

ENHANCING BRAIN TUMOR DETECTION USING MRI WITH K-FOLD CROSS-VALIDATION

¹Purvi Ravikumar Singh, ²Sunita Jena and ³Niloufer Kotwal¹Student and ²Assistant Professor, MSc. Big Data Analytics Department, Jai Hind College, Mumbai³Assistant Professor, Life Science Department, Jai Hind College, Mumbai**ABSTRACT**

A brain tumor, which happens when abnormal brain cells grow quickly, is a serious health risk for adults because it can lead to major organ problems and even be life-threatening. Manual segmentation of tumors from brain MRI is time-consuming and error-prone. Early detection allows doctors to get involved before extreme harm, bringing down permanent damage. In this paper, we will see the detection of tumors in Magnetic Resonance Imaging (MRI) where the Deep Learning model will be trained to detect whether the tumor is present. The dataset used in this study is sourced from Kaggle and contains two classes yes and no for brain tumor presence or absence. The tools and technologies used are Keras for developing the neural networks, scikit-learn used in data splitting and TensorFlow for managing data. Also, various libraries are used for image processing work, namely NumPy, OpenCV (cv2), etc. The model's performance is evaluated using an average of accuracy and precision as the primary metric. We aim to achieve accurate tumor detection by enhancing the model using K-fold cross-validation by converting the binary classification problem into categorical classification problem. We have trained the model using categorical cross-entropy for a binary classification problem, where the target variable has been converted into categorical format. The Saved model is used to predict the previously unseen data.

Keywords— Deep Learning CNN, K-fold Crossvalidation, Magnetic Resonance Imaging (MRI), Brain Tumors.

I. INTRODUCTION

In India, tumors like brain tumor rank as the second most common cancer among children and young adults. Ignoring early symptoms delays medical attention, risking tumor progression. Recognizing signs promptly is crucial to increasing survival chances. To understand the function of imaging, it is necessary to first understand what brain tumors are. In the brain, these growths happen when cells proliferate out of control, forming dense bulk. While malignant tumors pose serious health risks, benign tumors only cause minor harm. The capacity of MRI to identify brain tumors with measurable precision is one of its strongest points. It gives doctors high-resolution images that help them identify the location, size and the nature of tumor [1]. Detecting tumor as soon as possible allows doctors to correct things before extreme harm, bringing down the permanent damage risk. Magnetic Resonance Imaging of brains are difficult for brain tumor to be detected by humans manually. Convolutional Neural Networks (CNNs): The method used in this study is Convolutional Neural Networks. CNNs are designed to automatically learn spatial hierarchies of features from images, allowing them particularly effective for analyzing MRI scans of the brain. This method let us create the automated detection of tumors, significantly improving diagnostic accuracy when compared to traditional methods [2]. Treatments for brain tumor mainly depend upon the correct diagnosis and could be time-consuming as well as painful. Brain tumors are a major health concern, being a top cause of death globally. The survival rate for adults diagnosed with brain cancer is alarmingly low, with only 12% surviving beyond five years. This highlights the urgent need for effective diagnostic tools in the medical field [3].

II. LITERATURE REVIEW

The model was trained on three distinct brain MRI datasets from the Kaggle website, displaying its robustness and generalization capabilities across various data sources. The optimized CNN model achieved outstanding performance scores, with an average accuracy, precision, recall, and F1-score of 97% [3]. In this research [4] two machine learning based tumor detection systems were suggested and compared. MLP gives more accuracy and requires more time to build the model. Naïve bayes takes less time and produces less accurate model.

From the paper [5] MRI images of brain tumors don't clearly show where exactly the tumor is located. To find the tumor's exact position in the MRI images, they have used techniques like preprocessing, segmentation, morphological operations, and subtraction. These methods help create the tumor's specific shape in the MRI image, making it possible to accurately detect the brain tumor. From the paper [6] The classification results show whether the brain images are normal or have a tumor. A method called CNN, which uses layers to process data, is used for this classification in Python. they extracted features like depth, width, and height from the images. In this paper they have used gradient descent optimizer. The training accuracy here is 97.5%. They mentioned that losses were less but not in quantitative manner.

The paper [7] presents a completely automatic approach for classifying brain tumors using deep transfer learning to extract characteristics from MRI images. It provides greater classification accuracy compared to existing approaches and demonstrates robustness with limited training samples. However, challenges remain, including the standalone performance of the transfer model, misclassification of meningioma samples, and overfitting with small datasets. Future research should focus on addressing these issues through data augmentation and further model tuning.

III. MATERIALS AND METHODOLOGY

The dataset used in study is sourced from Kaggle, an open source. The dataset is very small having Brain MRI's with Yes (155 Images) and No (98 Images) separated via folders. The brain tumor images went through several steps to prepare them for classification tasks. Here's a simple overview of what was done:

A. Preprocessing

The colorful RGB images were changed to grayscale, which means they became black and white. This reduced complexity and made processing easier. All the images were resized to a standard size of 120x120 pixels. This ensured that every image was of the same size, that is making the next processing steps more consistent and straightforward. Since the image size affects computational load, memory usage, speed and efficiency. The processed images were stored as arrays and corresponding labels were appended, categorizing them as either "tumor" or "no tumor." Finally, the dataset was organized for further analysis. The target variables are converted to categorical format. It is equally important to standardize the image before training even sets foot on the pre-processing phase. This is done on standard procedure where images are normalized in order to increase the stability of the model and enhance efficiency. Image normalization includes standardizing the pixel values of the image thus removing the impact of intensity as well as contrast image amendment.

Data augmentation is a procedure of increasing the size of any given dataset by applying operations to the current data to enable independent transformations. This is especially so when working with small samples where the idea assists in increasing its ability to generalize by avoiding over-fitting. By generating multiple instances from the original data, including rotating, flipping, scaling, cropping, or adding noise to images, or using synonyms or back translations of words, we are able to provide the man to wider examples hence improving the results of the model.

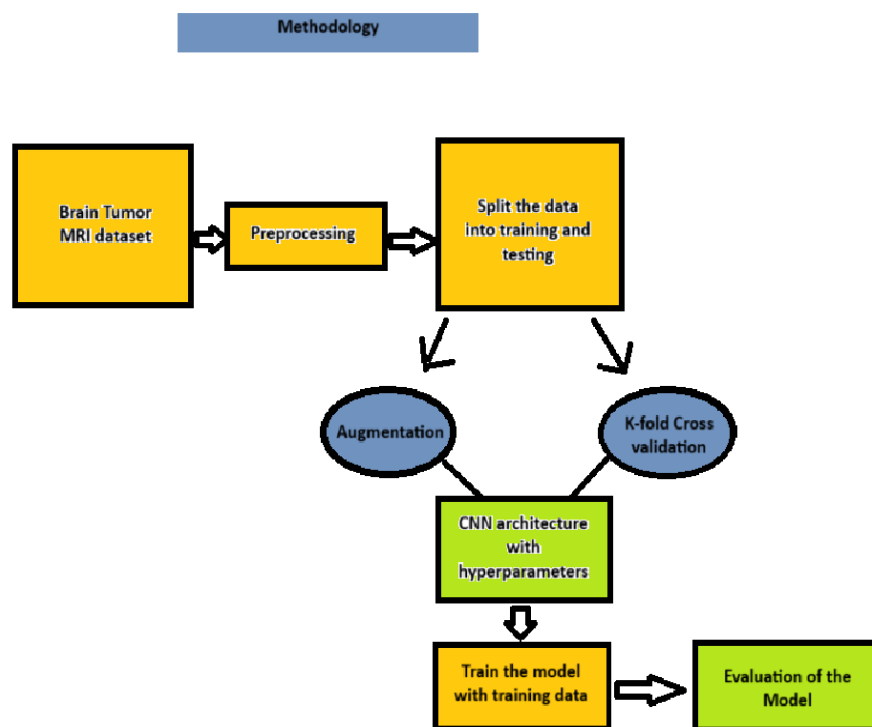


Fig 1. Flow chart of Methodology

B. Splitting

The processed dataset was split into training and testing subsets. The testing size taken as 25% and the training size was 75%. This ratio is general and go with almost all the dataset hence, was taken for splitting.

C. Architecture of the model

CNNs down-sample data using pooling layers, which lower dimensionality while preserving significant features. This improves the efficiency of computing. Pooling is frequently absent from traditional neural networks. CNN are made especially to handle grid-like information, like pictures, where the spatial correlations between pixels are very important. For sequential data, like time series or natural language, other neural network types, such as Recurrent Neural Networks (RNNs), are more appropriate. Weight or parameter Sharing is used to help decrease the number of parameters and enable the model to acquire translation-invariant features, CNNs apply the same filter (or weights) to several regions of the input image. Fully connected networks, on the other hand, assign distinct weights to each connection.

The loss function is approximated using the gradient descent approach. A scoring function is used to translate a raw image pixel into class scores. The degree to which the induced scores correspond to the ground truth labels defines the loss function, which evaluates the quality of a particular collection of parameters. In order to increase accuracy, it is crucial to compute the loss function when the accuracy is low and the loss function is high; in the opposite case, the accuracy is high. The gradient descent algorithm is computed by taking the derivative of loss function. Once more, assess the gradient to find the gradient of the loss function.

Normalization technique is mostly used on algorithms that re machine learning or deep learning based. It insures that all the images are having similar intensity. This technique improves the performance of the algorithm that works better with normalized data. This is used on our data and an image of MRI before and after normalization has been attached as shown in Fig. 3 and Fig. 4.

A Convolutional Neural Network model has been used that comprises of the max pooling layers of a 2x2 filter come after each of the four layers of convolutional. Then following each convolutional layer, ReLu activations are also applied. In the preprocessing stage, a sequential deep learning model was constructed using Keras. Initially, convolutional and pooling layers were added to take properties out of the images, followed by flattening the output to transition into fully connected layers. The model included activation functions like ReLU and softmax for non-linearity and classification, respectively.

K-fold cross-validation is one of the most popular statistical methods for assessing the effectiveness of a learned model. It means the data available is equally divided into K sets or folds more commonly known as the resampling subsets. The proposed model of the study is trained and tested 5 times, where in each test, one fold is used for testing while others are used for training. This way, the final result obtained for each image is an average of results that we get after 5 iterations hence providing us a better approximation of the model.

For binary classification, use binary cross entropy as loss and sigmoid activation function, similarly for categorical classification, we can use categorical cross entropy as loss function and the softmax activation function. Compiled the model with categorical cross entropy as loss, accuracy metric for evaluation and Adam optimizer for parameter optimization. Selecting hyperparameter is a crucial task which has no formula or rule. It can be determined by running and observing the performance of the model several times. As we know each and every parameter has different impact while training the model like speed and accuracy.

EVALUATION METRICS THAT WERE USED ARE GIVEN AS FOLLOWS

- 1) **Accuracy:** Accuracy quantifies how frequently the classifier made accurate predictions. The ratio of the number of accurate predictions to the total number of predictions is a way to define accuracy. Indicates the overall correctness of the model, the percentage of accurate forecasts among all the predictions. A higher level of precision means that the model's predictions are more in line with the labels that are actual labels.

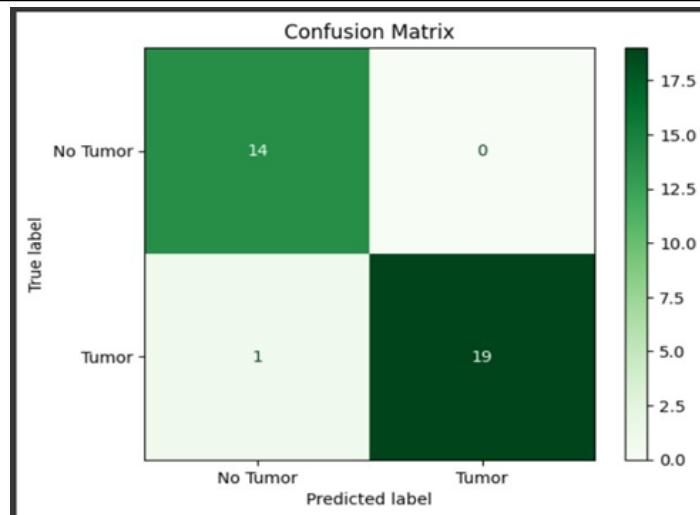


Fig. 2 Confusion matrix of proposed model

- 2) **Precision:** In machine learning, precision is a performance metric that calculates the accuracy of a model in identifying instances among the total instances it predicted as positive. It is especially useful where the cost of false positives is high. It answers the question: Out of all the instances the model classified as positive, how many are actually positive?
- 3) **Recall:** It is also known as "The true positive rate", or the amount of all actual positives that were appropriately categorized as positives. It evaluates the model's ability to record all true positive cases; a high recall denotes fewer false negatives.
- 4) **F1 score:** The F1 score is a statistical metric used to evaluate the accuracy of a model, particularly in classification tasks. It's an alternative to accuracy, which considers overall performance. It Provides a balance between precision and recall, making it especially useful when the data is imbalanced.

Table I Evaluation Metrics and Formulae

Metrics	Formula
Accuracy	$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$
Precision	$Precision = \frac{TP}{TP + FP}$
Recall	$Recall = \frac{TP}{TP + FN}$
F1 score	$F1 = \frac{2 * (precision * recall)}{(precision + recall)}$

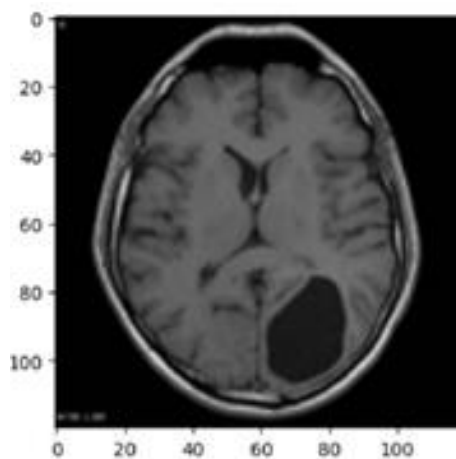


Fig. 3 MRI before normalization

D. Confusion matrix

A confusion matrix is a table that summarizes the performance of a classification model by showing the relationships between actual labels and predicted labels. It is a powerful tool to evaluate classification algorithms, especially for binary and multi-class classification. The matrix contains four main components shown in Table 1:

- 1) **True Positives:** Accurately predicted cases

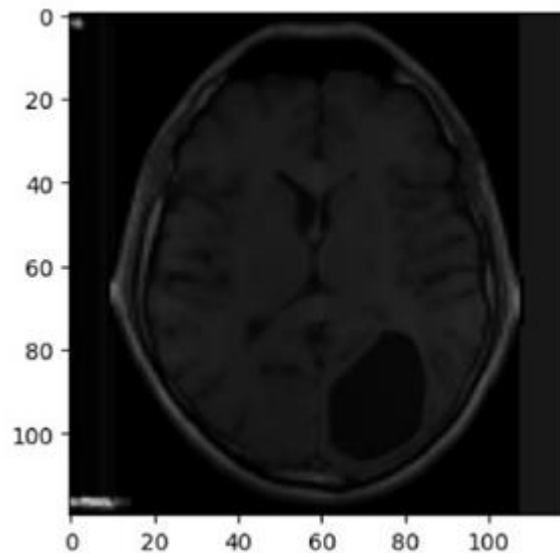


Fig. 4 MRI after normalization

- 2) **True Negatives:** Accurately predicted non-instances
- 3) **False Positives:** Incorrectly predicted cases (aka type I errors)
- 4) **False Negatives:** Incorrectly estimated non-instances (aka type II errors)

In simple words, both True Negatives (TN) and True Positives (TP) are accurate predictions. When you mistakenly mark a negative instance as positive, it is known as false positive (FP). False Negatives (FN) are mistakes where you mark a positive case as negative. The confusion matrix given by our model which uses categorical cross entropy as the loss function is as shown by Fig. 2. where in 19 cases were correctly predicted as “Tumor” whereas 14 cases were correctly predicted as “No Tumor”. Only 1 case was incorrectly predicted as “No Tumor” which was actually from class “Tumor” i.e False Negative.

E. Saving the model

Save the trained model, so it can be used to predict the class that is “Tumor”, or “No Tumor” from unseen images.

IV. EXPERIMENT AND ANALYSIS

The proposed model demonstrated strong performance in Brain MRI classification. The model has been trained with augmented data and the model was then evaluated using K-fold crossvalidation, with a batch size of 16 and a value of k as 5. The number of epochs was also set to 5. With an average accuracy of 95.89%, the model produced a high percentage of accurate classifications. Precision and recall then quantify how well positive classes are discovered and reduce false negatives and false positives. The mean of 0.122265 says that the model can effectively learn and uncover the hidden features of the data.

Fig. 6 shows a comparison of results obtained using different loss functions and epochs for a CNN model. Three loss functions were tested: Binary Cross-Entropy without augmentation of the data, Categorical Cross-Entropy (8 epochs), and Categorical CrossEntropy (5 epochs). The evaluation was done and the following are the key findings:

Loss Function: All loss functions yielded acceptable results, Categorical Cross-Entropy with 5 epochs demonstrated the highest Accuracy (0.95), Precision (0.98), Recall (0.94), and F1-score (0.96). Decreasing the number of epochs to 5 for Categorical Cross-

Entropy led to an improvement in performance compared to 8 epochs. Overall performance of the model was strong across all metrics, particularly with Categorical Cross-Entropy with 5 epochs.

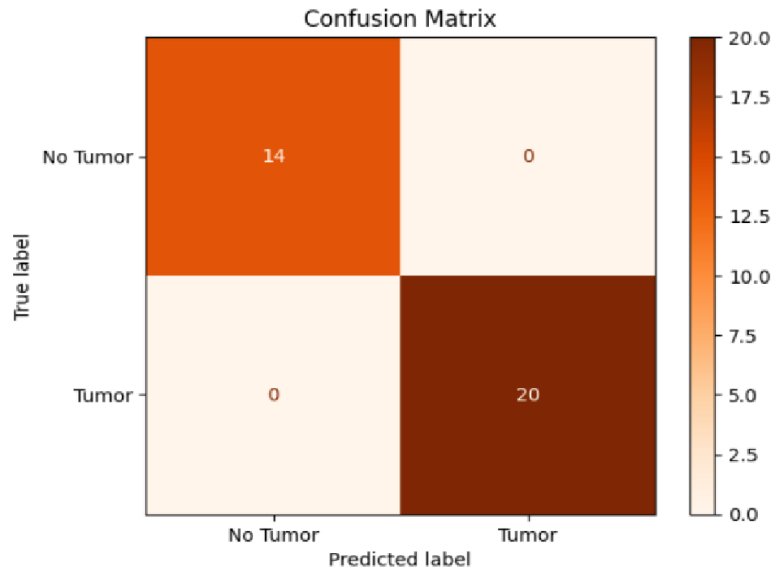


Fig. 5 Confusion Matrix final model

V. CONCLUSIONS

Fig. 5 shows the confusion matrix of the final model, this model was proposed to enhance brain tumor Detection using the CNN. So, after applying K-fold cross-validation on the MRI dataset we got better accuracy and least amount of loss which indicates that the model is well trained. Without applying augmentation, the model was overfitting with the 100% accuracy score. Since our dataset was very small so we applied augmentation for better result and that helped in removing the overfitting of the model so, we got the accuracy of 96%.

Loss Function Type	Binary cross entropy loss	Categorical cross-entropy loss (epochs=8)	Categorical cross-entropy loss (epochs=5)
Accuracy	0.85	0.93	0.95
Precision	0.86	0.93	0.98
Recall	0.83	0.94	0.94
F1 Score	0.83	0.93	0.96

Fig. 6 The comparison of loss functions

ACKNOWLEDGMENT

We acknowledge the limitations imposed by the relatively small dataset used in this study. The limited data availability may have impacted the model's generalization performance. Future research with bigger and more varied datasets could further enhance the model's capabilities and robustness. Also, the model's average accuracy score is very much acceptable. We deeply appreciate everyone who assisted us in completing this research and for their support. Above all, I would like to express our sincere gratitude to Mrs. Sunita Jena, the assistant professor, and Ms. Niloufer K. Kotwal, the head of the department of Life Sciences, for their helpful guidance, sensible recommendations, and constant encouragement during this project. We also like to thank Jai Hind College's MSC Big Data Analytics Department for providing the tools, space, and assistance we required to do this study. I want to express my gratitude to everyone listed above as well as to everyone who helped with this research, whether directly or indirectly.

REFERENCES

- [1] "American Oncology Insitute," American Oncology Insitute, 18 09 2024. [Online]. Available: <https://www.americanoncology.com/blogs/the-role-of-mri-andother-imaging-techniques-in-brain-tumor-diagnosis>. [Accessed 28 10 2024].
- [2] N. Kumar, "NeuroScan: Brain Tumor Detection using Convolution Neural Network," *Indian Scientific Journal Of Research In Engineering And Management*, vol. 3, 2024.
- [3] M. A. A. N. S. M. and N. A. , ""Brain Tumor Detection and Classification Using an Optimized Convolutional Neural Network"," *Taibah University*, 2024.

-
- [4] A. K. S. G. Komal Sharma, "Brain Tumor Detection based on Machine," *International Journal of Computer Applications* (0975 – 8887), p. 10, 2014.
- [5] N. P. K. N. S. K. S. N. and B. P. S. , "Tumor Detection using threshold operation in MRI Brain Images," Natarajan, P., Krishnan, N., Kenkre, N. S., Nancy, S., & Singh, B. P. (2012). *Tumor detection using threshold operation in MRI brain images. IEEE International Conference on Computational Intelligence and Computing Research*, 2012.
- [6] J. Seetha and . S. S. R. , "Brain Tumor Classification Using," *Biomedical & Pharmacology Journal*, vol. 11(3), pp. 14571461, 2018.
- [7] S. Deepak and P. A. , "Brain tumor classification using deep CNN features via transfer learning," *Computers in Biology and Medicine*, vol. 111, 2019.