A

Project Report

On

Brain Stroke Detection Using ML Approach

Submitted to

Sant Gadge Baba Amravati University, Amravati

Submitted in partial fulfilment of

the requirements for the Degree of

Bachelor of Engineering in

Computer Science and Engineering

Submitted by

Nikhil Kiran Kulkarni	**Pratik Pradip Kuntawar**
(PRN:213120298)	(PRN:213120249)
Niraj Milind Saraf	**Gaurav Umesh Mahore**
(PRN:223120434)	(PRN:213120292)

Under the Guidance of

Prof. V. S. Mahalle

Assistant Professor, CSE Department

Department of Computer Science and Engineering

Shri Sant Gajanan Maharaj College of Engineering,

Shegaon – 444 203 (M.S.)

Session 2024-2025

SHRI SANT GAJANAN MAHARAJ COLLEGE OF ENGINEERING
SHEGAON – 444 203 (M.S.)
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that **Mr. Nikhil Kiran Kulkarni, Mr. Pratik Pradip Kuntwar, Mr. Niraj Saraf and Mr. Gaurav Umesh Mahore** students of final year Bachelor of Engineering in the academic year 2024-25 of Computer Science and Engineering Department of this institute have completed the project work entitled "**Brain Stroke Detection Using ML Approach**" and submitted a satisfactory work in this report. Hence recommended for the partial fulfilment of degree of Bachelor of Engineering in Computer Science and Engineering.

Prof . V. S. Mahalle
Project Guide

Dr. J. M. Patil
Head Of Department

Dr. S. B. Somani
Principal
SSGMCE, Shegaon

SHRI SANT GAJANAN MAHARAJ COLLEGE OF ENGINEERING, SHEGAON – 444 203 (M.S.)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that **Mr. Nikhil Kiran Kulkarni, Mr. Pratik Pradip Kuntawar, Mr. Niraj Saraf, Mr. Gaurav Umesh Mahore** students of final year Bachelor of Engineering in the academic year 2024-25 of Computer Science and Engineering Department of this institute have completed the project work entitled "**Brain Stroke Detection Using ML Approach**" and submitted a satisfactory work in this report. Hence recommended for the partial fulfillment of degree of Bachelor of Engineering in Computer Science and Engineering.

Mahalle

Internal Examiner

V. S. Mahalle

Name and Signature

Date: 9/5/2025

ARLadole 9/5/25

External Examiner

AR Ladole.

Name and Signature

Date:

Acknowledgement

It is our utmost duty and desire to express gratitude to various people who have rendered valuable guidance during our project work. We would have never succeeded in completing our task without the cooperation, encouragement and help provided to us by then. There are a number of people who deserve recognition for their unwavering support and guidance throughout this report.

We are highly indebted to our guide **Prof. V. S. Mahalle** for his guidance and constant supervision as well as for providing necessary information from time to time. We would like to take this opportunity to express our sincere thanks, for his esteemed guidance and encouragement. His suggestions broaden our vision and guided us to succeed in this work.

We are sincerely thankful to **Dr. J. M. Patil** (HOD, CSE Department, SSGMCE, Shegaon), and to **Dr. S. B. Somani** (Principal, SSGMCE, Shegaon) who always has been kind to extend their support and help whenever needed.

We would like to thank all teaching and non-teaching staff of the department for their cooperation and help. Our deepest thank to our parents and friends who have consistently assisted us towards successful completion of our work.

Nikhil Kiran Kulkarni (55)

Pratik Pradip Kuntawar(64)

Niraj Milind Saraf (56)

Gaurav Umesh Mahore(45)

CONTENTS

Abstract

Stroke is a critical medical condition caused by an interruption of blood flow to the brain that leads to severe neurological damage or death if not properly diagnosed and treated. Traditional stroke diagnosis employs manual MRI and CT scan examinations, which are subjective, time-consuming, and prone to human error. Previous models were primarily binary classifiers that could only determine the presence or absence of a stroke, which is not adequate for proper treatment. To overcome this shortcoming, we developed our own CNN-based model that first detects the presence of a stroke and then classifies it as ischemic stroke, haemorrhagic stroke, or normal cases. This multi-classification is more accurate and provides fine-grained information needed for medical decision-making. Our approach compared to traditional approaches and previous models is more accurate, quicker, and more accurate in stroke classification. With deep learning, our model allows for earlier detection of a stroke, less human error, and better patient outcomes through automatic and accurate diagnosis.

Keywords: Convolutional Neural Network, Brain Stroke, Ischemic Stroke, Hemorrhagic Stroke, Deep Learning, Convulationallayer, Dense Layer, Confusion Matrix.

List of Abbreviations

Abbreviation	Description
Conv2D	Convolutional Layer
Max2D	MaxPooling layer
RELU	Rectified Linear Unit
MRI	Magnetic Resonance Imaging
CT	Computed TomoGraphy
ANN	Artifical Neural Network
CNN	Convolutional Neural Network

List of Figures

List of Tables

CHAPTER 1

INTRODUCTION

INTRODUCTION

1.1 PREFACE

Brain stroke is one of the most critical and life-threatening medical conditions that affect millions of people worldwide. A stroke occurs when the blood supply to a part of the brain is interrupted or significantly reduced, leading to a lack of oxygen and essential nutrients. This can cause brain cells to begin dying within minutes, resulting in long-term disabilities such as paralysis, speech difficulties, or memory loss—and in severe cases, it can lead to death. Due to the brain's extreme sensitivity to oxygen deprivation, early detection and immediate treatment are essential to minimize the damage and improve the chances of recovery.

Despite its severity, detecting a stroke at an early stage remains a major challenge. Symptoms like sudden weakness, confusion, difficulty speaking, blurred vision, or headaches often vary greatly among individuals and may be subtle or even go unnoticed, especially in elderly or unconscious patients. This problem is worsened in rural or underdeveloped regions, where access to advanced diagnostic tools like MRI scans and experienced neurologists may be limited.

In recent years, the advancement of Artificial Intelligence (AI) **and** Machine Learning (ML)has opened new possibilities in the field of healthcare. These technologies are being used to analyze large and complex datasets, including medical imaging data, to assist in faster and more accurate diagnoses. One of the most promising areas of application is in medical image analysis**,** where machine learning models are trained to detect patterns in images that might be difficult for the human eye to recognize.

This project, titled "Brain Stroke Detection Using Machine Learning Approach**"**, focuses on applying machine learning techniques to analyze MRI images of the brain in order to detect the presence of stroke. Instead of relying on structured data such as age, blood pressure, or lifestyle indicators, the system directly processes medical images using image processing and deep learning methods. The model learns from previously labeled MRI scans and is trained to distinguish between normal brain images and those showing signs of stroke.

The key goals of this project are:

- To build a machine learning model that can accurately detect whether a person has suffered a stroke by analyzing their MRI brain scan.
- To identify the affected region of the brain using image classification and segmentation techniques, which helps in determining the severity and possible effects of the stroke.

The proposed system aims to assist medical professionals by providing a second opinion that can reduce human error, speed up the diagnosis process, and support timely decision-making in emergencies. It can be particularly useful in areas with limited access to neurologists or radiologists, as the system can be deployed on digital platforms such as cloud servers or mobile applications.

This project aims to demonstrate how the integration of machine learning with medical imagingcan be a powerful tool in the early detection and management of strokes. By automating the analysis of MRI images, the system not only saves valuable time but also supports better treatment outcomes—thereby contributing positively to public health and medical technology.

1.2 MOTIVATION

Brain stroke is a major global health issue that affects individuals across all age groups, but especially the elderly. According to the World Health Organization (WHO) and various health surveys:

- Stroke is the second leading cause of death and the third leading cause of disability worldwide.
- Each year, around 15 million people suffer from a stroke globally.
- Out of these, 5 million die, and another 5 million are left permanently disabled.
- In India alone, the incidence of stroke is around 119–145 per 100,000 people, and the numbers are increasing due to sedentary lifestyles, poor diet, stress, and lack of awareness.

These numbers clearly show that stroke is not just a medical condition—it is a serious public health crisis. One of the biggest challenges in treating stroke is **the** timesensitive nature of the disease. The quicker the stroke is detected, the better the chances of saving brain function and reducing long-term damage.

In real-life situations, especially in emergency rooms or rural health centers, doctors often face difficulties in diagnosing strokes quickly due to:

- Lack of specialized doctors (neurologists or radiologists)
- Unclear or varied symptoms in patients
- Delay in processing and analyzing MRI scans manually
- Misinterpretation or human error in diagnosis

These challenges motivated us to explorehow machine learning can help automate and speed up stroke detection using MRI images. Machine learning models, especially those trained on medical image data, have shown promising results in detecting diseases like pneumonia, tumors, and now strokes. These models can analyze large numbers of images quickly and accurately, reducing the time doctors need to make decisions and improving patient care.

Additionally, this project also addresses a social challenge—bringing advanced medical technology to areas with limited resources. If deployed in rural hospitals or through cloud-based tools, this system can assist in stroke diagnosis where human expertise is unavailable.

In summary, the key motivations behind this project are:

- The urgency of early stroke detection to reduce death and disability rates
- The growing number of stroke cases globally and in India
- The lack of fast and accurate diagnostic tools, especially in remote or underdeveloped areas
- The power of machine learning and medical imaging in improving healthcare outcomes

1.3 PROBLEM STATEMENT

Brain stroke is a sudden medical emergency that can cause permanent brain damage or even death if not treated quickly. The key to reducing its impact lies in detecting it as early as possible. However, in many real-world situations, stroke detection is often delayed because the symptoms are not always clear and can be different for every person. Also, stroke diagnosis usually depends on the manual analysis of MRI (Magnetic Resonance Imaging) scans by expert doctors, which can take time and may not always be available—especially in rural or less-equipped healthcare centers.

Traditional stroke detection methods involve:

- Understanding patient symptoms
- Physically analyzing MRI brain images
- Consulting neurologists or radiologists for confirmation

But these steps can be time-consuming, error-prone, and may lead to delayed treatment—**w**hich can cost someone their life or lead to permanent disability**.**

With the advancement of technology, machine learningandimage processing techniques have shown great potential in helping doctors detect diseases quickly and more accurately. These technologies can be trained to understand patterns in medical images and make intelligent predictions. However, applying such methods effectively to stroke detection still remains a challenge in many areas.

Currently, stroke detection largely depends on the manual analysis of MRI scans by healthcare professionals. This process involves identifying subtle differences in the brain's structure that may indicate a stroke, which can be time-consuming, subjective, and prone to human error. Additionally, MRI image interpretation requires expertise and may not always be available promptly, particularly in rural or under-resourced medical centers.

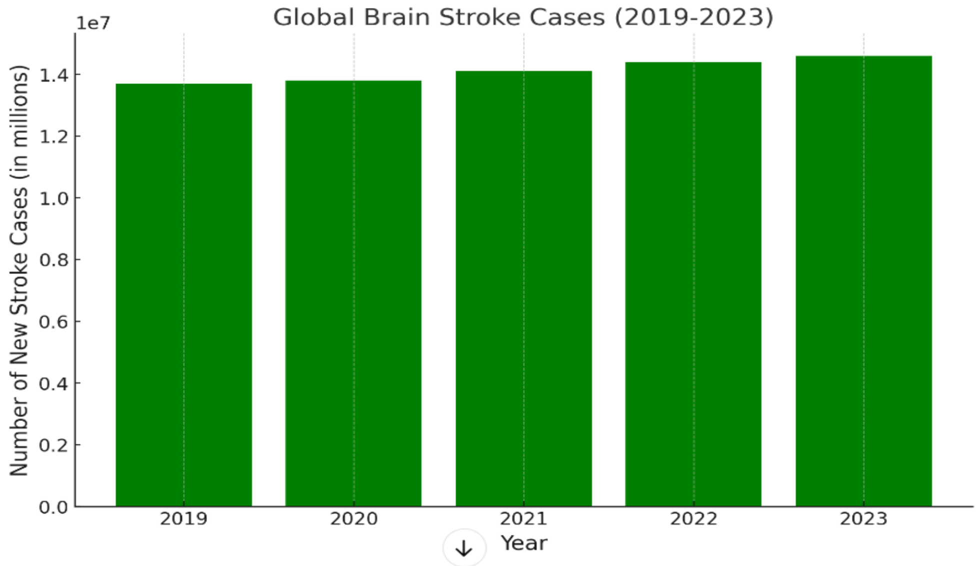
The main challenges faced in stroke detection are:

1. **Time Sensitivity**: A stroke causes brain damage within minutes, and delays in diagnosis can result in permanent disability or death. The current methods are often slow, and early signs of stroke may go unnoticed.
2. **Human Error**: Manual analysis of brain images is complex and prone to mistakes, especially when doctors are overburdened with work or lack experience with certain types of strokes.

3. **Limited Access to Experts**: In remote regions or developing countries, advanced medical imaging techniques and skilled neurologists may not be available, delaying diagnosis and treatment.
4. **Variability in Symptoms**: Stroke symptoms vary from person to person, and the signs may be subtle or confused with other conditions, making it difficult to recognize a stroke in its early stages.

1.4 STATISTICS

Here's a breakdown of brain stroke statistics globally.



Global Brain Stroke Cases (2019-2023) – Insights

- The bargraph represents the global brain stroke cases over the last five years
- The number of new stroke cases worldwide remains
- consistently high, ranging in tens of millions annually.
- There is a gradual increase in global stroke cases from 2019 to 2023, indicating a rising trend in stroke prevalence.

1.5 OBJECTIVES

1) To study brain MRI images terminology, existing brain MRI image preprocessing and brain stroke detection technique.
2) To preprocess and prepare MRI image data set use for brain stroke detection.
3) To design and develop a basic structure of brain stroke detection system.

4) To implement proposed feature extraction and classification technique used for efficiently detect brain stroke.
5) To find result of performance measuring parameters

1.6 SCOPE OF PROJECT

The **scope** of this project focuses on developing a machine learning-based system for brain stroke detection using MRI images. The project aims to automate the detection process, enhance diagnostic accuracy, and improve healthcare accessibility. The primary areas covered in the project are as follows:

1) **MRI Image-Based Stroke Detection**
    a. The system will take MRI images of the brain as input and use machine learning techniques to predict whether a person is suffering from a stroke. The model will analyze the MRI scan for signs of stroke, helping healthcare professionals in quick decision-making.
2) **Stroke Type Classification**
    a. Once the system detects the presence of a stroke, it will classify the type of stroke into two categories:
        i. **Ischemic Stroke**: Caused by blockage in the blood supply to the brain.
        ii. **Hemorrhagic Stroke**: Caused by bleeding in or around the brain.
    b. This classification will be based on visual patterns detected in the MRI images, providing doctors with immediate information about the type of stroke, aiding in faster treatment.
3) **Identification of Affected Brain Region**
    a. The system will go a step further by identifying the affected region of the brain. By analyzing the MRI scan, the system will provide predictions about which part of the brain has been impacted by the stroke, assisting doctors in determining the severity and possible treatment methods.
4) **Automated Stroke Diagnosis**
    a. The project aims to automate the stroke diagnosis process, reducing the dependence on manual interpretation by radiologists. By using

machine learning models trained on MRI images, the system will provide fasterand more consistent results, reducing the risk of human error.

5) **Accessibility for Healthcare Professionals**
    a. The system will be designed to be accessible to healthcare professionals, such as radiologists, doctors, and emergency room personnel. The goal is to create a user-friendly interface that will allow these professionals to interact with the system efficiently and obtain stroke diagnoses quickly and can help professionals to use this application efficiently for their patients.
6) **Real-Time Predictions for Emergency Care**
    a. The system will provide **real-time predictions** for stroke detection, making it suitable for use in emergency rooms and urgent care settings. The quicker a stroke is detected, the better the chances for treatment, so the system will be optimized for rapid results.
7) **Integration for Remote Healthcare Settings**
    a. One of the goals of this project is to make stroke detection accessible even in **remote or underdeveloped areas**. By using cloud-based or mobile systems, healthcare professionals in areas lacking advanced diagnostic tools can access the machine learning model and receive stroke predictions without needing a radiologist on-site.
8) **User-Friendly Interface for Medical Professionals**
    a. The project will include a **simple and intuitive user interface** that healthcare professionals can easily use. The interface will display stroke predictions, classification results, and details about the affected brain region in a clear, easy-to-understand format.

CHAPTER 2

LITERATURE REVIEW

LITERATURE REVIEW

[1]G. Ravi Kumar, P. Vyshnavi, S. Prasanna, T. Harshavardhan Reddy, C. Charanya, and P. Chandrababu, in their research paper ***Brain Stroke Detection Using Machine Learning*** (International Journal of Research in Engineering, Science and Management, Vol. 5, No. 3, March 2022), explored various machine learning algorithms for detecting brain strokes. The study focuses on using algorithms like Support Vector Machine, Logistic Regression, Random Forest, and XGBoost for stroke classification. The authors emphasize that ischemic stroke is the most common and fatal type, especially in developing countries. Their model involves data collection, preprocessing, training, and performance evaluation. Among the tested algorithms, Random Forest achieved the highest accuracy, demonstrating its effectiveness in early stroke detection. The proposed system aims to reduce stroke-related deaths by enabling faster and more reliable diagnosis.

[2]Senjuti Rahman, Mehedi Hasan, and Ajay Krishno Sarkar**,** in their paper *Prediction of Brain Stroke using Machine Learning Algorithms and Deep Neural Network Techniques* (European Journal of Electrical Engineering and Computer Science, Vol. 7, Issue 1, January 2023), investigate the use of both classical machine learning and deep neural networks for early stroke prediction. They employ a publicly available Kaggle dataset of patient records (including demographics, medical history, and lifestyle factors) and apply dimensionality reduction via PCA. Multiple classifiers—XGBoost, AdaBoost, LightGBM, Random Forest, Decision Tree, Logistic Regression, K-Nearest Neighbors, SVM (linear), and Naive Bayes—are trained alongside 3-layer and 4-layer feed-forward neural networks. Their results show that the Random Forest model attains the highest accuracy among the machine learning methods (99%), while the deeper ANN achieves 92.39% accuracy. Overall, their study demonstrates that ensemble machine learning techniques outperform the tested deep learning architectures for this tabular stroke dataset.

[3]Tessy Badriyah, Nur Sakinah, Daisy RahmaniaSyarif, and Iwan Syarif, in their paper ***Machine Learning Algorithm for Stroke Disease Classification*** (Proc. of the 2nd International Conference on Electrical, Communication and Computer Engineering, ICECCE 2020, Istanbul, Turkey), develop an end-to-end pipeline for classifying CT-scan images into ischemic and hemorrhagic strokes. They preprocess raw DICOM scans (conversion to JPG, cropping, scaling, grayscale, and augmentation), extract six texture features via Gray-Level Co-Occurrence Matrix, then compare eight classifiers (KNN, Naive Bayes, Logistic Regression, Decision Tree, Random Forest, MLP, a deep-learning model, and SVM) using 10-fold and Leave-One-Out validation. Their results show that Random Forest achieves the best performance—95.97% accuracy, 94.39% precision, 96.12% recall, and 95.39% F_1-score—demonstrating its effectiveness for automated stroke subtype detection from CT images.

[4]Bhagyashree Rajendra Gaidhani, Dr. R. Rajamenakshi, and Samadhan Sonavane, in their paper *Brain Stroke Detection Using Convolutional Neural Network and Deep Learning Models* (Proc. of the 2nd International Conference on Intelligent Communication and Computational Techniques, ICCT 2019, Manipal University Jaipur), present a two-stage deep-learning pipeline for MRI-based stroke diagnosis. They train a LeNet CNN on ATLAS T1-weighted scans (229 stroke and 400 normal volumes)—after denoising, normalization, and resizing—to classify images as normal or abnormal, achieving **96–97% accuracy**. Abnormal scans are then fed into a SegNet encoder–decoder for semantic segmentation of lesion areas, yielding **85–87% segmentation accuracy**. Their work demonstrates that lightweight CNN architectures can deliver fast, reliable stroke detection and localization in medical imaging.

[5]Md. Ashrafuzzaman1, Suman Saha2, and Kamruddin Nur3, in his paper "Prediction of Stroke Disease Using Deep CNN Based Approach," presents a deep learning-based methodology for early prediction of stroke. The study employs a Convolutional Neural Network (CNN) model to predict stroke likelihood using a publicly available healthcare dataset comprising 5110 samples with 11 key features, including age, gender, hypertension, and smoking status. The data is preprocessed by handling missing values and normalizing continuous features to ensure optimal model performance. The CNN model achieves a prediction accuracy of 95.5%, outperforming traditional machine learning models like Logistic Regression, Support Vector Machines (SVM), and Random Forest. The model is trained on 80% of the dataset and tested on the remaining 20%, showcasing high classification accuracy for stroke prediction. The results demonstrate that CNN can be a powerful tool in medical diagnostics, providing accurate early stroke detection for effective intervention.

Table .1 LITERATURE Review

Sr.No	Title of Paper	Publication details Journal/ Conference Year/ Vol/ Page	Methodology	Key Finding
1	Brain St Detection U Machine Learning	International Journal of Research in Engineering, Science and Management, Vol. 5, No. 3, March 2022, pp. 35–36	Various ML algorithms like SVM, XGBoost, SGD, Decision Tree, and Random Forest are applied to a stroke dataset, with PCA for preprocessing and evaluation based on accuracy, precision, and recall.	Random Forest algorithm achieved the highest accuracy among the applied machine learning models for predicting brain stroke, helping in early detection and reducing death rates.

2	Prediction of Brain Stroke u Machine Learning Algorithms and Deep Neural Network Techniques	European of Journal ıary Electrical Engineering and Computer Scienc Vol. 7, Issue 1, Ja 2023, pp. 23–30.	Performed PCA for dimensionality reduction on the Kaggle stroke dataset, then evaluated multiple classifiers (Random Forest, XGBoost, AdaBoost, LightGBM, Decision Tree, Logistic Regression, KNN, SVM, Naive Bayes) alongside 3- and 4layer ANNs.	Random Forest achieved the highest accuracy of 99%, while the 4layer ANN reached 92.39%. Overall, machine learning outperformed deep neural networks in stroke prediction
3	Machine Learning Algorithm for Stroke Disease Classification	Proc. of the 2nd International Conference on Electrical, Communication and Computer ring Enginee June (ICECCE), 12–13 2020, Istanbul, Turkey	The methodology involved data collection, preprocessing, feature extraction, and classification using machine learning algorithms.	Random Forest gave the best performance Hyperparamet er tuning via random/Bayes ian search on deep learning models can further boost accuracy up t

4	Brain Strok Detection Usin Convolutional Neural Network and Deep Learnin Models	Proc. 2nd Int'l Conf. on Intelligent Communication and Computational Techniques (ICCT), Manipal Univ. Jaipur, Sep 28–29 2019	- **Data Acquisition & Pre-processing:** ATLAS dataset (229 stroke, 400 normal MRI scans), conversion to 2D arrays, median filtering, normalization. - **Classification (Phase I):** LeNet CNN (3 conv + 2 pooling + 1 FC + softmax), SGD optimizer, binary cross-entropy, 25 epochs, batch 64. - **Segmentation (Phase II):** SegNet encoder–decoder, 4 conv + pooling + upsampling layers, MSE loss, SGD, 35 epochs, batch 5.	**Classification:** 97% train / 96% test accuracy. - **Segmentation:** 87% train / 84–85% test accuracy, overall 91% lesion delineation accuracy. - Demonstrate CNNs (LeNet/SegNet) can accurately detect and localize stroke regions in brain MRI.

CHAPTER 3

METHODOLOGY

METHODOLOGY

3.1 Background

Stroke is one of the leading causes of death and disability worldwide. Early and accurate detection of stroke type—ischemic or hemorrhagic—is critical for timely treatment. Manual diagnosis through CT or MRI scans can be time-consuming and subject to human error. Therefore, an automated system that can accurately detect brain stroke types using deep learning models like Convolutional Neural Networks (CNN) is highly beneficial. CNNs are capable of learning complex patterns in medical images and have shown excellent results in medical image classification tasks.

3.2 Proposed Brain Stroke Detection System using CNN

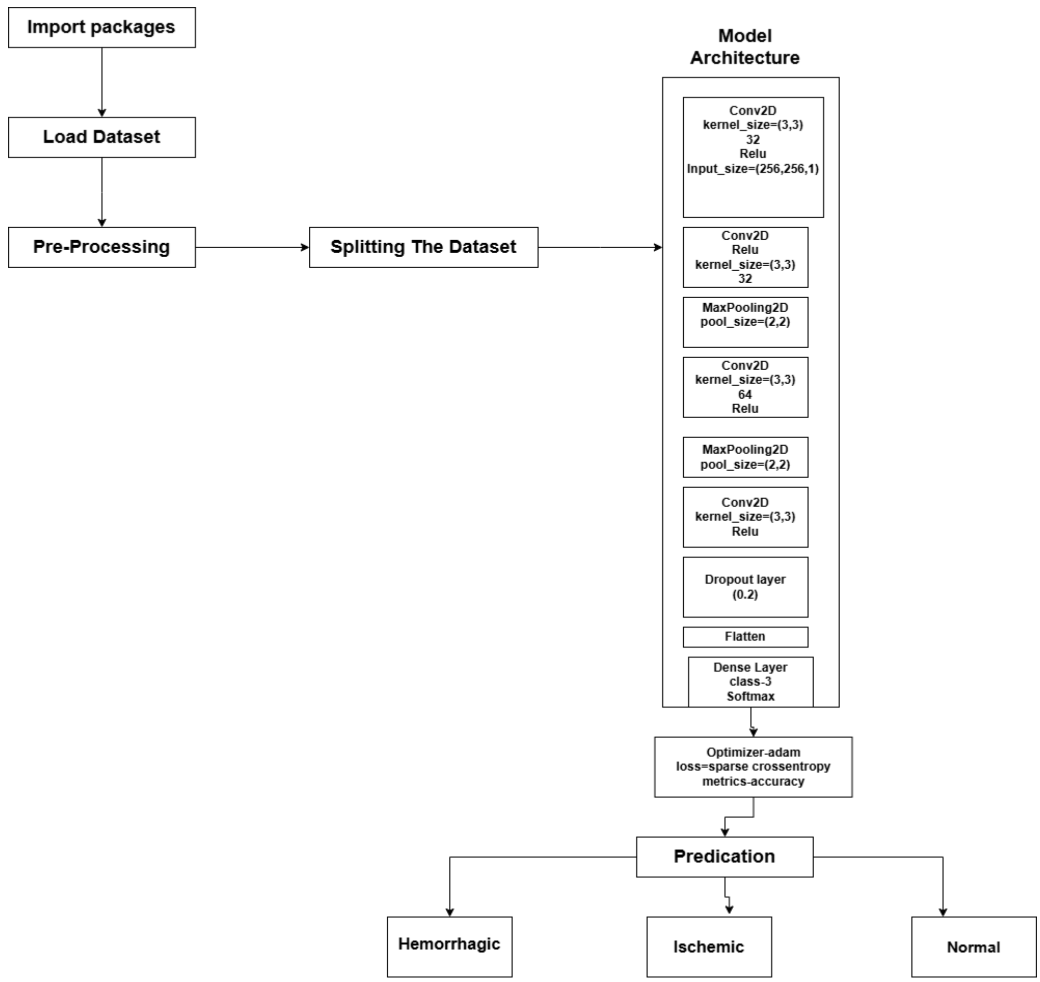


Fig 3.2 Block Diagram of Model

1. Import Packages

Essential Python libraries such as TensorFlow, Keras, NumPy, Matplotlib, etc., are imported.

These packages provide tools for deep learning model building, image processing, and result visualization.

---

2. Load Dataset

Brain CT or MRI scan images are loaded into the system.

The dataset contains labeled images representing different classes: Hemorrhagic Stroke, Ischemic Stroke, andNormal.

This data will be used for training and testing the CNN model.

---

3. Pre-processing

Input images undergo several transformations to prepare them for the CNN model:

Resizing all images to a uniform dimension (e.g., 256x256).

Normalization to scale pixel values between 0 and 1.

Noise Reduction to clean the images.

(Optional) **Data Augmentation** (rotation, zooming, flipping) to improve model generalization.

---

4. Splitting the Dataset

The dataset is split into:

Training Set (to teach the CNN).

Validation/Test Set (to evaluate how well the CNN performs on unseen images).

A typical split ratio is 80% for training and 20% for testing.

5. Model Building (CNN Architecture)

A Convolutional Neural Network (CNN) is designed, consisting of:

Conv2D Layers: Apply filters to detect features like edges, textures, and patterns.

ReLU Activation: Introduces non-linearity after each convolution.

MaxPooling2D Layers: Reduces spatial dimensions to focus on important features.

Flatten Layer: Converts the 2D feature maps into a 1D feature vector.

Dense Layer: Fully connected layer that combines extracted features for final classification.

Output Layer: Classifies the image into Hemorrhagic, Ischemic, or Normal using the

Softmax activation function.

6. Compilation (Optimizer, Loss, Metrics)

Optimizer: Adam optimizer is used to update weights during training efficiently.

Loss Function: Sparse categorical crossentropy is used since it's a multi-class classification problem.

Metrics: Accuracy is used to monitor model performance.

7. Model Training

The CNN is trained on the pre-processed training dataset for a set number of epochs. During training, the model learns to minimize the loss and improve its prediction accuracy.

8. Prediction

After training, the model predicts the type of stroke by analyzing new brain images:

Hemorrhagic Stroke

Ischemic Stroke

Normal (No Stroke)

The prediction is based on the features learned during training.

3.2.1 Input Query Image

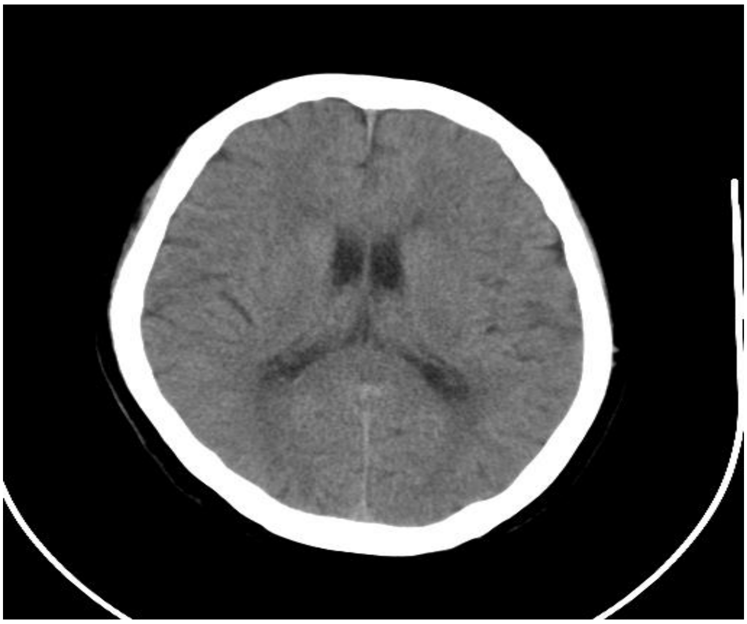


Fig 3.2.1 Input Image

3.2.2 Pre-processing of Images

1. **Resizing**

   All the images were resized to 256×256 pixels for a uniform input size. MRI scans are available in various resolutions, and these can cause inconsistencies while training the model. Resizing ensures the models are all equal in size, and this is what optimizes the computational efficiency and allows the CNN model to learn at its optimal capacity

2. **Grayscale**

   Since MRIscans predominantly contain structural information rather than color information, we have all the images converted to grayscale.

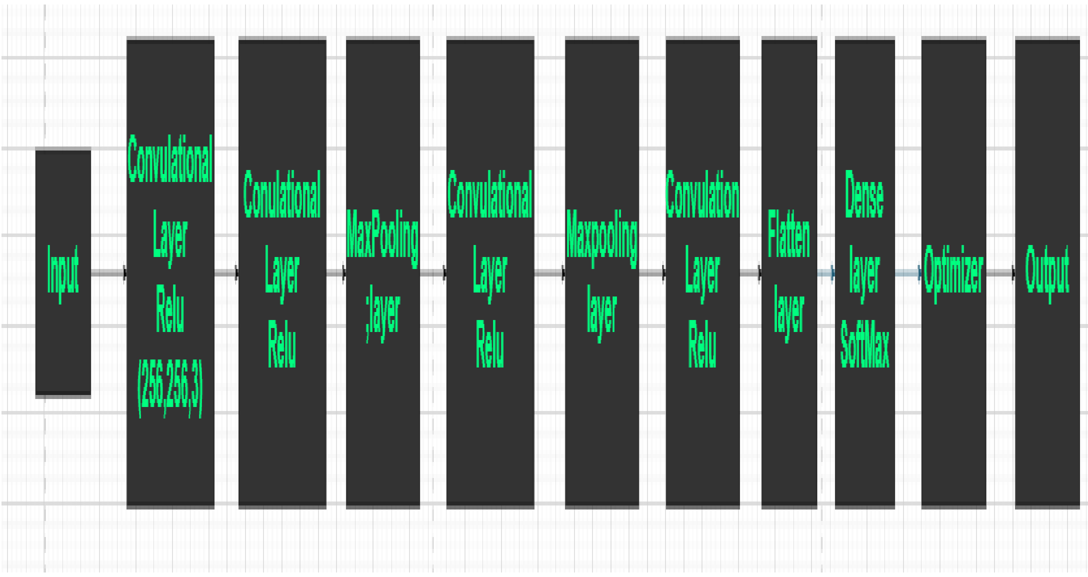


Figure 3.3 CNN Architecture

   Color channels reduces computational complexity without impacting crucial features required for stroke classification. This will help ensure that the model highlights the basic differences in intensity, which play a critical role in identifying ischemic and hemorrhagic strokes.

3. **Normalization**

   We normalized by rescaling pixel values to the range [0,1] by dividing them by 255. Normalization makes the training process stable, prevents large weight updates, and keeps pixel values in a standard range. Normalization speeds up convergence and helps the model generalize more to unknown MRI scans.

4. **Scaling(Optional)**

   Scaling techniques are employed to modulate intensity variations in MRI images to further support feature extraction. This step is not mandatory but enhances the contrast, with stroke-damaged areas being more prominent than normal brain

3.2.3 Feature Extraction and Classification using CNN

3.3 Model Architecture

The suggested Convolutional Neural Network (CNN) model is intended for brain stroke classification automatically, separating Ischemic, Hemorrhagic, and Normal brain scans. The structure includes several convolutional layers for feature extraction, pooling layers for reducing dimensions, and fully connected layers for classification

3.3.1 Input layer

The model begins with an input layer that takes images of brain scans with the size 256×256×3. The three channels are for RGB color space, which is preserved in order to maintain fine texture details in medical scans. Pixel values are normalized to 0 to 1 so that the training is stable and there are no problems associated with different scales of intensity.

3.3.2 Convulation Layer

The Convolutional Layer is the fundamental block of the CNN model, tasked with learning spatial hierarchies and patterns automatically from brain stroke images. In our architecture, the initial convolutional layer receives an input image of 256×256×3 size, where 3 indicates the RGB channels. The layer convolves 8 learnable filters (kernels) of 3×3 size over the image to capture local patterns like edges, textures, and intensity variations in the brain scan.Each filter traverses the input image, carrying out an element-wise multiplication and then summation to produce a feature map. This process captures essential features like stroke-affected areas and structural problems in the brain.

3.3.2.a) Activation Function: ReLU

Following convolution, a Rectified Linear Unit (ReLU) activation is used to bring in non-linearity and allow effective gradient propagation. The ReLU function is given by:

F(x)=max(0,x)

The activation enables the model to learn sophisticated patterns by removing negative values and only letting important positive activations pass through, enhancing convergence speed and avoiding the vanishing gradient issue.

3.3.3 Pooling layer

The Pooling Layer is a pivotal component of the suggested CNN model in that it decreases the spatial size of feature maps without sacrificing the most crucial information. For this model, a MaxPooling layer with the pool size (2×2) is utilized after every convolutional layer. This process decreases computational complexity without losing vital stroke-related features, thus making the model more

efficient. The main purpose of the pooling layer is to obtain dimensionality reduction, feature invariance, andoverfitting suppression. Through taking the maximum value of every 2×2 region, MaxPooling guarantees that the most important activations are preserved so that the model can concentrate on important stroke-related patterns while not caring about any worthless details or noise. Here, the input feature map size of 256×256 goes through MaxPooling and is then shrunk to 128×128, then by additional pooling it is brought down to 64×64. This progressive downsampling refines the feature extractions before it is passed through the fully connected layers. MaxPooling over AveragePooling has been used as the former captures the high-intensity features in a better manner, which plays an important role in medical images in identifying regions of stroke attack. By removing redundant features, pooling layers contribute to the development of a more effective and robust stroke classification model, allowing it to differentiate between ischemic stroke, hemorrhagic stroke, and normal brain scans with higher precision.

3.3.4 Flatten layer

The Flatten Layer acts as a transition layer between the convolutional and fully connected layers of the suggested CNN model. Following feature extraction by the convolutional and pooling layers, the output feature maps remain in a multi-dimensional spatial form. In order for these features to be processed by the Dense (fully connected) layers, the Flatten layer flattens the multi-dimensional feature maps into a one-dimensional vector. This process guarantees that each feature that wasextracted has its contribution in the final classification procedure. Since the final pooling layer gives out a smaller feature map, the Flatten layer reshapes these values in one long vector that can actually be efficiently handled by the fully connected layers in order to carry out the classification. This is an important step as it readies the data for learning sophisticated relationships between features and the resultant stroke types—ischemic, hemorrhagic, or normal.

3.3.5 Dense Layer

The Dense Layer, or fully connected layer, is employed for final classification. Here, Softmax activation function on the output layer is applied for multi-class classification to provide probability scores of all the classes (ischemic stroke, hemorrhagic stroke, and normal). The fully connected layers allow the model to represent complex, high-level information by combining extracted features in previous layers. The Dense layer neurons are given inputs from all the neurons of the Flatten layer above them, enabling end-to-end feature learning. Softmax function is also helpful in normalizing the output into a probability distribution such that the model is able to output the most probable class label of the input brain scan. The Dropout methods can also be applied to prevent overfitting and enhance the model's generalization. This design enables the CNN to identify stroke types precisely, enhancing diagnostic accuracy compared to traditional binary classification techniques.

3.3.5.a) SoftMax Function:

The Softmax activation function is used in the final Dense layer to handle multi-class classification, ensuring that the CNN model predicts one of the three categories—ischemic stroke, hemorrhagic stroke, or normal. The Softmax function transforms the raw output values (logits) from the Dense layer into probabilities, making them sum up to 1.

3.3.6 Output layer

The Output Layer is the last step of the CNN model, which is used to generate the results of classification of brain stroke. It contains three neurons, each of which is related to one of the categories of stroke:

- Ischemic Stroke
- Hemorrhagic Stroke
- Normal (No Stroke)

For ensuring the model yields useful predictions, the Softmax activation function is used here. This takes the raw scores (logits) produced by the preceding Dense layer and scales them into a probability distribution in which the sum of all the output values is 1. The class with the highest probability value is picked as the prediction.

3.3.7 Model Compilation And Optimzation

Once the model's CNN architecture has been defined, the model is compiled to get it ready to learn. Compilation includes choosing an optimizer, loss function, and measuring metric in order to learn efficiently and classify properly.

- Optimizer – Adam (Adaptive Moment Estimation): Adam is a sophisticated optimization algorithm that leverages the advantages of Momentum and RMSprop, dynamically adjusting learning rates for each parameter. It facilitates faster convergence and prevents getting trapped in local minima, making it highly suitable for deep learning applications such as stroke classification.
- Loss Function – Sparse Categorical Cross-Entropy: As the task of classification contains three unique classes (Ischemic, Hemorrhagic, and

Normal), Sparse Categorical Cross-Entropy is employed for calculating the discrepancy between predicted and true labels. Unlike standard categorical cross-entropy, it is effective for integer-labeled classes and enhances the performance of the model.

- Evaluation Criterion – Accuracy: Accuracy is used as the primary criterion to measure the performance of the model in identifying stroke types correctly. It measures the ratio of correctly predicted instances to the total predictions, giving a clear idea of the effectiveness of the model.

3.4 Summary

The proposed Brain Stroke Detection System using CNN provides an end-to-end solution for early detection and classification of stroke types from brain images. By automating feature extraction and classification, the CNN reduces human error, speeds up diagnosis, and assists medical professionals in providing timely treatment. The system leverages image pre-processing techniques and a carefully designed CNN model to achieve high accuracy and reliability.

CHAPTER 4

IMPLEMENTATION

IMPLEMENTATION

4.1Background

Brainstroke is a critical medical condition caused by the disruption of blood flow to the brain, leading to cell damage and life-threatening consequences. Detecting strokes early is crucial for providing timely treatment and improving patient survival rates. Traditionally, stroke diagnosis relies on expert interpretation of medical imaging such as CT scans or MRIs, which can be time-consuming and prone to human error. To overcome these challenges, deep learning techniques, particularly Convolutional Neural Networks (CNNs), have shown great potential by automatically learning features from medical images and performing accurate classification. This section discusses the process of preparing the dataset, defining the CNN model, and training it to detect different types of brain strokes efficiently.

4.2**Dataset Preparation**:In order to train the Convolutional Neural Network (CNN) efficiently, the dataset was split into three sets:

70% for training – For learning image patterns.

15% for validation – For hyperparameter adjustment and avoiding overfitting.

15% for testing – For testing the performance of the final model on new data

Categories of Images:

- Hemorrhagic Stroke
- Ischemic Stroke
- Normal (No Stroke)

Number of Images Taken:

- Hemorrhagic Stroke: 3556 images
- Ischemic Stroke: 3536 images
- Normal: 3525 images

Sources:

- Kaggle Datasets
- Hospital-provided anonymized dataset

```
/dataset
    /train
        /hemorrhagic
        /ischemic
        /normal
    /test
        /hemorrhagic
        /ischemic
        /normal
    /validation
        /hemorrhagic
        /ischemic
        /normal
```

Fig 4.2 Directory Classification

4.2.2 Dataset Preprocessing Techniques

Table 4.2.2 Preprocessing Technique

Name	**Syntax**	**Description**
Resizing	cv2.resize(image, (256, 256))	Resizes all images to a standard size (256x256 pixels).
Normalization	image/255.0	Scales pixel values between 0 and 1.
Grayscale Conversion	cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)	Converts color images to grayscale to reduce complexity.
Augmentation	ImageDataGenerator()	Applies rotation, zoom, flip to increase dataset variety.

4.2.3 Visualization of Pre-processing and Pre-processed Images

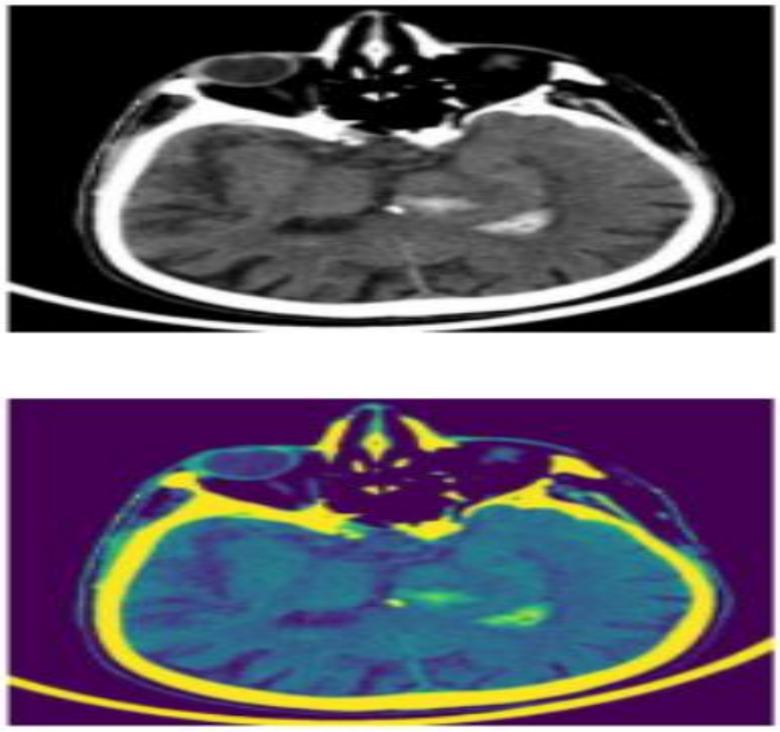


Fig.4.2.3 Original vs Preprocessed Image

4.2.4 Dataset Splitting and Loading:

Dataset Part	Percentage	Number of Images (out of 10,617)
Training Set	70%	7431 Images
Validation Set	15%	1593 Images
Testing Set	15%	1593 Images

Dataset Loading Parameters

Parameter	Value
Batch Size	32
Total Training Batches	7431 ÷ 32 ≈ 232 batches
Total Validation Batches	1593 ÷ 32 ≈ 50 batches
Total Testing Batches	1593 ÷ 32 ≈ 50 batches
Total Images	10,617
Image Size	(256 × 256 × 1)

4.3 Performance Metrics

Metric	Definition	Formula
Precision	Out of the predicted positives, how many are actually positive?	Precision = TP / (TP + FP)
Recall (Sensitivity)	Out of actual positives, how many were correctly predicted?	Recall = TP / (TP + FN)
F1-Score	Harmonic mean of Precision and Recall.	F1 = 2 × (Precision × Recall) / (Precision + Recall)
Accuracy	Correct predictions out of total predictions.	Accuracy = (TP + TN) / (TP + TN + FP + FN)

4.4 Defining CNN Model

1. **Import Required Libraries**:

```
from tensorflow import keras
from keras.models import load_model
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.preprocessing.image import load_img, img_to_array
from tensorflow.keras.models import Sequential
from tensorflow.keras.preprocessing import image_dataset_from_directory
from tensorflow.keras.layers import Conv2D,MaxPooling2D,Flatten,Dense,Dropout,InputLayer
from sklearn.metrics import confusion_matrix, classification_report, accuracy_score, precision_score, recall_score, f1_score
```

2. **Define Conv1 Layer**:

```
Conv2D(64, (3, 3), activation="relu",input_shape=(256,256,1)),
```

3. **Define Conv2 Layer**:

```
Conv2D(64, (3, 3), activation="relu"),
```

4. **Define Max Pooling Layer (after Conv2)**:

```
MaxPooling2D(pool_size=(2, 2)),
```

5. **Define Conv3 Layer**:

```
Conv2D(64, (3, 3), activation="relu"),
```

6. **Define Max Pooling Layer (after Conv3)**:

```
MaxPooling2D(pool_size=(2, 2)),
```

7. **Flatten Layer**:
   Flatten the output of the last convolutional layer to a 1D vector.

   model.add(Flatten())

8. **Define Fully Connected Layer (Dense Layer)**:
   model.add(Dense(3, activation='Softmax'))

9. **Compile the Model**
   model.compile(optimizer=Adam(), loss='categorical cross_crossentropy', metrics=['accuracy'])

10. **Model Summary**:
    Display the architecture of the model.

model.summary()

```
Model: "sequential_4"
```

Layer (type)	Output Shape	Param #
conv2d_8 (Conv2D)	(None, 254, 254, 8)	224
max_pooling2d_8 (MaxPooling2D)	(None, 127, 127, 8)	0
conv2d_9 (Conv2D)	(None, 125, 125, 16)	1,168
max_pooling2d_9 (MaxPooling2D)	(None, 62, 62, 16)	0
flatten_4 (Flatten)	(None, 61504)	0
dense_4 (Dense)	(None, 3)	184,515

```
Total params: 185,907 (726.20 KB)
Trainable params: 185,907 (726.20 KB)
Non-trainable params: 0 (0.00 B)
```

CHAPTER 5

RESULT

AND

DISCUSSION

RESULT AND DISCUSSION

5.1 Training and Validation Performance of Proposed CNN model:.

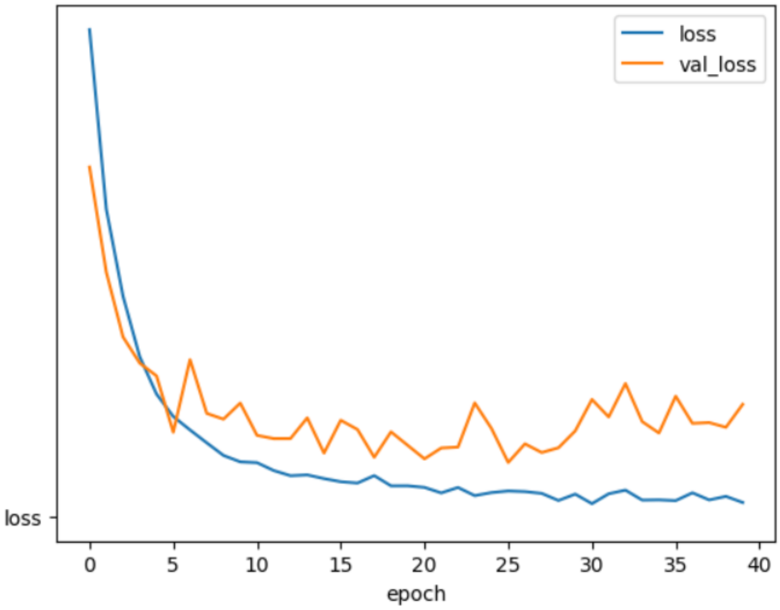
```
50/50 ━━━━━━━━━━━━━━━━━━━━ 13s 263ms/step - accuracy: 0.9731 - loss: 0.1535
Test Loss: 0.13331033289432526
Test Accuracy: 0.9768315553665161
dict_keys(['accuracy', 'loss', 'val_accuracy', 'val_loss'])
```

```
Epoch 1/40
233/233 ━━━━━━━━━━━━━━━━━━━━ 101s 363ms/step - accuracy: 0.6371 - loss: 0.7937 - val_accuracy: 0.8134 - val_loss: 0.4644
Epoch 2/40
233/233 ━━━━━━━━━━━━━━━━━━━━ 69s 298ms/step - accuracy: 0.8071 - loss: 0.4441 - val_accuracy: 0.8687 - val_loss: 0.3250
Epoch 3/40
233/233 ━━━━━━━━━━━━━━━━━━━━ 69s 297ms/step - accuracy: 0.8800 - loss: 0.2917 - val_accuracy: 0.9077 - val_loss: 0.2387
Epoch 4/40
233/233 ━━━━━━━━━━━━━━━━━━━━ 68s 291ms/step - accuracy: 0.9082 - loss: 0.2187 - val_accuracy: 0.9102 - val_loss: 0.2040
Epoch 5/40
233/233 ━━━━━━━━━━━━━━━━━━━━ 67s 288ms/step - accuracy: 0.9391 - loss: 0.1604 - val_accuracy: 0.9253 - val_loss: 0.1870
Epoch 6/40
233/233 ━━━━━━━━━━━━━━━━━━━━ 68s 291ms/step - accuracy: 0.9521 - loss: 0.1334 - val_accuracy: 0.9504 - val_loss: 0.1121
Epoch 7/40
233/233 ━━━━━━━━━━━━━━━━━━━━ 68s 290ms/step - accuracy: 0.9592 - loss: 0.1139 - val_accuracy: 0.9296 - val_loss: 0.2087
Epoch 8/40
233/233 ━━━━━━━━━━━━━━━━━━━━ 82s 290ms/step - accuracy: 0.9595 - loss: 0.1000 - val_accuracy: 0.9491 - val_loss: 0.1372
Epoch 9/40
233/233 ━━━━━━━━━━━━━━━━━━━━ 67s 289ms/step - accuracy: 0.9684 - loss: 0.0812 - val_accuracy: 0.9548 - val_loss: 0.1295
Epoch 10/40
233/233 ━━━━━━━━━━━━━━━━━━━━ 68s 293ms/step - accuracy: 0.9790 - loss: 0.0620 - val_accuracy: 0.9397 - val_loss: 0.1509
Epoch 11/40
233/233 ━━━━━━━━━━━━━━━━━━━━ 81s 289ms/step - accuracy: 0.9718 - loss: 0.0775 - val_accuracy: 0.9686 - val_loss: 0.1080
```

Screenshot 5.1Model Accuracy

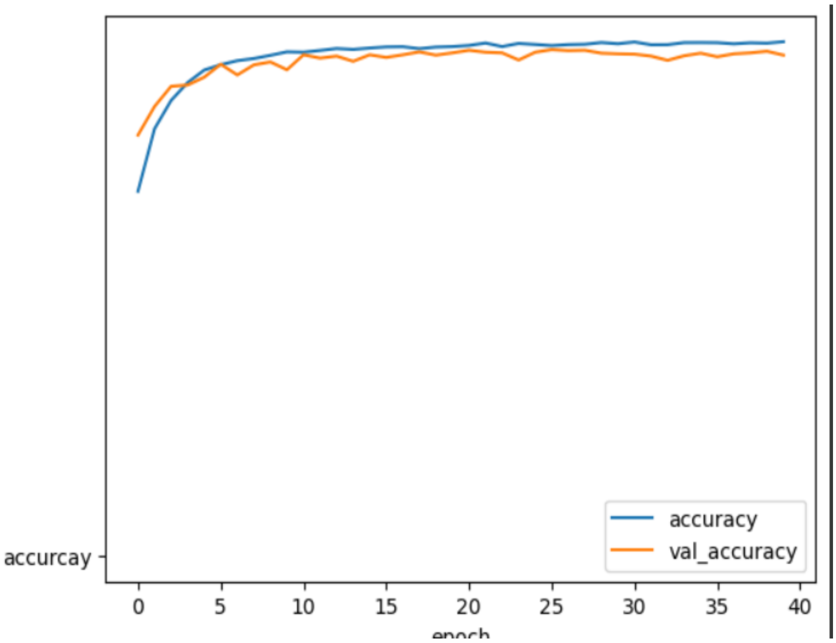
5.2 Loss and Accuracy Graphs:

To evaluate the model's performance during training, graphs were plotted for both training and validation accuracy and loss over each epoch. These graphs help visualize the learning behavior of the model. A steadily decreasing loss and increasing accuracy indicate proper convergence, while any signs of overfitting or underfitting can also be identified through these plots. In this project, the graphs showed consistent



Screenshot 5.2.1 Loss vs Epoch graph

improvement across epochs, confirming that the model was learning effectively without significant overfitting.



Screenshot 5.2.2Accuracy vs Epoch graph

5.3 Confusion Matrix:

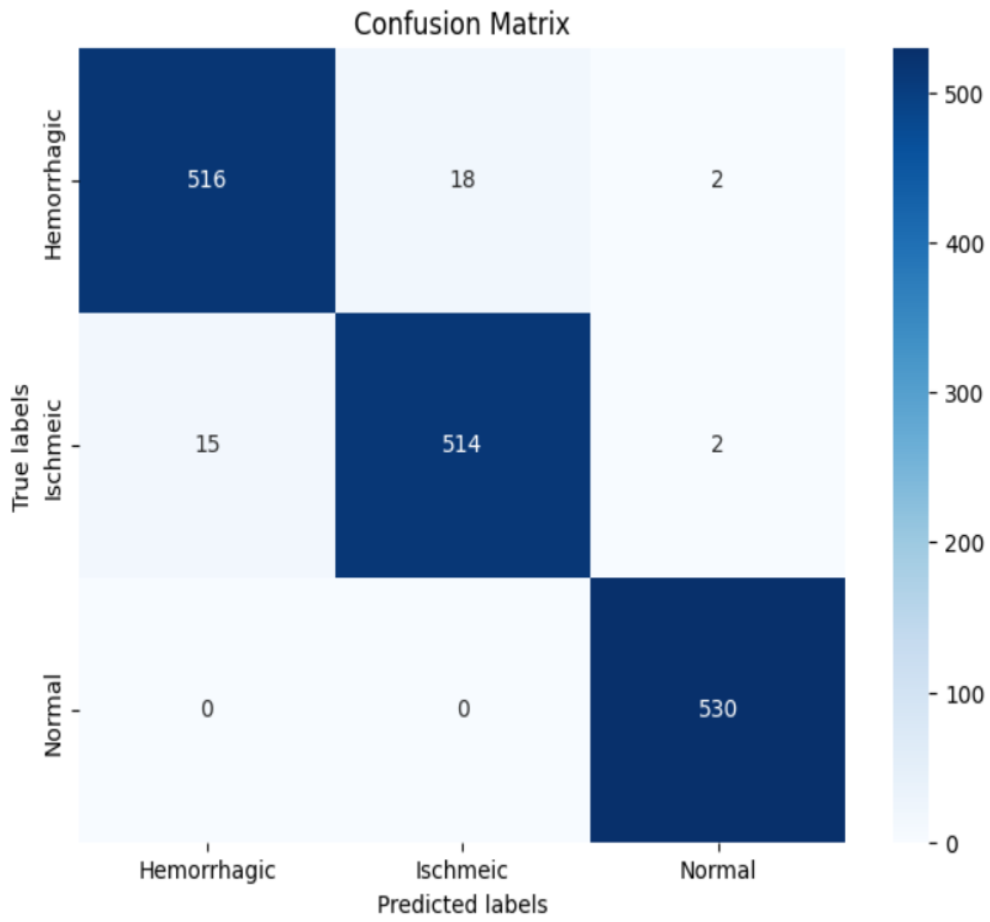
A Confusion Matrix is a performance measurement tool for classification problems. It is especially useful for evaluating the performance of a machine learning model when the output is of two or more classes. In our project, it helps us understand how well the model is able to distinguish between different types of brain strokes (Ischemic and Hemorrhagic).

The confusion matrix is structured as a table that shows:

- **True Positives (TP):** Cases where the model correctly predicted the positive class (e.g., correctly identified an ischemic stroke).
- **True Negatives (TN):** Cases where the model correctly predicted the negative class (e.g., correctly identified a hemorrhagic stroke).
- **False Positives (FP):** Cases where the model incorrectly predicted the positive class (e.g., predicted ischemic stroke when it was actually hemorrhagic).
- **False Negatives (FN):** Cases where the model incorrectly predicted the negative class (e.g., predicted hemorrhagic stroke when it was actually ischemic).

Using these values, we can calculate important performance metrics:

- **Accuracy** = (TP + TN) / (TP + TN + FP + FN)
- **Precision** = TP / (TP + FP)
- **Recall (Sensitivity)** = TP / (TP + FN)
- **F1 Score** = 2 × (Precision × Recall) / (Precision + Recall)



Screenshot 5.4

Table.5.3.1 Classification Report

Class	Precision	Recall	F1-Score	Support
Hemorrhagic	0.97	0.96	0.97	536
Ischemic	0.97	0.97	0.97	531
Normal	0.99	1.00	1.00	530
Accuracy			0.98	1597
Macro Avg	0.98	0.98	0.98	1597
Weighted Avg	0.98	0.98	0.98	1597

Screen Shot 5.5

CHAPTER 6

CONCLUSION

CONCLUSION

6.1 CONCLUSION

This project successfully showcases how deep learning and modern web development frameworks can be integrated to build an effective and interactive brain stroke detection system. The core idea behind the system was to utilize artificial intelligence to assist in the early and accurate detection of stroke types—specifically ischemic and hemorrhagic strokes—from brain scan images. The implementation involved a well-structured pipeline where the CNN model was designed and trained using Keras in Google Colab, benefiting from GPU acceleration and flexible notebook-based experimentation. The trained model was then deployed using FastAPI, which offered a high-performance and scalable backend solution capable of handling prediction requests efficiently.

The ReactJS-based frontend served as a user-friendly interface, allowing users to upload medical images and receive instant feedback in the form of prediction results. The seamless communication between the frontend and backend was achieved through API calls, demonstrating a practical full-stack deployment of a machine learning model. Furthermore, additional features such as the generation of a downloadable PDF report containing the patient's name, predicted stroke type, and possible symptoms added more value and usability to the application, especially in real-world medical settings where documentation is crucial.

This project not only achieved its primary objective but also laid the foundation for future enhancements. Potential improvements include training the model on a larger and more diverse dataset to further increase accuracy and generalization, integrating more advanced features such as heatmaps to visualize areas of interest in the scan, adding user authentication for security, and deploying the application on a cloud platform to allow remote access by healthcare professionals. Moreover, the system can be extended to detect other neurological disorders, making it more comprehensive and impactful.

6.2 FUTURE SCOPE

The system can be further enhanced to include age-group-based predictions. By analyzing stroke patterns across different age groups, the model can offer insights into which stroke types are more prevalent among certain demographics. This feature can help in identifying high-risk patients more effectively and tailoring preventive strategies accordingly. Additionally, the platform can be adapted to include a doctorauthorization module, where medical professionals can review and validate AI-generated predictions before finalizing the diagnosis. This not only ensures clinical accuracy but also builds trust in AI-assisted decision-making within healthcare environments..

REFERENCES

REFERENCES

[1]. G. R. Kumar, P. Vyshnavi, S. Prasanna, T. H. Reddy, C. Charanya, and P. Chandrababu, "Brain Stroke Detection Using Machine Learning", International Journal of Research in Engineering, Science and Management, vol. 5, no. 3, pp. – Mar. 2022.

[2]. B. R. Gaidhani, R. Rajamenakshi, and S. Sonavane, "Brain Stroke Detection Using Convolutional Neural Network and Deep Learning Models" , 2nd International Conference on Intelligent Communication and Computational Techniques (ICCT), Manipal University Jaipur, Sep. 28-29, 2019I. S. Jacobs and C. P. Bean, “Fine particles, thin films and exchange anisotropy,” in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350

[3]. S. Bashir, F. Firdous, S. Z. Rufai, and S. Kumar, "Brain International Conference on Augmented Intelligence and Sustainable Systems (ICAISS), Trichy, India,. IEEE.Issue:22nd Sep 2023 DOI:10.1109/ICREST51555.2021.9331090.

[4]. Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, “Electron spectroscopy studies on magneto-optical media and plastic substrate interface,” IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].

[5]. C.-L. Chin, B.-J. Lin, G.-R. Wu, T.-C. Weng, C.-S. Yang, and R.-C. Su, "An automated early ischemic stroke detection system using CNN deep learning algorithm" ,8th International Conference on Awareness Science and Technology (iCAST), IEEE, 2023..

[6]. Azhar Tursynova, Batyrkhan Omarov, Natalya Tukenova, Indira Salgozha, Onergul Khaaval, Rinat Ramazanov, Bagdat Ospanov, Deep Learning-Enabled Brain Stroke Classification on Computed Tomography Images.

[7]. Zeyad Ghaleb Al-Mekhlafi, Ebrahim Mohammed Senan, Taha H Rassem, Badiea Abdulkarem Mohammed, Nasrin M Makbol, Adwan Alownie Alanazi, Tariq S Almurayziq, Fuad A Ghaleb, Deep learning and machine learning for

early detection of stroke and haemorrhage, Comput. Mater. Contin. 72 (1) (2022) 775–796..

[8]. Suranjan Kumar, Anwesha Sengupta, EEG classification for stroke detection using deep learning networks, in: 2022 2nd International Conference on Emerging Frontiers in Electrical and Electronic Technologies, ICEFEET, IEEE, 2022, pp. 1–6.

[9]. Nouf Saeed Alotaibi, Abdullah Shawan Alotaibi, M Eliazer, Asadi Srinivasulu, Detection of ischemic stroke tissue fate from the MRI images using a deep learning approach, Mob. Inf. Syst. 2022 (2022).

[10]. Enhanced non-contrast computed tomography images for early acute stroke detection using machine learning approach, Expert Syst. Appl. 240 (2024) 122559.

DISSEMINATION OF WORK

Research Paper

International Journal on Science and Technology (IJSAT)

E-ISSN: 2229-7677 • Website: www.ijsat.org • Email: editor@ijsat.org

Deep Learning-Based Multi-Class Stroke Detection Using CNN

V.S. Mahalle[1], Nikhil K. Kulkarni[2], Pratik Kuntawar[3], Niraj Saraf, Gaurav Mahore[4], V.S. Mahalle[5]

[1]Professer, [2,3,4,5]Student, Computer science and engineering, Shri sant Gajanan Maharaj College of engineering

Abstract

Stroke is a critical medical condition caused by an interruption of blood flow to the brain that leads to severe neurological damage or death if not properly diagnosed and treated. Traditional stroke diagnosis employs manual MRI and CT scan examinations, which are subjective, time-consuming, and prone to human error. Previous models were primarily binary classifiers that could only determine the presence or absence of a stroke, which is not adequate for proper treatment[2].To overcome this shortcoming, we developed our own CNN-based model that first detects the presence of a stroke and then classifies it as ischemic stroke, haemorrhagic stroke, or normal cases. This multi-classification is more accurate and provides fine-grained information needed for medical decision-making.Our approach compared to traditional approaches and previous models is more accurate, quicker, and more accurate in stroke classification. With deep learning, our model allows for earlier detection of a stroke, less human error, and better patient outcomes through automatic and accurate diagnosis.

Keywords: Convolutional Neural Network, Brain Stroke, Ischemic Stroke, Hemorrhagic Stroke, Deep Learning

1. Introduction

A cerebrovascular accident, or stroke and brain attack as it is also known, is a condition in which there is a decrease or stoppage of the supply of blood to the brain, thereby removing neuronal cells from the necessary supply of oxygen and nutrients. Due to this deprivation, cell death ensues and is followed by neurological injury. It is one of the leading causes of death worldwide and affects an estimated 15 million people annually with 5.8 million of them dying as a consequence. It is estimated that a stroke will affect one in six people at some time or another during their lifetime.

Cerebral strokes are divided into two major categories:

Ischemic Stroke –

Narrowing or occlusion of the arteries that interrupts blood flow to the cerebral area is the main etiologic factor for this syndrome. This mechanism is responsible for almost 87% of the overall stroke attacks. The blockage can be due to the development of blood clots (thrombosis) or lipid plaque deposition (atherosclerosis). Emergency medical care, like the intravenous administration of thrombolytic drugs

Plagiarism Report

research_paper.pdf

Assignment

Electric

SRM Institute of Science & Technology

Document Details

Submission ID

trn:oid:::1:3176123417

Submission Date

Mar 7, 2025, 9:58 PM GMT+5:30

Download Date

Mar 7, 2025, 10:00 PM GMT+5:30

File Name

research_paper.pdf

File Size

532.8 KB

7 Pages

15% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography

Match Groups

49 Not Cited or Quoted 15%
Matches with neither in-text citation nor quotation marks

0 Missing Quotations 0%
Matches that are still very similar to source material

0 Missing Citation 0%
Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Top Sources

7% Internet sources

12% Publications

4% Submitted works (Student Papers)

PROJECT GROUP DETAILS

	**Name :Nikhil Kiran Kulkarni** **Address: Ujjwal Nagar no 3, Arni Road,Yavatmal ,445001** **Mobile No: 7385917247** **Email ID:mnkkulkarni19@gmail.com**
	**Name :Pratik Pradip Kuntawar** **Address: Ward No 2,At Post Parsapur Tal: AchalpurDist:Amravati, 444806** **Mobile No:7820881097** **Email ID:kuntawarpratik4@gmail.com**
	**Name :Niraj Milind Saraf** **Address: Civil Line, Near Sakalkale Complex, Tal: Khamgaon, Dis: Buldhana, 444303** **Mobile No: 9307651711** **Email ID:nirajsaraf31@gmail.com**
	**Name :Gaurav Umesh Mahore** **Address:Near new ganpati temple, rajapeth, Amaravati, 444606** **Mobile No: 9579509359** **Email ID:gauravum2808@gmail.com**