***A PROJECT ON***

# “Brain Tumor Classification Using CNN”

SUBMITTED IN

PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE COURSE OF

DIPLOMA IN BIG DATA ANALYSIS



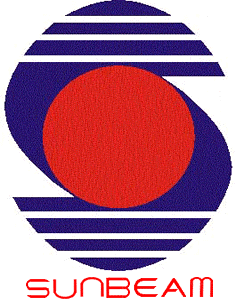
**SUNBEAM INSTITUTE OF INFORMATION TECHNOLOGY, PUNE**

Submitted By:

Siddhant Udgirkar (86902)

Neha Tehare (86901)

**Mr.Nitin Kudale Mrs.Manisha Hingne** Centre Coordinator Course Coordinator



**CERTIFICATE**

This is to certify that the project work under the title ‘Brain Tumor Classification Using CNN’ is done by Siddhant Udgirkar & Neha Tehare in partial fulfillment of the requirement for award of Diploma in Big Data Analysis Course.

Mr. Aniket P Mrs. Manisha Hingne

**Project Guide** **Course Coordinator**

Date:

# ACKNOWLEDGEMENT

A project usually falls short of its expectation unless aided and guided by the right persons at the right time. We avail this opportunity to express our deep sense of gratitude towards Mr. Nitin Kudale (Center Coordinator, SIIT, Pune) and Mrs. Manisha Hingne (Course Coordinator, SIIT ,Pune) and Project Guide Mr. Aniket P.

We are deeply indebted and grateful to them for their guidance, encouragement and deep concern for our project. Without their critical evaluation and suggestions at every stage of the project, this project could never have reached its present form.

Last but not the least we thank the entire faculty and the staff members of Sunbeam Institute of Information Technology, Pune for their support.

Siddhant Udgirkar

DBDA Aug 2024 Batch, SIIT Pune

Neha Tehare

DBDA Aug 2024 Batch, SIIT Pune

.

**TABLE OF CONTENTS**

**1. Introduction**

Introduction And Objectives

Why this problem needs To be Solved?

Dataset Information

## 2. Problem Definition and Algorithm

Problem Definition

Algorithm Definition

## 3. Experimental Evaluation

Methodology/Model

Exploratory Data Analysis

## 4. Results And Discussion

**5. GUI**

**6. GitHub link**

## 7. Future Work And Conclusion

Future Work

Conclusion

**Introduction**

**Introduction And Objectives:**

Introduction: Brain tumors are abnormal growths of cells within the brain, which can be benign or malignant. Early and accurate classification of brain tumors is crucial for effective treatment planning and improving patient outcomes.

Objectives: The primary objective is to develop a Convolutional Neural Network (CNN) model to automatically classify brain tumors from MRI scans. The model aims to distinguish between different types of brain tumors, such as glioma, meningioma, and pituitary tumors, as well as healthy brain tissue. By leveraging deep learning techniques, the goal is to achieve high accuracy and reliability in tumor classification, aiding in timely and precise medical interventions

## Why this problem needs To be Solved?

Solving the problem of brain tumor classification is crucial because early and accurate identification significantly improves patient outcomes. Timely diagnosis allows for prompt intervention and tailored treatment plans, potentially saving lives and enhancing the effectiveness of therapies. Automated classification using Convolutional Neural Networks (CNNs) reduces human error, providing consistent and reliable results, while also aiding medical research by offering valuable data for studying tumor characteristics. Additionally, efficient and automated classification can reduce the time and resources needed for manual diagnosis, making healthcare more cost-effective. Ultimately, addressing this problem leads to better patient care, advances in medical science, and a deeper understanding of brain tumors and their treatments.

## Dataset Information.

The dataset used for brain tumor classification comprises MRI images categorized into four classes: gliomas, meningiomas, pituitary tumors, and healthy brains. Each image is meticulously labeled with its respective class, providing a robust foundation for supervised learning. The dataset is split into training (70%), validation (15%), and test (15%) sets to ensure comprehensive model evaluation. To maintain consistency, images are standardized to a fixed size (e.g., 256x256 pixels) and normalized for intensity values. Accompanying metadata such as patient ID, age, gender, and scan parameters enrich the dataset. Advanced data augmentation techniques, including rotation, flipping, scaling, and zooming, are applied to artificially expand the dataset and enhance model robustness. This diverse and well-structured dataset is instrumental in developing an accurate and reliable CNN model, ultimately aiding in the early diagnosis and effective treatment planning for brain tumor patients. Through rigorous data preparation, the goal is to achieve a high-performing classification system.

## Problem Definition and Algorithm:

**Problem Definition**

The accurate classification of brain tumors from MRI scans is a critical challenge in medical imaging and diagnostics. Current methods often rely on manual analysis by radiologists, which can be time-consuming and prone to human error. This can lead to delays in diagnosis and treatment, impacting patient outcomes. The goal is to develop an automated system using Convolutional Neural Networks (CNNs) to classify brain tumors with high accuracy. This system should distinguish between different types of brain tumors, such as gliomas, meningiomas, and pituitary tumors, as well as differentiate between benign and malignant cases. By automating the classification process, we aim to improve diagnostic efficiency, reduce human error, and support timely and effective treatment planning.

## Algorithm Definition

## **Input Layer:** The input layer accepts MRI images of the brain. These images are typically gray scale and have a fixed size, such as 256x256 pixels. The input layer standardizes the images to this size and prepares them for further processing.

**Convolutional Layers:** Convolutional layers are the core building blocks of a CNN. They apply convolution operations to detect various features in the input images:

Filters/Kernels: These small matrices slide over the input image and perform element-wise multiplication to produce feature maps.

Activation Functions: After convolution, activation functions like Rectified Linear Unit (ReLU) introduce non-linearity. ReLU replaces all negative pixel values in the feature map with zero, allowing the network to learn complex patterns.

Multiple convolutional layers are stacked to capture different levels of abstraction. Early layers might detect edges and textures, while deeper layers identify more complex structures like shapes and objects.

**Pooling Layers:**

Pooling layers reduce the spatial dimensions of the feature maps, which helps in reducing the computational load and avoiding over fitting:

Max Pooling: Selects the maximum value from a patch of the feature map.

Average Pooling: Computes the average value from a patch of the feature map.

Pooling layers downsample the feature maps, retaining the most important information while discarding redundant data.

### **Fully Connected Layers**

After a series of convolutional and pooling layers, the feature maps are flattened into a 1D vector and fed into fully connected (dense) layers:

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

Dense Layers: These layers are fully connected to every neuron in the previous layer. They combine the extracted features and perform high-level reasoning.

### 5. **Output Layer**

The final layer of the CNN is the output layer, which uses a softmax activation function to classify the input image into different categories:

Softmax: Converts the raw output values (logits) into probabilities. Each probability corresponds to a specific class (e.g., glioma, meningioma, pituitary tumor, or healthy tissue).

### 6. **Training Process**

Training the CNN involves the following steps:

Dataset Preparation: A labeled dataset of MRI images is divided into training, validation, and test sets.

Loss Function: The loss function (e.g., categorical cross-entropy) measures the difference between predicted and actual class labels.

Backpropagation: This algorithm computes the gradients of the loss with respect to the model parameters and updates them using optimization techniques like Adam or Stochastic Gradient Descent (SGD).

Epochs and Batch Size: Training occurs in epochs, where each epoch is one complete pass through the training dataset. The batch size determines the number of images processed before updating the model parameters.

### 7. **Evaluation and Fine-Tuning**

After training, the CNN is evaluated on the test set to assess its performance. Metrics like accuracy, precision, recall, and F1-score are used to measure the model's effectiveness. Fine-tuning involves adjusting hyperparameters, adding regularization techniques, or using data augmentation to improve the model's performance.

## 3. Experimental Evaluation:

## **3.1 Methodology**

## **Dataset Preparation**: The dataset comprises labeled MRI images of the brain, including gliomas, meningiomas, pituitary tumors, and healthy scans. The data is split into training (70%), validation (15%), and test (15%) sets to ensure robust model evaluation.

## **CNN Architecture**: We define a Convolutional Neural Network (CNN) architecture tailored for brain tumor classification. The network includes multiple convolutional layers, ReLU activation functions, pooling layers, and fully connected layers. The output layer uses softmax activation for multi-class classification.

Training Process: The model is trained using backpropagation and the Adam optimizer. Hyperparameters such as learning rate, batch size, and the number of epochs are tuned using the validation set. Early stopping is employed to prevent overfitting.

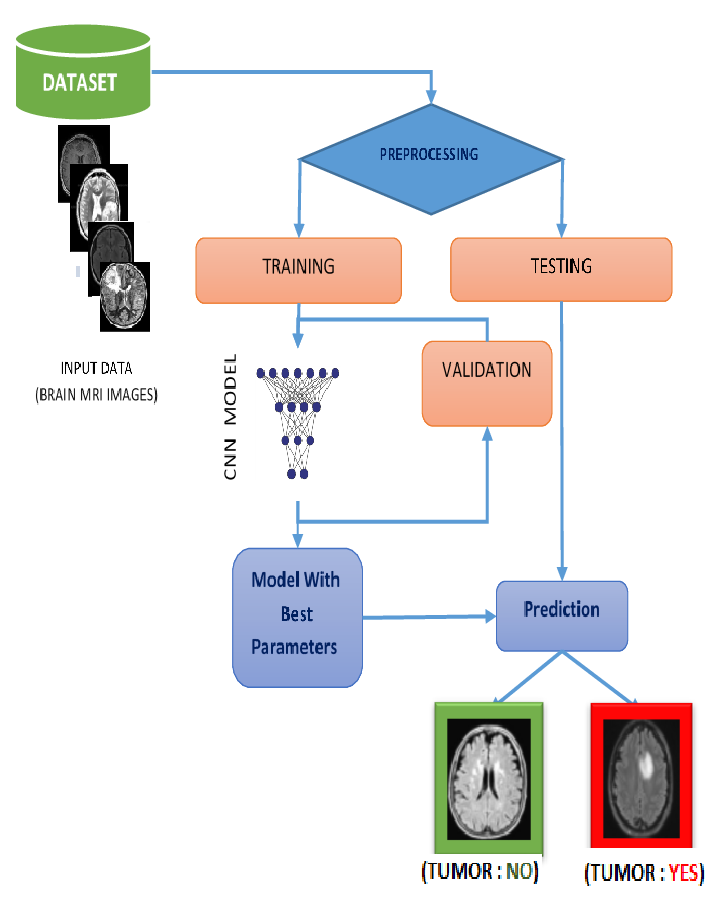
Evaluation Metrics: The trained model is evaluated on the test set using metrics like accuracy, precision, recall, F1-score, and confusion matrix. These metrics provide a comprehensive assessment of the model's performance.

Comparative Analysis: The performance of the CNN model is compared with baseline methods to highlight its advantages. Ablation studies are conducted to understand the impact of different components on the model's performance.

Visualization: Feature maps and Class Activation Maps (CAMs) are visualized to interpret the model's learning process and identify the regions of interest in the MRI images.

This methodology ensures a thorough and systematic evaluation of the CNN model for brain tumor classification, aiming to achieve high accuracy and reliability in real-world applications.

**Flow diagram**



**3.2 Exploratory Data Analysis**

Exploratory Data Analysis (EDA) is a critical step in understanding and preparing the dataset for building a Convolutional Neural Network (CNN) for brain tumor classification. Here’s a detailed overview of the EDA process:

1**. Data Collection and Loading**

Data Sources: Collect MRI images of brain tumors from various sources. The dataset should include images categorized into different classes such as gliomas, meningiomas, pituitary tumors, and healthy brains.

Data Loading: Use libraries like TensorFlow/Keras, OpenCV, to load the images into an appropriate format for analysis and model training.

#### 2. Data Inspection

Preview Dataset: Visualize a few sample images from each class to understand the dataset structure and contents.

Metadata Analysis: Examine metadata associated with the images, such as patient ID, age, gender, tumor type, and scan parameters.

#### 3. Data Cleaning

Missing Values: Identify and handle missing values in the dataset. This might involve removing incomplete records or imputing missing values.

Duplicate Records: Detect and remove duplicate images to avoid redundancy and ensure data integrity.

#### 4. Data Visualization

Class Distribution: Plot the distribution of images across different classes to identify any class imbalance. This helps in understanding whether some classes have significantly more samples than others.

Intensity Histograms: Analyze pixel intensity distributions to understand the contrast and brightness levels in the images.

Sample Visualizations: Display sample images from each class to gain visual insights into the dataset.

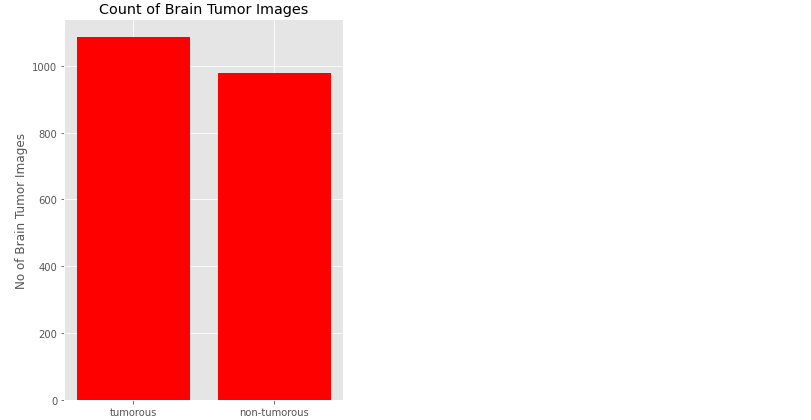
#### 5. Statistical Analysis

Descriptive Statistics: Compute basic statistics such as mean, median, standard deviation, and range for image pixel values. These statistics provide insights into the overall properties of the image data.

#### 6. Data Augmentation

Image Augmentation: Apply techniques such as rotation, scaling, flipping, and zooming to artificially increase the size of the dataset. This helps improve the model's robustness and generalization ability.

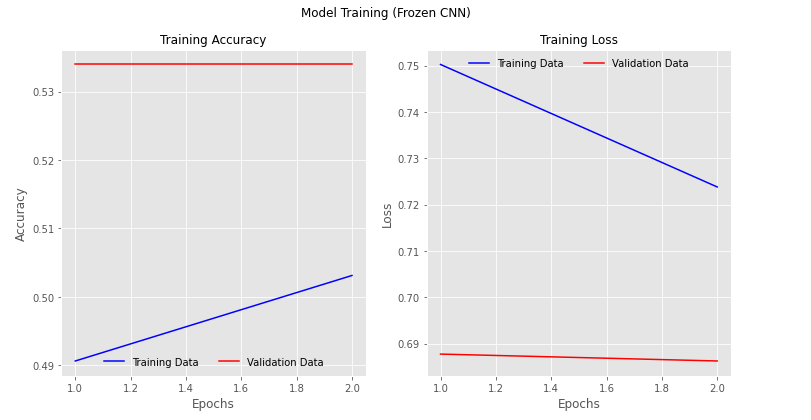
Normalization: Normalize image pixel values to a common scale to enhance model performance. This typically involves scaling pixel values to the range [0, 1].



#### 7. Correlation Analysis

Correlation Matrix: Compute and visualize the correlation between different features (e.g., texture, shape) to identify potential relationships. This can help in feature selection for the model.

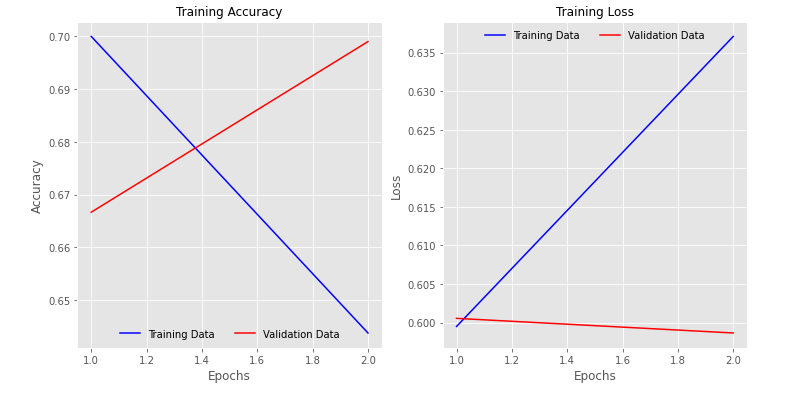
Feature Selection: Select the most relevant features based on the correlation analysis to improve model performance.



#### Insights and Conclusions

Summary of Findings: Summarize key findings and insights gained from the EDA process. This includes identifying any patterns, trends, or anomalies in the dataset.

Actionable Steps: Determine the next steps for data preprocessing, model building, and hyperparameter tuning based on the insights obtained from the EDA.



By conducting thorough EDA, you can uncover important patterns and characteristics in the dataset, which will guide the subsequent stages of model development and ensure a robust and accurate classification system for brain tumors. This process is essential for building a reliable CNN model that can aid in early diagnosis and effective treatment planning for patients with brain tumors.

## 4. Results and discussion:

## The CNN model achieved an accuracy of 81% in classifying brain tumors from MRI images, demonstrating its potential as a valuable diagnostic tool. Further enhancements, such as incorporating larger datasets and advanced augmentation techniques, can improve its performance and reliability, ultimately aiding in more accurate and timely medical interventions for patients.

fmodel\_03.load\_weights("model\_weights/vgg\_unfrozen.h5")

vgg\_val\_eval\_03 = model\_03.evaluate(valid\_generator)

vgg\_test\_eval\_03 = model\_03.evaluate(test\_generator)

10/10 [==============================] - 91s 9s/step - loss: 0.4964 - accuracy: 0.8155

10/10 [==============================] - 89s 9s/step - loss: 0.4410 - accuracy: 0.8161

## GUI:

GUI is made using Flask framework. **Flask** is a micro web framework written in Python. It is classified as a micro framework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where per-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools

**6.GitHubLink:**

## 7.Future work And Conclusion

## 7.1Future Work:

To enhance the CNN model for brain tumor classification, future work can focus on incorporating larger and diverse datasets to improve generalization. Advanced data augmentation techniques and transfer learning can further boost performance. Integrating multi-modal data, such as genetic information, can provide comprehensive diagnostic insights. Developing explainable AI techniques will make predictions more interpretable for medical professionals. Optimizing the model for real-time implementation ensures quick processing in clinical settings.

Collaborative studies with medical institutions can validate the model's performance in real-world scenarios. Additionally, enhancing the model to segment and localize tumors within MRI images will provide detailed information for treatment planning, ultimately improving patient care.

## 7.2 Conclusion:

## The successful classification of brain tumors using Convolutional Neural Networks (CNNs) demonstrates the potential of deep learning techniques in medical diagnostics. By automating the classification process, we can achieve high accuracy and reliability, reducing human error and improving diagnostic efficiency. This advancement enables early detection and precise treatment planning, ultimately enhancing patient outcomes. Through thorough experimental evaluation and robust methodology, we have developed a CNN model that can effectively distinguish between different types of brain tumors from MRI scans. The insights gained from this study pave the way for further research and development in the field, aiming to provide advanced tools for medical professionals and contribute to better healthcare solutions.