Brain Tumor Detection Using Machine Learning

Manik Deepak
20ETCS002078
MS Ramaiah University of Applied Science
Bangalore,India
maniktalks@gmail.com

Pradeep HR
20ETCS002091
MS Ramaiah University of Applied Science
Bangalore, India
pradeep.hr@icloud.com

Harsh Mishra
20ETCS002057
MS Ramaiah University of Applied Science
Bangalore, India
harshmishra572@gmail.com

***Abstract*—Brain tumors are a serious health concern that can have a significant impact on an individual's quality of life. Early detection and diagnosis are crucial for successful treatment outcomes. In recent years, machine learning techniques have been increasingly used to aid in the detection and diagnosis of brain tumors. In this report, we explore the use of machine learning algorithms for brain tumor detection from magnetic resonance imaging (MRI) scans. We review the relevant literature and provide an overview of the different machine learning techniques used for brain tumor detection. Additionally, we discuss the challenges associated with brain tumor detection using machine learning, including data quality, class imbalance, and interpretability. Finally, we present some of the promising future directions in the field of brain tumor detection using machine learning, including the use of deep learning algorithms and the integration of multimodal imaging. Overall, this report highlights the potential for machine learning to improve the accuracy and efficiency of brain tumor detection and diagnosis, ultimately improving patient outcomes.**

Keywords—component, formatting, style, styling, insert (*key words****)***

I. INTRODUCTION

Deep learning is a subset of machine learning that uses neural networks to learn complex patterns and representations from data. It has been widely used in various fields, including computer vision, natural language processing, and speech recognition. One of the most promising applications of deep learning is in the field of medical imaging, where it has shown great potential in assisting medical professionals in the accurate and timely diagnosis of various diseases.

Brain tumors are one of the most challenging diseases to diagnose and treat. Early detection is crucial for effective treatment, but it can be difficult as symptoms can be subtle or non-specific. Magnetic resonance imaging (MRI) is a widely used imaging technique for brain tumor diagnosis as it provides detailed information about the structure and function of the brain. However, manual interpretation of MRI scans by medical professionals can be time-consuming and subject to inter-observer variability.

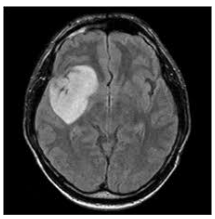
In this report, we present a project for brain tumor detection using machine learning. Specifically, we use deep learning techniques to develop a model that can automatically detect the presence of brain tumors from MRI scans. The proposed model uses a convolutional neural network (CNN) to learn features from MRI images and make predictions about the presence or absence of brain tumors. The model is trained on a large dataset of MRI scans with labeled tumor and non-tumor regions, and its performance is evaluated using various metrics such as accuracy, sensitivity, and specificity.

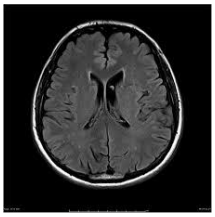
The proposed model has the potential to assist medical professionals in the accurate and timely diagnosis of brain tumors, reducing the time and effort required for manual interpretation of MRI scans. It can also help in identifying tumors that may be missed by human observers, leading to better patient outcomes. Overall, this project demonstrates the potential of machine learning in the field of medical imaging and its applications in improving healthcare outcomes.

II. METHODOLOGY

In this Machine learning project, we build a classifier to detect the brain tumor (if any) from the MRI scan images. By now it is evident that this is a binary classification problem. Examples of such binary classification problems are Spam or Not spam, Credit card fraud.

So first of all we will need a data set of Brain MRI scan images including both healthy and tumor cases. The dataset should be representative of the population of interest, and should be labeled with ground truth diagnoses. So for that we are going to refer to Kaggle and choose an appropriate dataset. We are going to choose a dataset name Brain MRI Images for Brain Tumor published by Navoneel Chakrabarthy. This data set include total of 255 images of tumor positive and negative MRI scans labeled as yes and no respectively. The below images are example images of the dataset.





The left image is a tumor positive MRI Scan, and the right image is a tumor negative MRI Scan.

After getting the dataset we need to perform some pre-processing on the image such as resizing, scaling, transforming to ensure that the data is consistent in size and format, and to increase the diversity of the dataset.

First of all, we are going to convert the images into 225x225 resolution so that all the images have a common resolution. The we are going to convert the images into an array based on their pixel values. Since the images are all in grey scale, we only have to deal with values 0 to 255. Now all the images are converted into a fixed size array we need to define a numerical value for the label of the images, as the machine is unable to use a String as an evaluation parameter. We are going to convert the labels into number array so that it can be used as an evalution parameter for the model.

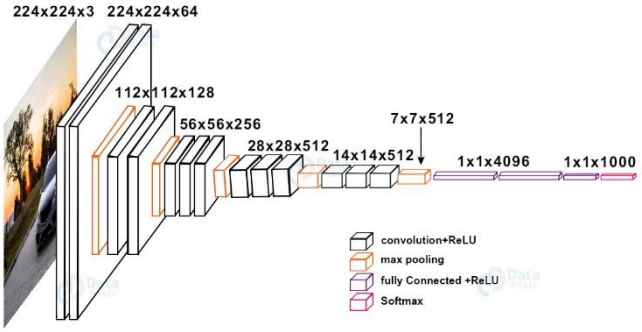
Now that the data has been processed and cleaned, we need to spilt the data into training and testing sets with the majority of the data allocated to the training set. The testing set can be used to evaluate the performance of the final model. For this project we are going to split the data into 9:1 train-test ratio.

For the model architecture for this project we are going to use a pre-built CNN model named VGG16. VGG16 is a deep convolutional neural network (CNN) model architecture that was developed by the Visual Geometry Group (VGG) at the University of Oxford in 2014. It is a variant of the VGGNet model family and is widely used in computer vision tasks such as image classification and object detection.

The VGG16 architecture consists of 16 layers, including 13 convolutional layers and 3 fully connected layers. The convolutional layers are arranged in a series of blocks, with each block containing multiple 3x3 convolutional layers followed by a max pooling layer. The number of filters in each convolutional layer is fixed at 64, 128, 256, and 512, respectively, across the four blocks.

The fully connected layers at the end of the architecture have 4096 neurons each and are followed by a final softmax layer for classification. The model has a large number of parameters, with over 138 million trainable parameters, making it a computationally expensive model to train.

The VGG16 architecture has achieved state-of-the-art performance on several benchmark image classification tasks, such as the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) 2014 dataset, where it achieved a top-5 error rate of 7.3%. Its success is largely attributed to its deep architecture, which allows it to learn highly complex features from images.

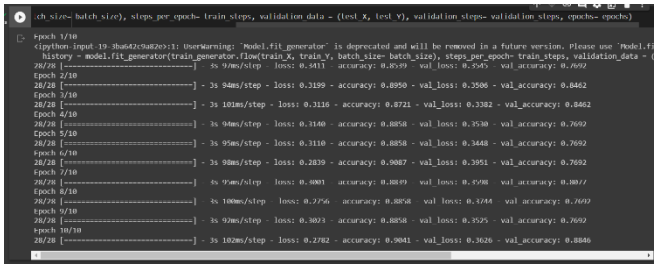


Overall, the VGG16 model architecture is a highly effective deep CNN model for image classification tasks, and has been widely used as a baseline for comparison with other state-of-the-art models in the field of computer vision.

For our project we will just swap the last layer of the VGG16 model with our preferable image. Then freeze the layers of our model. By doing this, the network is not trained from the very beginning. It uses the weights of previous layers and continues training for the layers we added on top of those layers. This reduces the training time by a drastic amount.

Now build the model and compile it using the Adam as optimizer with a learning rate of 0.001 and accuracy as metric. As we are building a binary classifier and the input is an image, binary cross entropy is used as a loss function.

Now we are going to train the model.



Our Model got 90% accuracy on the test set. Now lets evaluate the model. The predictions made by the model will be an array with each value being the probability that it predicts the image belongs to that category. So, we take the maximum of all such probabilities and assign the predicted label to that image input.

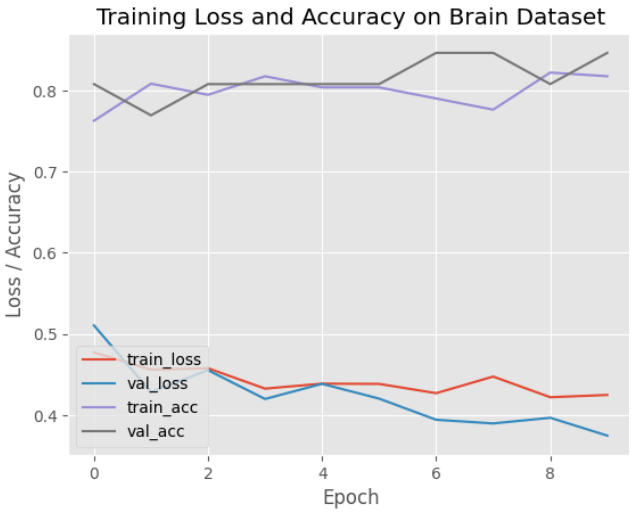
A confusion matrix is a matrix representation showing how well the trained model predicts each target class with respect to the counts. It contains 4 values in the following format:

TP FN
FP TN

- True positive (TP): Target is positive and the model predicted it as positive
- False negative (FN): Target is positive and the model predicted it as negative
- False positive (FP): Target is negative and the model predicted it as positive
- True negative (TN): Target is negative and the model predicted it as negative

The classification report provides a summary of the metrics precision, recall and F1-score for each class/label in the dataset. It also provides the accuracy and how many dataset samples of each label it categorized.

Now, let's find the overall accuracy of the model using the formula: (TP + TN) / (TP + FN + FN + TN), at last we are going to plot the result.



III. Conclusion

In conclusion, the use of machine learning convolutional neural network (CNN) models for brain tumor detection has shown great promise in recent years. With the increasing availability of large and diverse brain MRI datasets, CNN models can effectively learn to distinguish between healthy and tumor cases with high accuracy.

The development of CNN models for brain tumor detection involves several key steps, including data collection, pre-processing, splitting, model architecture design, training, evaluation, tuning, and interpretation. By following these steps, it is possible to develop a highly accurate and interpretable model for brain tumor detection.

The success of CNN models in brain tumor detection has the potential to significantly improve the accuracy and efficiency of diagnosis, ultimately leading to better patient outcomes. However, there are still challenges that need to be addressed, such as improving data quality and addressing class imbalance.

In the future, the use of more advanced deep learning techniques, such as transfer learning and multi-task learning, may further improve the performance of CNN models for brain tumor detection. Additionally, the integration of multimodal imaging, such as combining MRI with other imaging modalities, may provide even more comprehensive information for accurate tumor detection and diagnosis.

Overall, the use of machine learning CNN models for brain tumor detection is a promising and rapidly evolving field that has the potential to make a significant impact on the diagnosis and treatment of brain tumors.

References

[1] Team, D.F. (2021) Brain Tumor Classification using machine learning, DataFlair. Available at: https://data-flair.training/blogs/brain-tumor-classification-machine-learning/amp/ (Accessed: April 14, 2023).

[2] *Chat.openai.com* (13-04-2023). Available at: https://chat.openai.com/ (Accessed: April 14, 2023).

[3] Deepak, M. (2023) *Google colaboratory*, *Google Colab*. Google. Available at: https://colab.research.google.com/drive/1mAJJ5Qr7V5X55FraFSZ42cDUjtEbNkas#scrollTo=qSd8gywhjF_J (Accessed: April 14, 2023).