-size:100% ;
   }
   </style>
</head>
<body>
           <form action="{{ url_for('Upload') }}" method='POST'
enctype="multipart/form-data">
       <input type="file" class="image_input" id='image' name='image'>
       <input type="submit" value="submit">
   </form>
</body>
</html>
```

Result

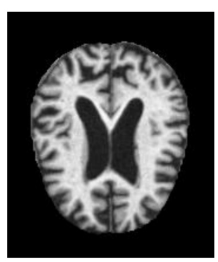
```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <form action="" method='POST' enctype="multipart/form-data">
        <p>{{value}}</p>

        <imgsrc="static\uploader\1.png" alt="">
    </form>
</body>
</html>
```

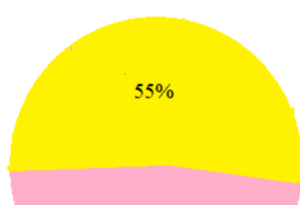
APPENDIX-II



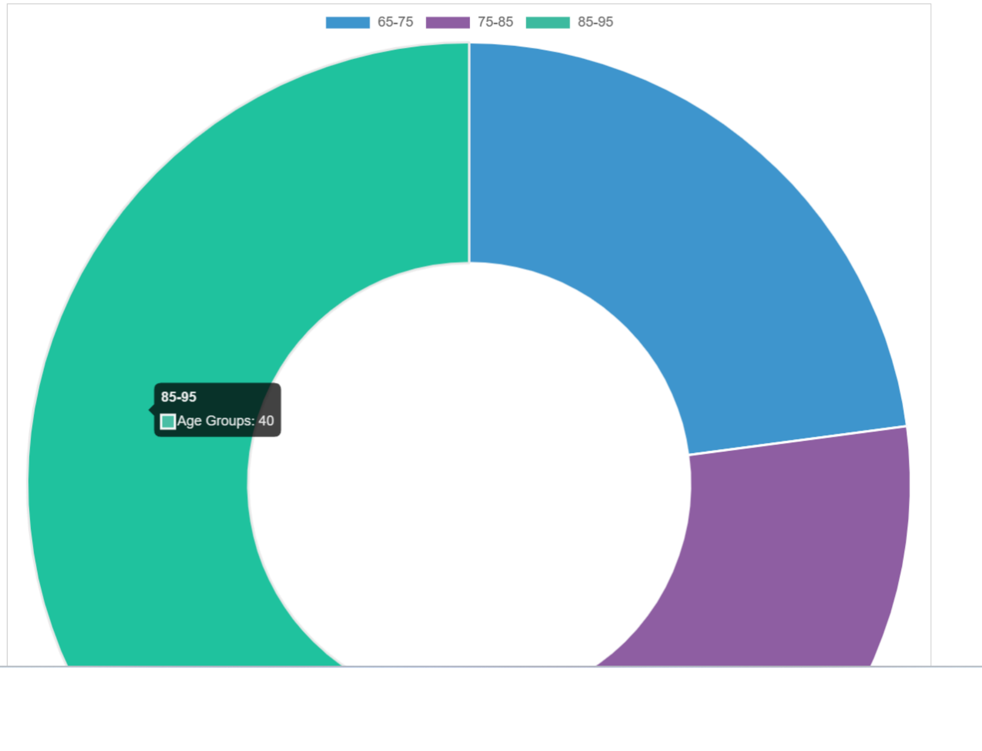
Predict: MildDemented



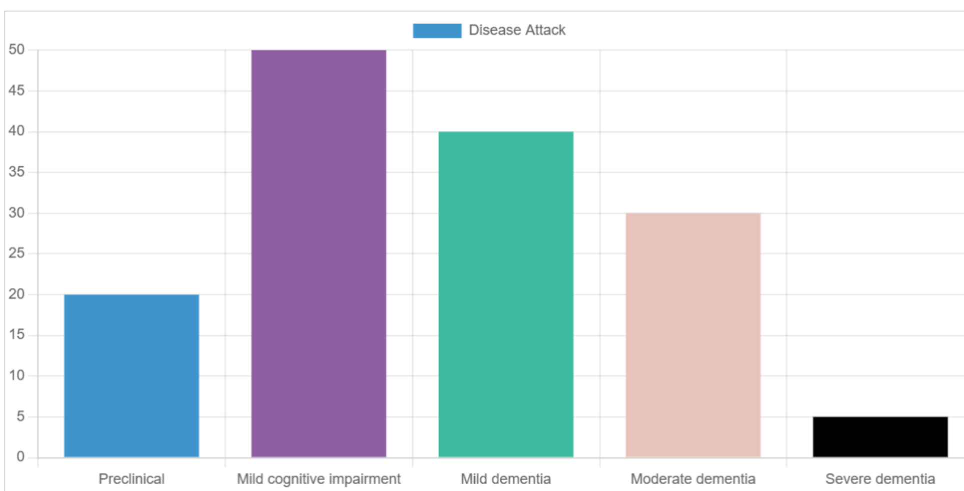
In the mild dementia stage, people may experience: Memory loss of recent events. Individuals may have an especially hard time remembering newly learned information and ask the same question over and over. Difficulty with problem-solving, complex tasks and sound judgments.

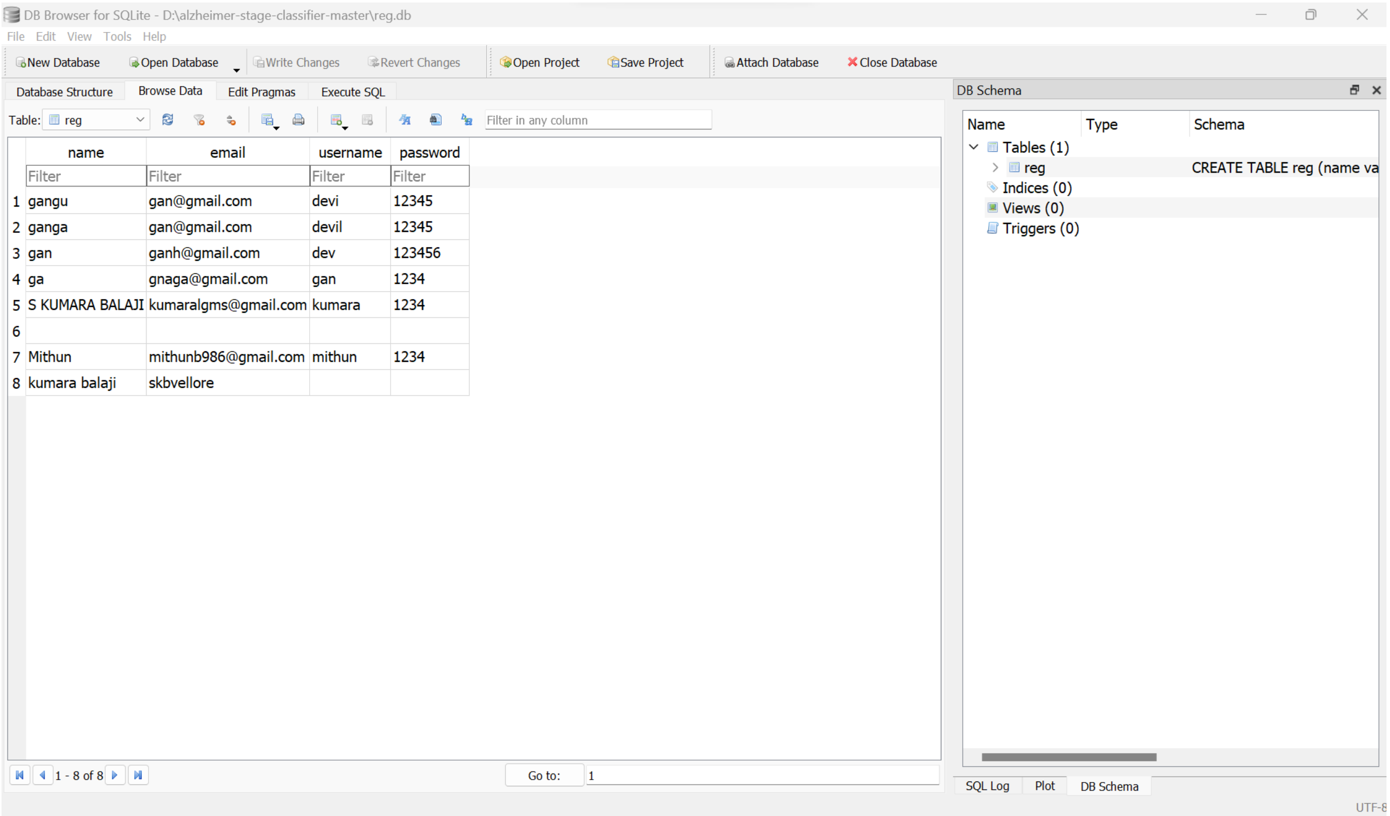


Age Groups Analysis



Disease Attack Analysis





	name	email	username	password
1	gangu	gan@gmail.com	devi	12345
2	ganga	gan@gmail.com	devil	12345
3	gan	ganh@gmail.com	dev	123456
4	ga	gnaga@gmail.com	gan	1234
5	S KUMARA BALAJI	kumaralgms@gmail.com	kumara	1234
6				
7	Mithun	mithunb986@gmail.com	mithun	1234
8	kumara balaji	skbvellore		

REFERENCES

[1] M. K. Hasan et al., "Lightweight Encryption Technique to Enhance Medical Image Security on Internet of Medical Things Applications," in IEEE Access, vol. 9, pp. 47731-47742, 2021

[2] Selvi, C.T., Amudha, J. & Sudhakar, R. Medical image encryption and compression by adaptive sigma filterizedsynorr certificateless signcryptiveLevenshtein entropy-coding-based deep neural learning. Multimedia Systems, 2021.

[3] Xiaohui Wang, Zheng Zhu, Fan Wang, Renjie Ni, Jun Wang, and Yuhen Hu, "Medical image encryption based on biometric keys and lower– upper decomposition with partial pivoting," Appl. Opt. 60, 24- 32 (2021)

[4]Hirano, H., Minagi, A. & Takemoto, K. Universal adversarial attacks on deepneural networks for medical image classification. BMC Med Imaging 21, 9 (2021).

[5]Sipola, T., S. Puuska, and T. Kokkonen, Model Fooling Attacks Against Medical Imaging: A Short Survey, Information & Security: An International Journal, vol. 46.

[6] Finlayson, S.G., Kohane, I.S., & Beam, A. (2018). Adversarial Attacks Against Medical Deep Learning Systems. ArXiv.

[7] Weijia Cao, Yicong Zhou, C.L. Philip Chen, Liming Xia, Medical image encryption using edge maps, Signal Processing, Volume 132, 2017.

[8] Weijia Cao, Yicong Zhou, C.L. Philip Chen, Liming Xia, Medical image encryption using edge maps, Signal Processing, Volume 132, 2017.

[9] M. Jain, R. C. Choudhary and A. Kumar, "Secure medical image steganography with RSA cryptography using decision tree," 2016 2nd International Conference on Contemporary Computing and Informatics (IC3I), 2016.

[10] Gerda Bortsova, Cristina González-Gonzalo, Suzanne C. Wetstein et al., Adversarial attack vulnerability of medical image analysis systems: Unexplored factors, Volume 73.

[11]Singh, Gurinder & Deep, Gaurav, MRI Medical Image and Steganography, Research gate , 2016

[12] Quist-Aphetsi Kester, Laurent Nana, Anca Christine Pascu, Sophie Gire, Jojo M. Eghan, NiiNarkuQuaynor, A Cryptographic Technique for Security of Medical Images in Health Information Systems, Procedia Computer Science,2015.